A STUDY OF PAY AT THE WASHINGTON POST

November 6, 2019
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>INTRODUCTION</td>
<td>3</td>
</tr>
<tr>
<td>PAY ANALYSIS</td>
<td>5</td>
</tr>
<tr>
<td>Full workforce</td>
<td>5</td>
</tr>
<tr>
<td>Newsroom</td>
<td>8</td>
</tr>
<tr>
<td>Commercial</td>
<td>15</td>
</tr>
<tr>
<td>TESTIMONIES</td>
<td>19</td>
</tr>
<tr>
<td>RECOMMENDATIONS</td>
<td>22</td>
</tr>
<tr>
<td>PAY HISTORY AT THE POST</td>
<td>24</td>
</tr>
<tr>
<td>PAY STUDIES AT OTHER COMPANIES</td>
<td>26</td>
</tr>
<tr>
<td>METHODOLOGY</td>
<td>27</td>
</tr>
<tr>
<td>CONTRIBUTORS</td>
<td>28</td>
</tr>
<tr>
<td>APPENDIX</td>
<td>29</td>
</tr>
</tbody>
</table>
INTRODUCTION

Washington Post employees work every day to ensure that our company is a leader in the journalism industry. Members of our company’s union, The Washington Post Newspaper Guild, believe The Post should also lead the way in how it treats its staff.

We want to foster an environment where all people, regardless of their gender, race, religion, sex, age or job, feel they are heard, respected and paid fairly.

Our company has expressed a commitment to these values. But the members of the Post Guild believe that true progress can only be achieved when we begin with the facts. And the facts tell us that The Post has a problem with pay disparity.

The Post has never conducted and released to the public a comprehensive pay study of its own. So this year, Post Guild decided to do one itself.

Our union contract with Post management mandates that the company give us pay data on Guild-covered employees on an annual basis. We requested this information in July 2019 and spent four months analyzing the data, a reporting effort led by Pulitzer Prize-winning data journalist Steven Rich and supported by a team of dozens of other Post Guild members. We took care to protect the integrity of the data and the privacy of our colleagues.

The result of those efforts is this new report — the most comprehensive study to date of pay at The Washington Post.

This is what we found.

IN THE NEWSROOM

• **Women as a group are paid less than men.**

• **Collectively, employees of color are paid less than white men,** even when controlling for age and job description. White women are paid about the median for their age. Women of color in the newsroom receive $30,000 less than white men — a gap of 35 percent when comparing median salaries.

• **The pay disparity between men and women is most pronounced among journalists under the age of 40:** When adjusting for similar age groups, which in most cases is a good stand-in for years in journalism, it becomes clear that the pay disparity between men and women exists almost exclusively among employees under the age of 40.

• **Men receive a higher percentage of merit pay raises than women,** despite accounting for a smaller proportion of the newsroom.

• **The Post tends to give merit raises based on performance evaluation scores, but those who score the highest are overwhelmingly white.** The Post is fairly consistent across races/ethnicities and genders at awarding raises to those who do well on performance evaluations. But in 85 percent of instances in which a 4 or higher was awarded to a salaried newsroom employee, that employee was white. Employees are rated on a scale of 1 to 5. On the flip side, 37 percent of scores below 3 were given to employees of color in the newsroom (the newsroom is about 24 percent nonwhite).

• **Pay disparities have narrowed from the Graham era to the Bezos era,** but most have not shrunk to within what could be considered parity.
IN THE COMMERCIAL DIVISION
• Men and women are paid about the same. Gender pay disparities are nearly nonexistent among salaried employees in the commercial division’s nine departments.
• Pay disparities do exist, however, when analyzing for race or ethnicity. The median salary for white employees in commercial is $88,000, compared with $83,445 for people of color — a difference of $4,555, or 5 percent.
• The disparity is even larger when adjusted for age, suggesting that employees of color in commercial are paid less than their white peers despite having more experience.

The Guild recognizes that these are complicated problems and reflect deeply entrenched disparities in our society. But we believe the company can and must make a significant and urgent effort to address them.

The results of the study were shared with the company ahead of publication. Members of the Guild also met with representatives of Post management to review the findings and invited management to respond. The company declined to comment. If The Post disagrees with any of the Guild’s conclusions, we welcome the company to conduct and share a study of its own.

We must note that the ability to analyze pay disparities at The Post has been hindered by the company’s lack of specific data on the professional experience of its employees, who sometimes have built lengthy careers before joining The Post. The relative lack of diversity at The Post, particularly the relatively low numbers of black and Hispanic or Latino newsroom employees, also complicated our analysis because of the small sample sizes — but in itself demonstrates that the company must do better to recruit and retain a diverse staff.

We know there are common-sense steps the company can take to eliminate these disparities, and we have outlined a list of those recommendations at the end of this report.

We believe in The Washington Post’s ability to do better. We want to help our company get there. This is our guide to making it happen.
PAY ANALYSIS

THE POST’S FULL WORKFORCE
Among all current Guild-covered employees, about two-thirds (707 in total) are salaried. Among those employees, the mean salary is $112,383, while the median salary is $99,904. The median salary is generally a better metric for salaries. The higher mean suggests that the highest salaries have skewed the average upward.

The other third of employees (243 in total) are hourly. The median hourly rate for those employees is $29.23 an hour. While this study will largely focus on salaried employees, some sections will analyze hourly employees. The data does not have a field for hours worked per week or average hours worked per week, so take-home pay is difficult to discern — a major difficulty encountered in this study.

Conducting a pay study of an organization like The Washington Post is not easy. Because the organization isn’t flat, meaning not everyone with the same amount of experience is working the same job, topline numbers such as median salary by gender or race and ethnicity cannot capture the entire story of pay at The Post. Those figures presented here should be understood primarily as a starting point for discussion. Ultimately, the goal in this report is to add nuance to this analysis, and demonstrate truer metrics for pay at The Post, accurately capturing the landscape to determine where the organization has genuine pay parity, and where it has disparities.

Of all current employees, 53.4 percent are female and 46.6 percent are male. Among salaried employees, 52.3 percent are female and 47.7 percent are male. The median salary for the 337 salaried male employees at The Post is about 20 percent higher than the median salary for the 370 salaried female employees: $109,928 compared with $91,816. One potential reason for some of the $18,000 disparity is the median age of each gender. For men, the median age is 41, while the median age of women is 35.

COMPANY-WIDE

Gender of Guild-eligible Post employees

<table>
<thead>
<tr>
<th></th>
<th>Female</th>
<th>Male</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total</td>
<td>507</td>
<td>443</td>
</tr>
</tbody>
</table>
To study employees by race and ethnicity, the Guild again relied on information provided by management, which means the provenance of the data was unclear. The information on race and ethnicity was combined into just one field, which prevented the Guild from separating the two for analysis. Not every employee has a race or ethnicity listed, but the vast majority do. Only 22 current employees, just over 2 percent, do not have this information listed in the database.

**Of 950 current employees, the racial and ethnic breakdown is as follows:**
- White employees: 64.4 percent
- Black employees: 16.5 percent
- Asian employees: 8.1 percent
- Hispanic or Latino employees: 4.7 percent
- Employees with two or more races: 1.9 percent

**For the 707 salaried employees, the racial and ethnic breakdown is as follows:**
- White employees: 71.4 percent
- Black employees: 8.8 percent
- Asian employees: 8.3 percent
- Hispanic or Latino employees: 4.7 percent
- Employees with two or more races: 2 percent
The median salary by race and ethnicity for those salaried employees is as follows:

- White employees: $102,880
- Black employees: $91,881
- Asian employees: $90,780
- Hispanic or Latino employees: $82,000
- Employees with two or more races: $79,860

### COMPANY-WIDE

**Median pay by race and gender (salaried employees)**

- **White men**: $111,035
- **Black men**: $99,931
- **White women**: $95,780
- **Men of two or more races**: $94,875
- **Asian women**: $91,115
- **Hispanic or Latino men**: $90,780
- **Asian men**: $90,431
- **Men of undisclosed race**: $88,280
- **Black women**: $87,808
- **Hispanic or Latino women**: $80,250
- **Women of two or more races**: $75,000
- **Women of undisclosed race**: $73,000
For the 290 salaried male newsroom employees working at The Post, the median salary is $116,065. For the 284 salaried female newsroom employees, it is $95,595. These groups have disparities in age and years of service: The median age for men working in the newsroom is 41, compared with 35 for women.

When adjusting for similar age groups, which in most cases is the best stand-in for years of experience in journalism included in the data, it becomes clear that the pay disparity between men and women exists almost exclusively among employees under the age of 40. For men and women 40 and over, the median salaries are separated by less than 1.5 percent: $127,765 compared to $126,000, respectively.

For men and women under the age of 40, the gap is more than 14 percent, with the median salary for men at $95,890, compared to $84,030 for women. It’s unclear why this topline disparity exists only for this age bracket. One possible explanation is a hiring disparity in positions that The Post considers more prestigious, and therefore higher-paying. Another explanation might be the pay disparities across races and ethnicities: The younger women at the organization are more diverse.

For 96 hourly employees across the newsroom, there is virtual pay equity. The median hourly wage for men
is $33.33, compared with $32.75 for women. Comparing by age is difficult because only 33 men work hourly jobs in the newsroom and their ages vary widely. That said, there is virtual pay parity between male and female hourly workers under the age of 40. Women make more money than men working hourly over 40, but the sample size for men is low (15), meaning a few employees can lower or raise the median fairly drastically.

In the newsroom, 71 percent of salaried Guild-eligible employees are white and 24 percent of employees are nonwhite. Below are the median salaries by race and ethnicity across the newsroom:

- White: $106,212
- Black: $97,276
- Asian: $95,205
- Hispanic or Latino: $82,890
- Two or more races: $79,860

The total gap between white journalists in the newsroom and journalists of color is more than 15 percent, with a median salary of $106,212 for 406 white journalists and $92,080 for 139 journalists of color. Much like the gender gap, some of this could be explained by age and years of service: White journalists have a median age of 40 and journalists of color have a median age of 36.
But age doesn’t explain everything. Young employees of color across the newsroom don’t have complete parity with their young white colleagues. Among those under 40, newsroom employees of color make about 7 percent less than white journalists, with median salaries of $84,780 and $90,780, respectively. The disparity widens for journalists 40 and over: Newsroom employees of color have a median salary of $110,845, while their white colleagues have a median salary of $128,484 — a gap of nearly 16 percent.

![Median pay by age and race (salaried employees)](image)

*Data is not displayed for six employees who preferred not to disclose their race*
About two-thirds of hourly employees in the newsroom are white, making this category more diverse than salaried workers. However, a racial pay gap still exists among hourly employees, with white employees making a median wage of $33.59 an hour, compared to $30.07 for employees of color. That gap of 11.7 percent is well outside the range that the Guild would consider parity in pay. When accounting for age, gaps still exist, though the analysis is difficult because there are only 30 hourly employees of color in the newsroom.

We would be remiss if this study did not examine gender, race and ethnicity through an intersectional lens. Across most industries, disparities increase when multiple factors are taken into account. Our analysis shows a similar pattern. The median salaries by group are as follows:

- White men: $117,452
- Men of color: $101,575
- White women: $99,640
- Women of color: $86,511
The gender pay gap is fairly similar across races and ethnicities. White men make about 18 percent more than white women, and men of color make 17 percent more than women of color across the newsroom. Likewise, racial pay gaps are similar across genders, with white men making about 16 percent more than men of color and white women making 15 percent more than women of color.

When comparing white men in the newsroom to women of color in the newsroom, the gap is over 35 percent, with the median salaries separated by more than $30,000. Here, again, some of this may be attributed to the fact that white men working at The Post have the oldest median age of any group across the newsroom. Median ages by group are as follows:
- White men: 41
- Men of color: 40
- White women: 37
- Women of color: 33

Controlling for age does, in fact, close the gap significantly between white men and men of color and also between white women and women of color. The gender gaps remain fairly consistent.

The Guild attempted to determine median salary by age group as a way to analyze pay by gender and race and ethnicity, and to determine which groups were paid above and below those benchmarks when age disparities were corrected. Controlling for age, here is how the median salaries for the four groups stack up:
- White men are paid an average of 7.27 percent higher than the median for their age group
- Men of color are paid an average of 1.73 percent lower than the median for their age group
- White women are paid an average of 0.14 percent higher than the median for their age group
- Women of color are paid an average of 3.26 percent lower than the median for their age group

Based on this analysis, The Post underpays men and women of color relative to white men. It pays white women about the median for their age.

One unexplored detail in this analysis is how desks factor into the equation: Do disparities exist within desks, and if they do, to what degree? For this, the Guild grouped cost centers into desks where appropriate. But many desks are not captured in the analysis because they are too small to evaluate.

In general, we found the same pattern of disparities throughout the newsroom, but also discovered that desks with some of the highest median salaries — such as National, Financial and Investigative — also had higher percentages of white men. This suggests that The Post must do more to cultivate women and people of color for those desks that demand the highest levels of skill and experience and therefore command the highest salaries.

Of 14 desks that have at least five men and five women, the median pay for men is at least 5 percent higher on 10 desks. Two have a median pay disparity that favors women by at least 5 percent, and two have approximate pay equity between the genders (within 5 percent). Of 10 desks for which there were at least five white journalists and five journalists of color, seven desks have a median pay disparity favoring white journalists by at least 5 percent. Zero have a median pay disparity favoring journalists of color by at least 5 percent, and three have approximate pay equity between the two (within 5 percent).

There are only three desks that have at least five white men, five white women, five men of color and five women of color: National, Local and Design. All three have racial and gender pay disparities. Of those three desks, all have disparities in median salary of more than 30 percent between white men and women of color.
One prominent factor for these pay disparities is that the higher the median salary is for a desk, the higher its percentage of journalists who are male and the higher percentage of journalists who are white.

For desks in which the median salary is higher than $125,000, 80 percent of journalists are white and 57 percent of journalists are men. Those desks include National, Financial and Investigative. In this group, 47 percent of journalists are white men and 10 percent are women of color. For desks in which the median salary is below $92,000, 68 percent of journalists are white and 40 percent of journalists are men. In this group, 28 percent of journalists are white men and 21 percent are women of color.

In an equitable pay environment, one would expect that 50 percent of people in each group would be above the median salary and that 50 percent would be below. Controlling for age and median desk salary, the following represents how many employees are above the median expected salary:

- White men: 57.6 percent
- White women: 48.9 percent
- Men of color: 41.2 percent
- Women of color: 38.5 percent

The deviation from the median for each of these groups when controlling for age and median desk salary is as follows:

- The median salaried white male employee makes $2,448 a year more than the expected median salary for their age and assignment
- The median salaried white female employee makes $14 a year less than the expected median salary for their age and assignment
- The median salaried male employee of color makes $407 a year less than the expected median salary for their age and assignment
- The median salaried female employee of color makes $1,360 a year less than the expected median salary for their age and assignment

We recognize that these groups aren’t monoliths, and so in a normal distribution of pay, the Guild would expect about a third of employees to make within 5 percent of the median for their age and desk, about a third to make more than 5 percent below and about a third to make more than 5 percent above. Those distributions also show disparities among these groups.

- White men: 32.4 percent of employees make more than 5 percent below the expected median salary and 48.1 percent make more than 5 percent above the expected median salary
- White women: 38 percent of employees make more than 5 percent below the expected median salary and 35.3 percent make more than 5 percent above the expected median salary
- Men of color: 41.2 percent of employees make more than 5 percent below the expected median salary and 29.4 percent make more than 5 percent above the expected median salary
- Women of color: 46.2 percent of employees make more than 5 percent below the expected median salary and 25.6 percent make more than 5 percent above the expected median salary

The data also shed light on who received raises over the past five years and their performance evaluation scores for the past four.

For men and women in the newsroom, the median performance evaluation score is even, at 3.4 for 3,664 evaluations conducted over four years. Analyzed by race and ethnicity, scores started to diverge. Among groups for whom more than 20 evaluations were done over the four years from 2015 through 2018, the median performance ratings were as follows:

- White: 3.5
- Asian: 3.4
For men, performance ratings were always at least equal to those of their female counterparts of the same race or ethnicity. Ratings for men and women by race and ethnicity were as follows:

- White men: 3.5
- White women: 3.4
- Asian men and women: 3.4
- Hispanic or Latino men and women: 3.3
- Black men: 3.3
- Black women: 3.25
- Men and women of two or more races: 3.2

It is unclear what accounts for these disparities in performance evaluation scores.

Most pay raises in the newsroom are a result of Guild-negotiated contracts that award across-the-board increases in salaries. In addition, people also receive merit pay increases, which are based on performance evaluations and awarded solely at the discretion of management. Merit pay increases account for 26 percent of all raises. They are an important way to reward effort and initiative but can also create and magnify disparities if the company does not take steps to ensure fairness in the way performances are evaluated and rewarded.

Men received 51.7 percent of those merit raises, and women received 48.3 percent. (The newsroom’s gender makeup is 48.2 percent men and 51.8 percent women.)

The percentages of merit raises distributed by race and ethnicity for salaried journalists are as follows:

- White: 75.7 percent
- Black: 9.3 percent
- Asian: 8.3 percent
- Hispanic or Latino: 3.6 percent
- All others: 3.1 percent

For contrast, over that time the racial and ethnic makeup of salaried employees is as follows:

- White: 70.1 percent
- Black: 9.1 percent
- Asian: 8.5 percent
- Hispanic or Latino: 4.6 percent
- All others: 7.7 percent

The Post contends that merit raises are tied mostly to performance evaluations, and the data bears that out. Those who score higher, regardless of race, ethnicity or gender, tend to be the ones who get merit raises most frequently. However, an analysis of every performance evaluation score over the past four years shows that those who score the highest are overwhelmingly white.

In cases in which a 4 or higher was awarded to a salaried newsroom employee, 85 percent were white, and over half of scores of 4 or higher were awarded to white men. And in cases in which salaried newsroom employees were given a score of 3 or below, 37 percent of those scores were given to employees of color (the newsroom is about 24 percent nonwhite).
GRAHAM FAMILY ERA VS. BEZOS ERA
Finally, the Guild wanted to examine whether pay disparities had changed after Amazon founder Jeff Bezos bought The Washington Post in 2013 from the Graham family. Analysis shows that from the Graham era to the Bezos era, pay disparities have shrunk slightly, but not to within a range that the Guild would recognize as the point of parity. While white men are paid closer to the median salary across age groups, all other groups are also closer to white men than they were before.

Overall, the pay disparity between current white newsroom employees and current newsroom employees of color who were hired under the Graham family is 12 percent. For current employees hired after Bezos acquired the newspaper, that disparity is down to 5 percent. In particular, the gender pay gap has narrowed for new hires. Whereas the disparity between men and women who were hired under the Graham era is 5 percent, that figure is down to 3 percent for current employees hired after Bezos purchased the paper. A big drop occurred between white men and women of color, down from a 16 percent disparity to an eight percent disparity.

For current employees hired under Bezos’s ownership, salaries when accounting for age (years in journalism) and median desk pay are as follows:
- White men are paid an average of 9 percent higher than the median for their age group and desk
- White women are paid an average of 6 percent higher than the median for their age group and desk
- Men of color are paid an average of 3 percent higher than the median for their age group and desk
- Women of color are paid an average of 1 percent higher than the median for their age group and desk

COMMERCIAL
Analysis of the organization’s commercial side as a whole is difficult, because it includes nine different departments across the organization and only 133 salaried employees and 147 hourly employees. These numbers are large enough for topline analysis, but with the introduction of more and more factors, the results become less reliable. In many departments, it is difficult to ascertain pay equity or disparity across races/ethnicities and genders because they have too few employees.

The following section attempts to examine pay where possible. As with the newsroom section, topline numbers are only reliable insofar as they reveal broad trends, though they cannot capture other factors that influence pay, such as years of experience or the demands of the job itself.

About one-third of Guild-covered employees work on the commercial side of The Post. Overall, gender pay disparities are nearly nonexistent. For the 47 salaried men working in commercial at The Post, the median salary is $86,880; for the 86 women, it’s $85,977.

The difference of 1 percent indicates that the two groups are effectively on par. That equity does not hold when it comes to race and ethnicity, though disparities are smaller on the newsroom side. For the 99 salaried employees on the commercial side who are white, the median salary is $88,000; For the 32 employees of color working alongside them, the median salary is $83,445. That disparity of 5.5 percent is lower than the disparity for commercial’s hourly employees.

For the 43 hourly employees in commercial who are white, the median hourly rate is $30.38; for the 101 employees of color working alongside them, the median hourly rate is $25.16. That represents a disparity of more than 20 percent.
When adjusting for age, the disparity grows even larger. The gap between the median ages of employees of color and white employees overall is five years; the median age for white salaried employees is 35, and for employees of color it is 40. For hourly employees, the median ages are 39 and 47, respectively. So employees of color in commercial tend to be paid less than their white counterparts, despite tending to have more experience in their jobs.

Examining race and ethnicity within genders reveals an interesting pattern. Among women, the racial pay disparity is quite low. The 67 salaried white women in commercial have a median salary that is 1.3 percent higher than that of the 17 women of color. This gap falls within the range that the Guild considers approximate pay parity. The women are clustered around the median salary for all of commercial.

But between white men and men of color, the pay disparity is stark. For the 32 white men, the median salary is $94,497; for the 15 men of color, the median salary is $76,866. That disparity is 22.9 percent, one of the highest disparities seen across the entire organization.
This trend is different for hourly employees. In many workplaces, the biggest pay gap tends to exist between white men and women of color, but in commercial, the biggest gap among hourly employees is the gap between white women and men of color. In fact, the gap between white men and women of color is almost nonexistent. The median hourly rate for the 21 white men in commercial is $26.76, compared with $26.54 for the 52 women of color — a difference of just 0.8 percent. On the other hand, the median hourly rate for 22 white women is $31.76 and for the 49 men of color is $23.33, a 36.1 percent disparity.

One of the issues with attempting to determine disparities across commercial is that the data does not allow us to determine how many hours an employee works. The data distinguishes between full- and part-time hourly employees but does not count how many hours part-time employees work (which would be difficult, because of the fluidity of many part-time schedules).

When analyzing just the full-time hourly employees in commercial, the disparities shrink and shift. For the men and women of color, the wages are virtually the same for full-time employees on an hourly wage. For white men and women, hourly wages are about $2 apart. But while full-time men and women of color have median hourly rates of $26.04 and $26.82, respectively, full-time white men and women have median hourly rates of $29.91 and $31.84, respectively.

There is a slight gender disparity in the scores that commercial employees have received on performance evaluations. While the median score for women is 3.3, the median score for men is 3.2. Similarly, when it comes to race and ethnicity, while white and Asian employees in commercial each have a median score of 3.3, black employees have a median score of 3.2 and Hispanic or Latino employees each have a median
score of 3.15.

Finally, an analysis of raises for The Post’s commercial employees shows few disparities in who receives them. The most common type of raise among commercial employees, with the exception of those mandated by Guild contracts, is merit raises.

Men received 44.3 percent of those merit raises, and women received 55.7 percent. (The gender makeup of commercial is 42.9 percent male and 57.1 percent female.)

The percentages of merit raises distributed by race and ethnicity for salaried commercial employees were as follows:
- Black: 47.4 percent
- White: 40.7 percent
- Asian: 7.1 percent
- Hispanic or Latino: 3.5 percent
- All others: 1.3 percent

For contrast, over that time the racial and ethnic makeup of salaried employees was as follows:
- Black: 37.5 percent
- White: 47.8 percent
- Asian: 7.3 percent
- Hispanic or Latino: 4.2 percent
- All others: 3.1 percent

Merit raises went to these groups in the following percentages, compared to the percentages they make up in commercial:
- White men: 17.1 percent of merit raises vs. 18 percent of employees
- White women: 23.6 percent of merit raises vs. 29.8 percent of employees
- Men of color: 27.1 percent of merit raises vs. 24.9 percent of employees
- Women of color: 31.9 percent of merit raises vs. 27.3 percent of employees

**ADDITIONAL ANALYSIS**
The analysis provided above is a fraction of the analysis completed as part of this pay study. If we had written it all up, this report would be much, much longer. The numbers in this report represent the most relevant topline numbers in the analysis, and all attempts were made to present those numbers alongside context including factors such as age and job.

If you are interested in seeing the Post Guild’s full analysis, it is attached as an appendix to this report.
In interviews, employees across desks and departments described troubling pay disparities between colleagues doing the same job, even with the same level of experience. They expressed frustration with a system that encourages employees to seek offers elsewhere in order to receive significant raises. They spoke of a hiring process that benefits industry insiders coming from higher-paying competitors and that too often sets back women, people of color, and journalists from smaller publications.

The employees that the Post Guild interviewed come from a variety of departments in the newsroom, including Video, Photo, Local, Foreign, Style and Graphics, as well as from commercial. All asked for their names and certain identifying details to be withheld.

Veteran employees hired as interns or entry-level staff members described being pigeonholed into jobs with slow, incremental raises and limited opportunities for substantial salary boosts. One news reporter hired as an intern nearly 20 years ago said he still makes less than $70,000 a year. He’s thinking about getting a part-time job, he said, possibly driving for Uber.

“This is not to disparage anybody ... but it’s tough to sit there and look at someone who’s 15 years younger than you, with 10, 15, 20 years less experience, making significantly more than you,” he said. “It feels like a caste system.”

One veteran reporter said it took her more than two decades at The Post to feel that she had any substantial disposable income. At one point, while working as a foreign bureau chief, she learned that the man who previously held her job, a reporter of the same age with more managerial experience but a fraction of her experience at The Post, was making $50,000 more than her.

Another female employee, a 35-year-old award-winning journalist who started as an intern in the mid-2000s, recently found out that all of the men on her team are paid more than her — even though she’s been at The Post longer than all of them and has been working in journalism longer than most of them. One of the men on her team is paid more than $30,000 more than her.

While she has received incremental raises, The Post only gave her a substantial raise after a competitor offered her a job several years ago.

“It’s always disgusted me that the only way we can get what we deserve is by getting an offer somewhere else,” she said. “How is that a way to show that you value someone?”

She has noticed a pattern of young women like her being hired at low salaries and getting “stuck.”

“We don’t know what we should be paid, because no one tells you that in college,” she said. “You take what you’re offered, and then you’re on this track.”

One 32-year-old female newsroom employee was hired about a year and a half ago after two years in journalism and about a decade of other related job experience. The Post job application asked her how much she made in her previous job, as a fellow at a nonprofit journalism outlet, and how much she would like to make in her new job.
Not long after she joined The Post, she found out that a colleague who started on the exact same day, in the same position, with a similar level of experience, was hired at a salary $14,000 higher than hers. That colleague told her that before she was hired, she had known another member of the team who encouraged her to negotiate for a higher salary. The 32-year-old employee, meanwhile, did not know that she could negotiate upward of 20 percent of her salary upfront.

She said she then went through the salary review process, which concluded that she was, compared to her colleagues and other market rates, underpaid by at least $10,000. But neither the information nor the process that furnished it guaranteed corrective measures of any kind.

In her annual review this year, she did get a sizable merit raise, which cut her pay disparity with her colleague in half. Still, she said, “it’s this continual uphill battle of just trying to get even.”

“That money should really not be used for corrective purposes,” she said of the merit raises. “It’s very hard, once you start at a lower point, to ever fully play catch-up. “Because as I’m playing catch-up, people are getting actual merit increases. There’s no — as far as I can see — clearly defined process or path for correcting these pay inequities.”

One 29-year-old female employee started at The Post more than five years ago in an entry-level job at a low salary. The role was a demotion from her former job but was described to her as a starting point with opportunity for advancement. She was quickly promoted to editing roles with demanding evening and weekend hours. While she has received raises almost every year, she recently learned that at least three colleagues on her team, at her same level, make tens of thousands of dollars more than her.

“It made me feel undervalued and made me question what I’m doing and if this is really a sustainable place to be,” she said. “Can I actually buy a house here? I don’t want to just live this kind of life forever.”

She wants to ask how she can bump up her salary, but she isn’t sure how or when to have that conversation with her supervisor.

“I’ve definitely noticed morale sinking for me,” she said.

On the commercial side, one employee in Client Solutions described a lack of opportunities for upward mobility, especially for people of color. The employee, a 40-year-old woman of color who has worked in Client Solutions for more than five years, said she has been trained to manage new departmental tasks outside of her role. But when she has expressed interest in applying for available positions, she has not been considered. Instead, she has taken on new responsibilities without a pay increase.

She would ideally like to transition to a role that is more in line with her new skills and training, but she has seen the department focus its hiring on young people fresh out of college. At her age, she said, “I’ve already accepted that that’s not an opportunity for me.”

The Post Guild also found pay disparities among newsroom aides, whose salary floors are much lower than those of other employees. One newsroom aide, a full-time staffer, was hired at a salary of less than $40,000 a year. As a recent college graduate, the aide did not negotiate for a higher salary.

“I think I was scared of them rescinding my job offer,” the aide said. “I feel so stupid for not doing it ... It makes me so upset to even think about it. But I didn’t even think to negotiate or ask for more money.”

The employee has since found out that another aide, who is the same age, has a similar level of experience
and was hired around the same time, makes $4,000 more. Another slightly more senior aide makes about $12,000 more.

The aide thought back to orientation day at The Post, when new employees were told they had been hired for long careers at the company. With such a low salary and an unclear path for moving up, the employee wondered how likely such a career at The Post might be.

“It just feels very almost disingenuous,” the employee said.
The Guild believes that a diverse, equitable and inclusive workplace is key to The Post’s success as an organization. While the company has made progress, there is still work to do. To help promote such an environment, the Guild offers the following recommendations:

**The Post should strengthen and better formalize the salary review process.** This process — which the Guild negotiated for in its last contract — is intended as a mechanism to empower employees to understand how much they are being paid in relation to their co-workers, ideally providing the groundwork for salary negotiation conversations with their direct managers. Currently, only one Post employee conducts salary reviews for the entire company. We believe this is too much work for just one person, and we recommend that more Post employees be trained on the salary review process to ensure reviews are completed in a reasonable time frame. Furthermore, it should be made clear in new employee onboarding and orientation that employees may request a salary review at any time. We also recommend that the organization conduct salary reviews for all employees hired in the past five years. Employees who go through the salary review process should also have the option of having a Guild representative present during the process if they wish. We believe these enhancements to the salary review process can be accomplished in a reasonable time frame and would signal that the company values empowering its employees to have these conversations.

**The Post should allow direct managers to know how much their reports make.** Currently, many direct managers do not know this information, despite being in the best position to negotiate raises on their reports’ behalf. Also, these managers are often responsible for having salary conversations with new hires and should be appropriately informed about pay expectations. This would empower direct managers to look out for and be aware of pay disparities on their teams and help correct them, taking some of that burden off top management.

**The Post should ensure that pay disparities do not begin during the hiring process.** The Post currently requires prospective applicants to list their salary history and desired salary during the application process. We believe the company should remove these questions — that of previous salary history and desired salary — entirely from the application and interview process. These questions often serve to perpetuate gender pay gaps, and have already been banned from states such as California and cities such as New York. The company should not use these questions as a proxy to determine applicants’ starting salaries. As one of the largest and best-resourced newsrooms in the nation, The Post clearly has enough information to determine what the expected salary for a Post-caliber position should be.

**The Post also should re-evaluate the existing two-year intern program.** For many years, The Post would hire summer interns into full-time positions but classify them as “two-year interns.” Two-year interns are essentially full-time Guild-covered employees with benefits but are slotted into a salary of less than $50,000 for two years. Once they “graduate” from the two-year program, these employees are at a disadvantage with regard to pay in comparison to their peers, despite often having the same amount of experience. The Post has moved away from this model in recent years, instead hiring some interns as contractors and some as full-time staff. Many contractors eventually transition to full-time staff, but the two-year intern classification has not been eradicated. While we applaud The Post for hiring interns and giving young people opportunities, we believe this classification, if no longer being used, should be retired entirely.

**The Post must do more to ensure that the company reflects the diversity of American society.**
Post must do more to recruit and retain employees from underrepresented backgrounds, especially black and Hispanic or Latino journalists. We recommend that The Post create a new job position for a recruiter to scout talent from underrepresented backgrounds. The Post should be present and actively recruiting at journalism conferences, such as those held by the National Association of Black Journalists, the Asian American Journalists Association, the National Association of Hispanic Journalists and the Native American Journalists Association. We are inspired by the recent news that Vox Media has committed to ensuring that 40 percent of applicants who pass a phone interview round must be from underrepresented backgrounds — a stipulation negotiated by the Vox Media Union. We recommend The Post adopt a similar approach and support having diverse candidate pools for positions, as an investment in our belief that diverse, representative institutions — especially in the journalism industry — better serve their communities.

To hold the company accountable in creating an equitable and diverse workplace, we also recommend that The Post hire an equity, diversity and inclusion chair/consultant and form a diversity committee. This consultant should be hired by the end of 2020. The committee should be created as soon as possible. This consultant can help the company establish goal posts for creating a more diverse workplace, draft initiatives to support these endeavors and also act as an accountability arm. The diversity committee would work closely with this consultant, but would also create and enact initiatives that support increased equity, diversity, and inclusion throughout The Post. One such project could be creating guidelines for inclusive journalism, similar to a guide published by the Seattle Times diversity committee. This committee should be made of a mix of employees from all parts of the organization.
The fight for equal pay at The Washington Post has spanned decades of hard work, collective action and uncommon courage from past generations of Post employees.

When the Civil Rights Act of 1964 passed the Senate after the longest debate in its history, Title VII came into law, prohibiting employment discrimination based on race, sex, color, religion and national origin. The act not only prohibited discriminatory pay, but also discrimination in recruitment, hiring, assignment, promotions, benefits, discipline and layoffs.

It also created the Equal Employment Opportunity Commission, a committee tasked with eliminating employment discrimination. It was not until 1972, however, that the EEOC gained enforcement powers, unleashing a decade of legal fights for equal employment and pay, including at The Post.

In 1972, 117 female employees of The Post joined the Washington-Baltimore Newspaper Guild to file a sex discrimination suit against The Post with the EEOC. Women at The Post had seen little improvement in their pay and working conditions since signing a memorandum of understanding with management two years earlier, in 1970. The suit was finally settled in 1980 with $50 to $250 in back pay distributed to 567 women and the company’s agreement to a five-year affirmative action plan that guaranteed one-third of editorial and commercial jobs would be filled by women.

The settlement did not require the company to admit sex discrimination or a violation of Title VII — though the EEOC in 1974 had found “reasonable cause to believe” that The Post discriminated against women in salaries, promotions, and aspects of hiring.

Not everyone was thrilled with this settlement. “If this is a victory, I’d hate to see a defeat,” one female assistant managing editor told a Post reporter in 1980.

In the same year as the 1972 sex discrimination suit, black Post employees were also organizing. Nine black reporters delivered a list of 20 pointed questions to editor Ben Bradlee about the lack of black editors and reporters, particularly on prestigious desks, and the nature of their assignments. In her memoir, Dorothy Butler Gilliam relates how one of the petitioners, LaBarbara “Bobbi” Bowman, described meetings with the editor: “His hands were shaking, and I thought, ‘We have scared Ben Bradlee.’”

Seven of the original group went on to file a complaint with the EEOC about discriminatory hiring and promotion of black reporters in a city that was then over 70 percent black. They were all junior reporters, and would come to be known as the Metro Seven. The EEOC found that the group had grounds to go to court — but they lacked the resources to do so, and so never filed a lawsuit.

Despite this, Gilliam writes, the Metro Seven’s work “did cause movement inside The Washington Post newsroom. Managers broadened subject beats, increased the number of columnists of color, and stepped up promotion and hiring” of black employees.

“The progress in hiring Blacks in daily newspapers,” Gilliam writes, “was not simply due to the largesse of white editors.”

Katharine Graham herself noted in her autobiography, “Personal History,” that “without the suits and with-
out the laws adopted by the country," The Post would not have seen the significant improvements it did to the numbers and working conditions of women and people of color at The Post in the 1970s.

But the story did not end there.

In the 1980s, The Post still had a pay gap problem, and the lawsuits kept coming. A new “dual” pay system instituted by management in the 1979 contract led to widening pay gaps, lower minimum salaries, and more managerial discretion in the setting of salaries and merit increases, particularly in the newsroom.

Post management argued that experience and years spent at The Post could account for pay disparity — a position it maintained as recently as 2016 — but a 1986 pay study by the Guild showed that “in the newsroom, where management has the most flexibility in setting pay, long service at the Post and experience count less for women and blacks than for white males, who often command higher salaries even with less experience and fewer years’ service.” Among reporters in 1986, the average salary for white men was $988.68 per week. For black women, it was $791.33.

One year later in 1987, Gwen Ifill wrote in a Post Guild bulletin that “The patterns for pay discrimination that have contributed over the years to low morale and wide gaps in the amount Washington Post managers are willing to pay their employees not only persist, but have widened significantly” for women and black people since 1986. White men were then earning $240.59 a week more than black women, a gap that had widened by $43.24 from the year before. “We can’t eat prestige,” Ifill wrote.

In the years since Ifill’s bulletin, pay disparities have contracted and expanded, but never closed completely. In 1988, the Guild filed a charge of discrimination with the D.C. Office of Human Rights and the EEOC. The suit was settled in 1997 with an agreement to provide final and binding arbitration by employees asserting a claim of discrimination. Eighteen such claims were filed, with the last one settled in 2003.

The Guild conducted a pay study at The Post in 2016 that revealed significant disparities, although Post management dismissed the results and declined the Guild’s invitation to address the issue collaboratively. However, the Guild’s persistence in 2017-2018 contract negotiations resulted in a new contract section: Article XVIII(c), which empowered employees to initiate individual pay equity reviews conducted by the Human Resources department.

The advancements made by previous generations of Post employees, empowered by civil rights legislation and by the Guild, are a testament to the power of collective action and the bravery of those who put their jobs on the line in the name of equality. As the latest pay study will attest, there is still much work to be done.
The Post Guild, of course, is not the only entity to conduct a pay study. Google and Citigroup are two major companies that have recently conducted public pay equity studies.

In March 2019, Google released a summary of its 2018 pay analysis only for groups consisting of 30 employees or more, containing five or more people per demographic (i.e. women, men, minority, non-minority) to ensure statistical accuracy. Google says the purpose was to “identify any unexplained differences between groups of Googlers who are doing the same job.” Google found in one particular job code, men received less discretionary funds than women, but did not elaborate on any other job code discrepancies. Google spent $9.7 million on pay gap adjustments across 10,677 employees, 49 percent of which was spent correcting hire offers. The company was critiqued for examining only if demographic groups were being paid equally for the same job, rather than addressing which jobs and pay groups certain demographic groups are placed into.

The Citigroup study, released January 2018, differentiated between “adjusted” pay gaps, which account for “job function, level and geography,” and unadjusted, or raw, pay gaps. The adjusted analysis found negligible disparities between women and men, and between minorities and non-minorities. But the raw data showed that the median pay for women globally was 71 percent of the median for men, and that the median pay for U.S. minorities was 93 percent of the median for non-minorities. Citigroup resolved to increase representation of women and minorities in managerial levels.

A number of media unions have also conducted their own pay studies of their newsrooms, such as Bloomberg. Published in January 2019, the study found that Bloomberg BNA has a newer workforce (more than half of the bargaining unit employees having less than five years at the company) and is above average in terms of gender diversity compared to the news industry. Still, the study concluded that significant pay disparities exist. The median salary for black employees is on average $7,800 less than their white counterparts. For Hispanic employees, on average $10,609 less than their white counterparts. Pay disparities appeared to be widening for newly hired women into the commercial and IT departments. Across the board, women make less than their male counterparts and employees of color make less than their white counterparts. It’s also worth noting that white employees were found to be overrepresented in the newsroom.

While Bloomberg has made an effort to bring in new hires, the merit pay system already in place meant the pay disparities inevitably continued, and the company does not have a mechanism to balance pay. The study recommended a proactive pay policy for new hires and compensation for current employees. BBNA’s guild chair noted that she found the methodology flawed because it is difficult to quantify experience. She also noted that disparities varied across groups in the company, and that that should be kept in mind as solutions are proposed.
The Washington Post Newspaper Guild received pay data from Post management on July 2, 2019, after Alice Li and Sophie Ho, co-chairs of the Guild Diversity and Equity Committee, made a request pursuant to the Guild’s contract with The Post. On that date, data was transmitted to the Washington-Baltimore News Guild via a thumb drive, which was transferred to Steven Rich. Data was transferred to an air-gapped machine — one that wasn’t connected to the Internet — and the thumb drive was returned to the News Guild. Rich was the only member of the Post Guild granted access to the data, and data was promptly destroyed upon completion of the analysis to prevent the full data set from becoming public.

The data comprised two Microsoft Excel spreadsheets with three tabs each: one spreadsheet for The Post’s current Guild-covered employees and one spreadsheet for terminated employees, as of July 2. This only includes employees who were covered by the union’s collective bargaining agreement with The Post, regardless of whether they are dues-paying members of the Guild, and is not a complete survey of salaries across the organization; it is unclear how many Post employees, such as managers, are excluded. The data lists 950 current employees and 539 terminated employees. Here, “terminated” means terminated from the Guild, which in most cases means that the employee left The Post, but can also mean that the employee was promoted to a position ineligible for Guild membership.

The first tab of the data contains 167 fields of information about each current employee and 128 for employees who left or are no longer covered by the Guild. (Only one field in the latter doesn’t appear in the former: date of termination.) This provides a fairly comprehensive look at employees at The Post, excluding one field that the Guild is not entitled to: full legal name of the employee. It would be feasible to determine some names using the available information, but the Guild made no attempt to identify anyone in the data and, over the course of its analysis, never perused the records of individual employees.

Identifying the median salaries of small numbers of people would make it easy to discern individuals’ salaries, so the Guild took two preventive measures, while also aiming to accumulate as many accurate results as possible. The first involved grouping employees by factors such as age, race and newsroom desk. The Guild created these groups in consultation with experts who study pay trends as well as Guild members familiar with the newsroom’s structure. Second, we suppressed results from any group or subgroup that had fewer than five people, because results for small groups can be misleading. For example, in a group of three people, the person who makes the second-largest or second-smallest amount of money has the median salary. Additionally, all three members would know exactly who they are in the analysis, an outcome that seemed preferable to avoid.

To study salaries by gender, the Guild relied on a field provided by The Post. The field is binary, containing no additional information beyond “male” or “female.” It is unclear how The Post determined this information for its employees. That caveat aside, the field had information for every employee in the data set.

Analysis in Python was written by Steven Rich and audited by Aaron Williams. Analysis in r was written by Steven Rich and audited by Andrew Ba Tran.
CONTRIBUTORS

Jacob Brogan, assistant editor
Helen Carefoot, editorial aide
Hau Chu, Post Guild secretary, editorial aide
Jessica Contrera, staff writer
Dave DeJesus, Post Guild co-chair for commercial, client solutions
Rick Ehrmann, Post Guild representative
Armand Emamdjomeh, graphics reporter
Joe Fox, graphics reporter
Sophie Ho, Post Guild co-chair for Equity and Diversity Committee, audience analyst
Pat Jacob, Post Guild vice chair for commercial, account executive
Hannah Jewell, video reporter
Sarah Kaplan, Post Guild chief steward, staff writer
Fredrick Kunkle, Post Guild chair emeritus, staff writer
Marissa Lang, staff writer
Joyce Lee, video editor
Alice Li, Post Guild co-chair for Equity and Diversity Committee, video reporter
Katie Mettler, Post Guild co-chair for news, staff writer
Justin Moyer, Post Guild vice chair for news, staff writer
Razzan Nakhlawi, social media producer
Sophia Nguyen, staff writer
Lynn Olson, Post Guild vice chair for nightside, assistant editor
Steven Rich, data editor
Tonya Riley, researcher
Ric Sanchez, social media producer
Samantha Schmidt, staff writer
Matt Schnabel, multiplatform editor
Andrew Ba Tran, data reporter
Aaron Williams, data reporter
Jamie Zega, multiplatform editor
We want this study to be transparent and accessible, so we have decided to publish the results in full in this appendix section.

It contains breakdowns by job title, desk and more, but suppresses results that are less than five people to ensure more accurate results and to protect the identities of our colleagues.

This code was written by data editor Steven Rich – in two languages, Python and R. It is our hope that other news organizations within our NewsGuild family, and those fighting to unionize right now, can use our pay study as a model for how to bring transparency to their own workplaces.
Appendix A: Analysis in Python

November 6, 2019

1 Washington Post Newspaper Guild Pay Study 2019

This is the study of Washington Post Guild members' salaries based on data turned over by management of The Washington Post on July 2, 2019, pursuant to a request by members of the Guild. Management turned over two Excel files: one file detailing the salaries of current guild members working for The Post (as of the date of transmission) and one file detailing the salaries of past guild members who worked for The Post and have left the organization in the past five years.

What follows is an attempt to understand pay at The Washington Post. No individual analysis should be taken on its own to mean that disparities in pay do or do not exist. This study will start with summary analysis of trends and will dive deeper as the study goes on.

The only data manipulation done prior to analysis was taking the data out of Excel and putting the files into CSV files, converting dates from ‘MM/DD/YYYY’ to ‘YYYY-MM-DD’ and removing commas from monetary columns where values exceeded 1,000.

1.1 Importing data

[1]: from pathlib import Path

import re
import numpy as np
import pandas as pd
import statsmodels.api as sm
from statsmodels.iolib.summary2 import summary_col
from linearmodels.iv import IV2SLS
import seaborn as sns

pd.options.display.max_columns = None

pd.set_option('display.float_format', lambda x: '%.2f' % x)

[2]: BASEDIR = Path.cwd()
CSVPATH = BASEDIR.joinpath('csvs')

[3]: active_wd_schema = {
'department': str,
'employee_id': str,
'gender': str,
'race_ethnicity': str,
'date_of_birth': str,
'original_hire_date': str,
'hire_date': str,
'pay_rate_type': str,
'current_base_pay': np.float64,
'job_profile_current': str,
'time_type_current': str,
'cost_center_current': str,
'effective_date1': str,
'business_process_type1': str,
'business_process_reason1': str,
'pay_rate_type1': str,
'base_pay_change1': np.float64,
'job_profile1': str,
'time_type1': str,
'cost_center1': str,
'effective_date2': str,
'business_process_type2': str,
'business_process_reason2': str,
'pay_rate_type2': str,
'base_pay_change2': np.float64,
'job_profile2': str,
'time_type2': str,
'cost_center2': str,
'effective_date3': str,
'business_process_type3': str,
'business_process_reason3': str,
'pay_rate_type3': str,
'base_pay_change3': np.float64,
'job_profile3': str,
'time_type3': str,
'cost_center3': str,
'effective_date4': str,
'business_process_type4': str,
'business_process_reason4': str,
'pay_rate_type4': str,
'base_pay_change4': np.float64,
'job_profile4': str,
'time_type4': str,
'cost_center4': str,
'effective_date5': str,
'business_process_type5': str,
'business_process_reason5': str,
'pay_rate_type5': str,
'base_pay_change5': np.float64,
'job_profile5': str,
'time_type5': str,
'cost_center5': str,
'effective_date6': str,
'business_process_type6': str,
'business_process_reason6': str,
'pay_rate_type6': str,
'base_pay_change6': np.float64,
'job_profile6': str,
'time_type6': str,
'cost_center6': str,
'effective_date7': str,
'business_process_type7': str,
'business_process_reason7': str,
'pay_rate_type7': str,
'base_pay_change7': np.float64,
'job_profile7': str,
'time_type7': str,
'cost_center7': str,
'effective_date8': str,
'business_process_type8': str,
'business_process_reason8': str,
'pay_rate_type8': str,
'base_pay_change8': np.float64,
'job_profile8': str,
'time_type8': str,
'cost_center8': str,
'effective_date9': str,
'business_process_type9': str,
'business_process_reason9': str,
'pay_rate_type9': str,
'base_pay_change9': np.float64,
'job_profile9': str,
'time_type9': str,
'cost_center9': str,
'effective_date10': str,
'business_process_type10': str,
'business_process_reason10': str,
'pay_rate_type10': str,
'base_pay_change10': np.float64,
'job_profile10': str,
'time_type10': str,
'cost_center10': str,
'effective_date11': str,
'business_process_type11': str,
'business_process_reason11': str,
'pay_rate_type11': str,
'base_pay_change11': np.float64,
'job_profile11': str,
'time_type11': str,
'cost_center11': str,
'effective_date12': str,
'business_process_type12': str,
'business_process_reason12': str,
'pay_rate_type12': str,
'base_pay_change12': np.float64,
'job_profile12': str,
'time_type12': str,
'cost_center12': str,
'effective_date13': str,
'business_process_type13': str,
'business_process_reason13': str,
'pay_rate_type13': str,
'base_pay_change13': np.float64,
'job_profile13': str,
'time_type13': str,
'cost_center13': str,
'effective_date14': str,
'business_process_type14': str,
'business_process_reason14': str,
'pay_rate_type14': str,
'base_pay_change14': np.float64,
'job_profile14': str,
'time_type14': str,
'cost_center14': str,
'effective_date15': str,
'business_process_type15': str,
'business_process_reason15': str,
'pay_rate_type15': str,
'base_pay_change15': np.float64,
'job_profile15': str,
'time_type15': str,
'cost_center15': str,
'effective_date16': str,
'business_process_type16': str,
'business_process_reason16': str,
'pay_rate_type16': str,
'base_pay_change16': np.float64,
'job_profile16': str,
'time_type16': str,
'cost_center16': str,
'effective_date17': str,
'business_process_type17': str,
'business_process_reason17': str,
'pay_rate_type17': str,
'base_pay_change17': np.float64,
'job_profile17': str,
'time_type17': str,
'cost_center17': str,
'effective_date18': str,
'business_process_type18': str,
'business_process_reason18': str,
'pay_rate_type18': str,
'base_pay_change18': np.float64,
'job_profile18': str,
'time_type18': str,
'cost_center18': str,
'effective_date19': str,
'business_process_type19': str,
'pay_rate_type19': str,
'base_pay_change19': np.float64,
'job_profile19': str,
'time_type19': str,
'cost_center19': str,
'2015_annual_performance_rating': np.float64,
'2016_annual_performance_rating': np.float64,
'2017_annual_performance_rating': np.float64,
'2018_annual_performance_rating': np.float64}

parse_dates = ['date_of_birth', 'original_hire_date',
'hire_date', 'effective_date1', 'effective_date2', 'effective_date3', 'effective_date4', 'effective_date5', 'effective_date6', 'effective_date7', 'effective_date8', 'effective_date9', 'effective_date10', 'effective_date11', 'effective_date12', 'effective_date13',

[4]: terminated_wd_schema = {
'department': str,
'employee_id': str,
'gender': str,
'race_ethnicity': str,
'date_of_birth': str,
'original_hire_date': str,
'hire_date': str,
'termination_date': str,
'pay_rate_type': str,
'current_base_pay': np.float64,
'job_profile_current': str,
'time_type_current': str,
'cost_center_current': str,
'effective_date1': str,
'business_process_type1': str,
'business_process_reason1': str,
'pay_rate_type1': str,
'base_pay_change1': np.float64,
<table>
<thead>
<tr>
<th>column_name</th>
<th>data_type</th>
</tr>
</thead>
<tbody>
<tr>
<td>base_pay_change7</td>
<td>np.float64</td>
</tr>
<tr>
<td>job_profile7</td>
<td>str</td>
</tr>
<tr>
<td>time_type7</td>
<td>str</td>
</tr>
<tr>
<td>cost_center7</td>
<td>str</td>
</tr>
<tr>
<td>effective_date8</td>
<td>str</td>
</tr>
<tr>
<td>business_process_type8</td>
<td>str</td>
</tr>
<tr>
<td>business_process_reason8</td>
<td>str</td>
</tr>
<tr>
<td>pay_rate_type8</td>
<td>str</td>
</tr>
<tr>
<td>base_pay_change8</td>
<td>np.float64</td>
</tr>
<tr>
<td>job_profile8</td>
<td>str</td>
</tr>
<tr>
<td>time_type8</td>
<td>str</td>
</tr>
<tr>
<td>cost_center8</td>
<td>str</td>
</tr>
<tr>
<td>effective_date9</td>
<td>str</td>
</tr>
<tr>
<td>business_process_type9</td>
<td>str</td>
</tr>
<tr>
<td>business_process_reason9</td>
<td>str</td>
</tr>
<tr>
<td>pay_rate_type9</td>
<td>str</td>
</tr>
<tr>
<td>base_pay_change9</td>
<td>np.float64</td>
</tr>
<tr>
<td>job_profile9</td>
<td>str</td>
</tr>
<tr>
<td>time_type9</td>
<td>str</td>
</tr>
<tr>
<td>cost_center9</td>
<td>str</td>
</tr>
<tr>
<td>effective_date10</td>
<td>str</td>
</tr>
<tr>
<td>business_process_type10</td>
<td>str</td>
</tr>
<tr>
<td>business_process_reason10</td>
<td>str</td>
</tr>
<tr>
<td>pay_rate_type10</td>
<td>str</td>
</tr>
<tr>
<td>base_pay_change10</td>
<td>np.float64</td>
</tr>
<tr>
<td>job_profile10</td>
<td>str</td>
</tr>
<tr>
<td>time_type10</td>
<td>str</td>
</tr>
<tr>
<td>cost_center10</td>
<td>str</td>
</tr>
<tr>
<td>effective_date11</td>
<td>str</td>
</tr>
<tr>
<td>business_process_type11</td>
<td>str</td>
</tr>
<tr>
<td>business_process_reason11</td>
<td>str</td>
</tr>
<tr>
<td>pay_rate_type11</td>
<td>str</td>
</tr>
<tr>
<td>base_pay_change11</td>
<td>np.float64</td>
</tr>
<tr>
<td>job_profile11</td>
<td>str</td>
</tr>
<tr>
<td>time_type11</td>
<td>str</td>
</tr>
<tr>
<td>cost_center11</td>
<td>str</td>
</tr>
<tr>
<td>effective_date12</td>
<td>str</td>
</tr>
<tr>
<td>business_process_type12</td>
<td>str</td>
</tr>
<tr>
<td>business_process_reason12</td>
<td>str</td>
</tr>
<tr>
<td>pay_rate_type12</td>
<td>str</td>
</tr>
<tr>
<td>base_pay_change12</td>
<td>np.float64</td>
</tr>
<tr>
<td>job_profile12</td>
<td>str</td>
</tr>
<tr>
<td>time_type12</td>
<td>str</td>
</tr>
<tr>
<td>cost_center12</td>
<td>str</td>
</tr>
<tr>
<td>effective_date13</td>
<td>str</td>
</tr>
<tr>
<td>business_process_type13</td>
<td>str</td>
</tr>
<tr>
<td>business_process_reason13</td>
<td>str</td>
</tr>
</tbody>
</table>
'pay_rate_type13': str,
'base_pay_change13': np.float64,
'job_profile13': str,
'time_type13': str,
'cost_center13': str,
'effective_date14': str,
'business_process_type14': str,
'business_process_reason14': str,
'pay_rate_type14': str,
'job_profile14': str,
'time_type14': str,
'cost_center14': str,
'2015_annual_performance_rating': np.float64,
'2016_annual_performance_rating': np.float64,
'2017_annual_performance_rating': np.float64,
'2018_annual_performance_rating': np.float64
)

parse_dates2 = ['date_of_birth', 'original_hire_date',
'hire_date', 'termination_date', 'effective_date1', 'effective_date2', 'effective_date3', 'effective_date4', 'effective_date5', 'effective_date6', 'effective_date7', 'effective_date8', 'effective_date9', 'effective_date10', 'effective_date11', 'effective_date12', 'effective_date13', 'effective_date14']

[5]:
df = pd.read_csv(CSVPATH.joinpath('active_wd.csv'),
dtype=active_wd_schema,
parse_dates=parse_dates2)
df2 = pd.read_csv(CSVPATH.joinpath('terminated_wd.csv'),
parse_dates=parse_dates2)

### 1.2 Add fields for analysis

[6]:
date_received = np.datetime64('2019-07-02')

df['age'] = (date_received - df['date_of_birth']).astype('<m8[Y]')
df['years_of_service'] = (date_received - df['hire_date']).astype('<m8[Y]')
df2['age'] = (date_received - df2['date_of_birth']).astype('<m8[Y]')
df2['years_of_service'] = (date_received - df2['hire_date']).astype('<m8[Y]')

### 1.2.1 Add field for 5-year age groups

[7]:
bins= [0,25,30,35,40,45,50,55,60,65,100]
labels = ['<25', '25-29', '30-34', '35-39', '40-44',
'45-49', '50-54', '55-59', '60-64', '65+']
df['age_group_5'] = pd.cut(df['age'], bins=bins, labels=labels, right=False)
df2['age_group_5'] = pd.cut(df2['age'], bins=bins, labels=labels, right=False)
1.2.2 Add field for 10-year age groups

```python
bins = [0, 25, 35, 45, 55, 65, 100]
labels = ['<25', '25-34', '35-44', '45-54', '55-64', '65+']
df['age_group_10'] = pd.cut(df['age'], bins=bins, labels=labels, right=False)
df2['age_group_10'] = pd.cut(df2['age'], bins=bins, labels=labels, right=False)
```

1.2.3 Add field for years-of-service groups

```python
bins = [0, 1, 3, 6, 11, 16, 21, 26, 100]
labels = ['0', '1-2', '3-5', '6-10', '11-15', '16-20', '21-25', '25+']
df['years_of_service_grouped'] = pd.cut(df['years_of_service'], bins=bins, labels=labels, right=False)
df2['years_of_service_grouped'] = pd.cut(df2['years_of_service'], bins=bins, labels=labels, right=False)
```

1.2.4 Group departments

```python
def dept(row):
    NEWS_DEPTS = ['News', 'Editorial', 'News Service and Syndicate']
    COMMERCIAL_DEPTS = ['Client Solutions', 'Circulation', 'Finance', 'Marketing', 'WP News',
                         'Media Services', 'Production', 'Public Relations', 'Administration',
                         'Product', 'Audience Development and Insights', 'Customer Care and Logistics', 'Legal', 'Washington Post Live']
    if row['department'] in NEWS_DEPTS:
        return 'News'
    elif row['department'] in COMMERCIAL_DEPTS:
        return 'Commercial'
    else:
        return 'Unknown'

df['dept'] = df.apply(lambda row: dept(row), axis=1)
df2['dept'] = df2.apply(lambda row: dept(row), axis=1)
```

1.2.5 Group desks

```python
def desk(row):
    OPERATIONS = ['110000 News Operations', '110001 News Digital Operations']
    AUDIENCE = ['Audience Development and Engagement']
    AUDIO = ['110620 News Audio']
    DESIGN = ['110604 Presentation Design', '110605 Presentation']
    EMERGING = ['110664 News National Apps', '110665 News The Lily', '110666 News Snapchat', '110667 News By The Way']
```
FINANCIAL = ['113210 Economy and Business']
FOREIGN = ['114000 Foreign Administration', '114095 News Foreign
Brazil', '114100 Foreign Latam', '114220 News Foreign Istanbul', '114235
Foreign Western Europe', '114300 News Foreign West Africa', '114415 Foreign
Hong Kong', '114405 Foreign Beijing Bureau', '114405 Foreign Mexico
Bureau', '114000 Foreign Beirut Bureau', '114400 Foreign India Bureau
Foreign Tokyo Bureau', '114205 Foreign Islamabad Bureau', '114305 Foreign
Nairobi Bureau', '114240 Foreign Rome Bureau', '114230 Foreign Moscow Bureau', '114225 Foreign Cairo Bureau', '114215 Foreign
Berlin Bureau']
GRAPHICS = ['110603 Presentation Graphics']
INVESTIGATIVE = ['110450 Investigative']
LOCAL = ['112300 Local Politics and Government']
MULTI = ['110601 Multiplatform Desk']
NATIONAL = ['110500 Magazine', '113200 National Politics and
Government', '113205 National Security', '113215 News National Health &
Science', '113220 National Enterprise', '113235 National America', '113240 News
National Environment']
RESEARCH = ['110006 News Content & Research']
LOGISTICS = ['110455 News Logistics']
OUTLOOK = ['110410 Book World', '110460 Outlook']
POLLING = ['110475 Polling']
SPORTS = ['110015 Sports Main']
STYLE = ['110300 Style', '110435 Food', '110485 Travel', '110495 Local
Living', '110505 Weekend']
UNIVERSAL = ['110600 Universal Desk']
VIDEO = ['110652 News Video - General']
OTHER = ['110663 Wake Up Report']
EDITORIAL = ['115000 Editorial Administration']

if row['cost_center_current'] in OPERATIONS:
    return 'Operations'
elif row['cost_center_current'] in AUDIENCE:
    return 'Audience Development and Engagement'
elif row['cost_center_current'] in AUDIO:
    return 'Audio'
elif row['cost_center_current'] in DESIGN:
    return 'Design'
elif row['cost_center_current'] in EMERGING:
    return 'Emerging News Products'
elif row['cost_center_current'] in FINANCIAL:
    return 'Financial'
elif row['cost_center_current'] in FOREIGN:
    return 'Foreign'
elif row['cost_center_current'] in GRAPHICS:
    return 'Graphics'
elif row['cost_center_current'] in LOCAL:
    return 'Local'
elif row['cost_center_current'] in MULTI:
    return 'Multiplatform'
elif row['cost_center_current'] in NATIONAL:
    return 'National'
elif row['cost_center_current'] in RESEARCH:
    return 'News Content and Research'
elif row['cost_center_current'] in LOGISTICS:
    return 'News Logistics'
elif row['cost_center_current'] in OUTLOOK:
    return 'Outlook'
elif row['cost_center_current'] in POLLING:
    return 'Polling'
elif row['cost_center_current'] in SPORTS:
    return 'Sports'
elif row['cost_center_current'] in STYLE:
    return 'Style'
elif row['cost_center_current'] in UNIVERSAL:
    return 'Universal Desk'
elif row['cost_center_current'] in VIDEO:
    return 'Video'
elif row['cost_center_current'] in OTHER:
    return 'Other'
elif row['cost_center_current'] in EDITORIAL:
    return 'Editorial'
else:
    return 'non-newsroom'

df['desk'] = df.apply(lambda row: desk(row), axis=1)
df2['desk'] = df2.apply(lambda row: desk(row), axis=1)

1.2.6 Group desks by median salary ranges

[12]: def tier(row):
    TIER1 = ['National', 'Foreign', 'Financial', 'Investigative']
    TIER2 = ['Style', 'Local', 'Graphics', 'Universal Desk', 'Outlook', 'Editorial']
    TIER3 = ['Audio', 'Polling', 'Design', 'Operations', 'Multiplatform', 'Video', 'Audience Development and Engagement']
    TIER4 = ['News Logistics', 'News Content and Research', 'Emerging News Products', 'Other']
    if row['desk'] in TIER1:
        return 'Tier 1'
elif row['desk'] in TIER2:
        return 'Tier 2'
elif row['desk'] in TIER3:
return 'Tier 3'
elif row['desk'] in TIER4:
    return 'Tier 4'
else:
    return 'other'

df['tier'] = df.apply(lambda row: tier(row), axis=1)
df2['tier'] = df2.apply(lambda row: tier(row), axis=1)

1.2.7 Group race and ethnicity

def race_groups(row):
    WHITE = ['White (United States of America)']
    NONWHITE = ['Black or African American (United States of America)', 'Asian (United States of America)', 'Hispanic or Latino (United States of America)', 'Two or More Races (United States of America)', 'American Indian or Alaska Native (United States of America)', 'Native Hawaiian or Other Pacific Islander (United States of America)']
    if row['race_ethnicity'] in WHITE:
        return 'white'
    elif row['race_ethnicity'] in NONWHITE:
        return 'person of color'
    else:
        return 'unknown'

df['race_grouping'] = df.apply(lambda row: race_groups(row), axis=1)
df2['race_grouping'] = df2.apply(lambda row: race_groups(row), axis=1)

1.2.8 Employee pay change grouping

def reason_for_change1 =
    df[['business_process_reason1','base_pay_change1','effective_date1','pay_rate_type1','gender']]
    rename(columns={'business_process_reason1':
          'business_process_reason','base_pay_change1':
          'base_pay_change','effective_date1':
          'effective_date1':'effective_date','pay_rate_type1':
          'pay_rate_type'})
reason_for_change2 =
    df[['business_process_reason2','base_pay_change2','effective_date2','pay_rate_type2','gender']]
    rename(columns={'business_process_reason2':
          'business_process_reason','base_pay_change2':
          'base_pay_change','effective_date2':
          'effective_date2':'effective_date','pay_rate_type2':
          'pay_rate_type'})
reason_for_change3 =
  df[['business_process_reason3', 'base_pay_change3', 'effective_date3', 'pay_rate_type3', 'gender']]
  rename(columns={'business_process_reason3':
    'business_process_reason', 'base_pay_change3':
    'base_pay_change', 'effective_date3': 'effective_date', 'pay_rate_type3':
    'pay_rate_type'})
reason_for_change4 =
  df[['business_process_reason4', 'base_pay_change4', 'effective_date4', 'pay_rate_type4', 'gender']]
  rename(columns={'business_process_reason4':
    'business_process_reason', 'base_pay_change4':
    'base_pay_change', 'effective_date4': 'effective_date', 'pay_rate_type4':
    'pay_rate_type'})
reason_for_change5 =
  df[['business_process_reason5', 'base_pay_change5', 'effective_date5', 'pay_rate_type5', 'gender']]
  rename(columns={'business_process_reason5':
    'business_process_reason', 'base_pay_change5':
    'base_pay_change', 'effective_date5': 'effective_date', 'pay_rate_type5':
    'pay_rate_type'})
reason_for_change6 =
  df[['business_process_reason6', 'base_pay_change6', 'effective_date6', 'pay_rate_type6', 'gender']]
  rename(columns={'business_process_reason6':
    'business_process_reason', 'base_pay_change6':
    'base_pay_change', 'effective_date6': 'effective_date', 'pay_rate_type6':
    'pay_rate_type'})
reason_for_change7 =
  df[['business_process_reason7', 'base_pay_change7', 'effective_date7', 'pay_rate_type7', 'gender']]
  rename(columns={'business_process_reason7':
    'business_process_reason', 'base_pay_change7':
    'base_pay_change', 'effective_date7': 'effective_date', 'pay_rate_type7':
    'pay_rate_type'})
reason_for_change8 =
  df[['business_process_reason8', 'base_pay_change8', 'effective_date8', 'pay_rate_type8', 'gender']]
  rename(columns={'business_process_reason8':
    'business_process_reason', 'base_pay_change8':
    'base_pay_change', 'effective_date8': 'effective_date', 'pay_rate_type8':
    'pay_rate_type'})
reason_for_change9 =
  df[['business_process_reason9', 'base_pay_change9', 'effective_date9', 'pay_rate_type9', 'gender']]
  rename(columns={'business_process_reason9':
    'business_process_reason', 'base_pay_change9':
    'base_pay_change', 'effective_date9': 'effective_date', 'pay_rate_type9':
    'pay_rate_type'})
reason_for_change10 =
- df[['business_process_reason10', 'base_pay_change10', 'effective_date10', 'pay_rate_type10', 'gender10'],
- rename(columns={
- 'business_process_reason10':
- 'business_process_reason',
- 'base_pay_change10':
- 'base_pay_change',
- 'effective_date10': 'effective_date',
- 'pay_rate_type10':
- 'pay_rate_type'})

reason_for_change11 =
- df[['business_process_reason11', 'base_pay_change11', 'effective_date11', 'pay_rate_type11', 'gender11'],
- rename(columns={
- 'business_process_reason11':
- 'business_process_reason',
- 'base_pay_change11':
- 'base_pay_change',
- 'effective_date11': 'effective_date',
- 'pay_rate_type11':
- 'pay_rate_type'})

reason_for_change12 =
- df[['business_process_reason12', 'base_pay_change12', 'effective_date12', 'pay_rate_type12', 'gender12'],
- rename(columns={
- 'business_process_reason12':
- 'business_process_reason',
- 'base_pay_change12':
- 'base_pay_change',
- 'effective_date12': 'effective_date',
- 'pay_rate_type12':
- 'pay_rate_type'})

reason_for_change13 =
- df[['business_process_reason13', 'base_pay_change13', 'effective_date13', 'pay_rate_type13', 'gender13'],
- rename(columns={
- 'business_process_reason13':
- 'business_process_reason',
- 'base_pay_change13':
- 'base_pay_change',
- 'effective_date13': 'effective_date',
- 'pay_rate_type13':
- 'pay_rate_type'})

reason_for_change14 =
- df[['business_process_reason14', 'base_pay_change14', 'effective_date14', 'pay_rate_type14', 'gender14'],
- rename(columns={
- 'business_process_reason14':
- 'business_process_reason',
- 'base_pay_change14':
- 'base_pay_change',
- 'effective_date14': 'effective_date',
- 'pay_rate_type14':
- 'pay_rate_type'})

reason_for_change15 =
- df[['business_process_reason15', 'base_pay_change15', 'effective_date15', 'pay_rate_type15', 'gender15'],
- rename(columns={
- 'business_process_reason15':
- 'business_process_reason',
- 'base_pay_change15':
- 'base_pay_change',
- 'effective_date15': 'effective_date',
- 'pay_rate_type15':
- 'pay_rate_type'})

reason_for_change16 =
- df[['business_process_reason16', 'base_pay_change16', 'effective_date16', 'pay_rate_type16', 'gender16'],
- rename(columns={
- 'business_process_reason16':
- 'business_process_reason',
- 'base_pay_change16':
- 'base_pay_change',
- 'effective_date16': 'effective_date',
- 'pay_rate_type16':
- 'pay_rate_type'}}
reason_for_change17 =
    df[['business_process_reason17', 'base_pay_change17', 'effective_date17', 'pay_rate_type17', 'gender']
    rename(columns={'business_process_reason17':
        'business_process_reason', 'base_pay_change17':
        'base_pay_change', 'effective_date17': 'effective_date', 'pay_rate_type17':
        'pay_rate_type'})

reason_for_change18 =
    df[['business_process_reason18', 'base_pay_change18', 'effective_date18', 'pay_rate_type18', 'gender']
    rename(columns={'business_process_reason18':
        'business_process_reason', 'base_pay_change18':
        'base_pay_change', 'effective_date18': 'effective_date', 'pay_rate_type18':
        'pay_rate_type'})

reason_for_change19 =
    df2[['business_process_reason1', 'base_pay_change1', 'effective_date1', 'pay_rate_type1', 'gender']
    rename(columns={'business_process_reason1':
        'business_process_reason', 'base_pay_change1':
        'base_pay_change', 'effective_date1': 'effective_date', 'pay_rate_type1':
        'pay_rate_type'})

reason_for_change20 =
    df2[['business_process_reason2', 'base_pay_change2', 'effective_date2', 'pay_rate_type2', 'gender']
    rename(columns={'business_process_reason2':
        'business_process_reason', 'base_pay_change2':
        'base_pay_change', 'effective_date2': 'effective_date', 'pay_rate_type2':
        'pay_rate_type'})

reason_for_change21 =
    df2[['business_process_reason3', 'base_pay_change3', 'effective_date3', 'pay_rate_type3', 'gender']
    rename(columns={'business_process_reason3':
        'business_process_reason', 'base_pay_change3':
        'base_pay_change', 'effective_date3': 'effective_date', 'pay_rate_type3':
        'pay_rate_type'})

reason_for_change22 =
    df2[['business_process_reason4', 'base_pay_change4', 'effective_date4', 'pay_rate_type4', 'gender']
    rename(columns={'business_process_reason4':
        'business_process_reason', 'base_pay_change4':
        'base_pay_change', 'effective_date4': 'effective_date', 'pay_rate_type4':
        'pay_rate_type'})

reason_for_change23 =
    df2[['business_process_reason5', 'base_pay_change5', 'effective_date5', 'pay_rate_type5', 'gender']
    rename(columns={'business_process_reason5':
        'business_process_reason', 'base_pay_change5':
        'base_pay_change', 'effective_date5': 'effective_date', 'pay_rate_type5':
        'pay_rate_type'})
reason_for_change24 =
- df2[['business_process_reason', 'base_pay_change', 'effective_date6', 'pay_rate_type', 'gender']]
- rename(columns={'business_process_reason':
    'business_process_reason', 'base_pay_change':
    'base_pay_change', 'effective_date6': 'effective_date', 'pay_rate_type':
    'pay_rate_type'})
reason_for_change25 =
- df2[['business_process_reason', 'base_pay_change7', 'effective_date7', 'pay_rate_type', 'gender']]
- rename(columns={'business_process_reason':
    'business_process_reason', 'base_pay_change7':
    'base_pay_change', 'effective_date7': 'effective_date', 'pay_rate_type':
    'pay_rate_type'})
reason_for_change26 =
- df2[['business_process_reason', 'base_pay_change8', 'effective_date8', 'pay_rate_type', 'gender']]
- rename(columns={'business_process_reason':
    'business_process_reason', 'base_pay_change8':
    'base_pay_change', 'effective_date8': 'effective_date', 'pay_rate_type':
    'pay_rate_type'})
reason_for_change27 =
- df2[['business_process_reason', 'base_pay_change9', 'effective_date9', 'pay_rate_type', 'gender']]
- rename(columns={'business_process_reason':
    'business_process_reason', 'base_pay_change9':
    'base_pay_change', 'effective_date9': 'effective_date', 'pay_rate_type':
    'pay_rate_type'})
reason_for_change28 =
- df2[['business_process_reason', 'base_pay_change10', 'effective_date10', 'pay_rate_type', 'gender']]
- rename(columns={'business_process_reason':
    'business_process_reason', 'base_pay_change10':
    'base_pay_change', 'effective_date10': 'effective_date', 'pay_rate_type10':
    'pay_rate_type'})
reason_for_change29 =
- df2[['business_process_reason', 'base_pay_change11', 'effective_date11', 'pay_rate_type11', 'gender']]
- rename(columns={'business_process_reason':
    'business_process_reason', 'base_pay_change11':
    'base_pay_change', 'effective_date11': 'effective_date', 'pay_rate_type11':
    'pay_rate_type'})
reason_for_change30 =
- df2[['business_process_reason', 'base_pay_change12', 'effective_date12', 'pay_rate_type12', 'gender']]
- rename(columns={'business_process_reason':
    'business_process_reason', 'base_pay_change12':
    'base_pay_change', 'effective_date12': 'effective_date', 'pay_rate_type12':
    'pay_rate_type'})

rename(columns={'business_process_reason13': 'business_process_reason', 'base_pay_change13': 'base_pay_change', 'effective_date13': 'effective_date', 'pay_rate_type13': 'pay_rate_type'})

reason_for_change1 = pd.DataFrame(reason_for_change1)
reason_for_change2 = pd.DataFrame(reason_for_change2)
reason_for_change3 = pd.DataFrame(reason_for_change3)
reason_for_change4 = pd.DataFrame(reason_for_change4)
reason_for_change5 = pd.DataFrame(reason_for_change5)
reason_for_change6 = pd.DataFrame(reason_for_change6)
reason_for_change7 = pd.DataFrame(reason_for_change7)
reason_for_change8 = pd.DataFrame(reason_for_change8)
reason_for_change9 = pd.DataFrame(reason_for_change9)
reason_for_change10 = pd.DataFrame(reason_for_change10)
reason_for_change11 = pd.DataFrame(reason_for_change11)
reason_for_change12 = pd.DataFrame(reason_for_change12)
reason_for_change13 = pd.DataFrame(reason_for_change13)
reason_for_change14 = pd.DataFrame(reason_for_change14)
reason_for_change15 = pd.DataFrame(reason_for_change15)
reason_for_change16 = pd.DataFrame(reason_for_change16)
reason_for_change17 = pd.DataFrame(reason_for_change17)
reason_for_change18 = pd.DataFrame(reason_for_change18)
reason_for_change19 = pd.DataFrame(reason_for_change19)
reason_for_change20 = pd.DataFrame(reason_for_change20)
reason_for_change21 = pd.DataFrame(reason_for_change21)
reason_for_change22 = pd.DataFrame(reason_for_change22)
reason_for_change23 = pd.DataFrame(reason_for_change23)
reason_for_change24 = pd.DataFrame(reason_for_change24)
reason_for_change25 = pd.DataFrame(reason_for_change25)
reason_for_change26 = pd.DataFrame(reason_for_change26)
reason_for_change27 = pd.DataFrame(reason_for_change27)
reason_for_change28 = pd.DataFrame(reason_for_change28)
reason_for_change29 = pd.DataFrame(reason_for_change29)
reason_for_change30 = pd.DataFrame(reason_for_change30)
reason_for_change31 = pd.DataFrame(reason_for_change31)

reason_for_change_combined = pd.concat([reason_for_change1, reason_for_change2, reason_for_change3, reason_for_change4, reason_for_change5, reason_for_change6, reason_for_change7, reason_for_change8, reason_for_change9, reason_for_change10, reason_for_change11, reason_for_change12, reason_for_change13, reason_for_change14, reason_for_change15, reason_for_change16, reason_for_change17, reason_for_change18, reason_for_change19, reason_for_change20, reason_for_change21, reason_for_change22, reason_for_change23, reason_for_change24, reason_for_change25, reason_for_change26, reason_for_change27, reason_for_change28, reason_for_change29, reason_for_change30, reason_for_change31])

1.2.9 Employee performance evaluation grouping

[15]:

fifteen1 = df[['2015_annual_performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept']].
rename(columns={'2015_annual_performance_rating': 'performance_rating'})
1.2.10 Create departmental data frames

```python
sixteen1 = df[['2016_annual_performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept']]. rename(columns={'2016_annual_performance_rating': 'performance_rating'})
sixteen2 = df[['2016_annual_performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept']]. rename(columns={'2016_annual_performance_rating': 'performance_rating'})
seventeen1 = df[['2017_annual_performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept']]. rename(columns={'2017_annual_performance_rating': 'performance_rating'})
seventeen2 = df[['2017_annual_performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept']]. rename(columns={'2017_annual_performance_rating': 'performance_rating'})
eighteen1 = df[['2018_annual_performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept']]. rename(columns={'2018_annual_performance_rating': 'performance_rating'})
eighteen2 = df[['2018_annual_performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept']]. rename(columns={'2018_annual_performance_rating': 'performance_rating'})
fifteen1 = pd.DataFrame(fifteen1)
fifteen2 = pd.DataFrame(fifteen2)
sixteen1 = pd.DataFrame(sixteen1)
sixteen2 = pd.DataFrame(sixteen2)
seventeen1 = pd.DataFrame(seventeen1)
seventeen2 = pd.DataFrame(seventeen2)
eighteen1 = pd.DataFrame(eighteen1)
eighteen2 = pd.DataFrame(eighteen2)
ratings_combined = pd.concat([fifteen1, fifteen2, sixteen1, sixteen2, seventeen1, seventeen2, eighteen1, eighteen2])
```

1.2.10 Create departmental data frames

```python
news_salaried = df[(df['dept'] == 'News') & (df['pay_rate_type'] == 'Salaried')]
news_hourly = df[(df['dept'] == 'News') & (df['pay_rate_type'] == 'Hourly')]
commercial_salaried = df[(df['dept'] == 'Commercial') & (df['pay_rate_type'] == 'Salaried')]
commercial_hourly = df[(df['dept'] == 'Commercial') & (df['pay_rate_type'] == 'Hourly')]
```

```python
news_salaried2 = df2[(df2['dept'] == 'News') & (df2['pay_rate_type'] == 'Salaried')]
news_hourly2 = df2[(df2['dept'] == 'News') & (df2['pay_rate_type'] == 'Hourly')]
```
commercial_salaried2 = df2[(df2['dept'] == 'Commercial') & (df2['pay_rate_type'] == 'Salaried')]
commercial_hourly2 = df2[(df2['dept'] == 'Commercial') & (df2['pay_rate_type'] == 'Hourly')]

1.3 Suppress Results

1.3.1 Suppress results where there are less than five employees

```python
[17]:
df['count'] = 1
df2['count'] = 1

def suppress(results):
    results.columns = results.columns.get_level_values(1)
    return results[results['count_nonzero'] >= 5]
```

1.3.2 Suppress results and order them by count of employees

```python
[18]:
def suppress_count(results):
    results.columns = results.columns.get_level_values(1)
    return results[results['count_nonzero'] >= 5].sort_values('count_nonzero', ascending=False)
```

1.3.3 Suppress results and order them by median salary of employees

```python
[19]:
def suppress_median(results):
    results.columns = results.columns.get_level_values(1)
    return results[results['count_nonzero'] >= 5].sort_values('median', ascending=False)
```

1.4 Summary Analysis

1.4.1 Employee counts

```python
[20]:
current_employee_count = df.shape[0]
terminated_employee_count = df2.shape[0]

print('Total employees in data: ' + str(current_employee_count + terminated_employee_count))
print('Current employees: ' + str(current_employee_count))
print('Terminated employees: ' + str(terminated_employee_count))
```

Total employees in data: 1489
Current employees: 950
Terminated employees: 539
current_salaried_employee_count = df[df['pay_rate_type'] == 'Salaried'].shape[0]
terminated_salaried_employee_count = df2[df2['pay_rate_type'] == 'Salaried'].shape[0]

print('Total salaried employees in data: ' + str(current_salaried_employee_count + terminated_salaried_employee_count))
print('Current salaried employees: ' + str(current_salaried_employee_count))
print('Terminated salaried employees: ' + str(terminated_salaried_employee_count))

Total salaried employees in data: 989
Current salaried employees: 707
Terminated salaried employees: 282

current_hourly_employee_count = df[df['pay_rate_type'] == 'Hourly'].shape[0]
terminated_hourly_employee_count = df2[df2['pay_rate_type'] == 'Hourly'].shape[0]

print('Total hourly employees in data: ' + str(current_hourly_employee_count + terminated_hourly_employee_count))
print('Current hourly employees: ' + str(current_hourly_employee_count))
print('Terminated hourly employees: ' + str(terminated_hourly_employee_count))

Total hourly employees in data: 500
Current hourly employees: 243
Terminated hourly employees: 257

1.4.2 Salary information

current_mean_salary = df[df['pay_rate_type'] == 'Salaried']['current_base_pay'].mean()
current_median_salary = df[df['pay_rate_type'] == 'Salaried']['current_base_pay'].median()

print('The mean yearly pay for current salaried employees is $' + str(current_mean_salary) + '.')
print('The median yearly pay for current salaried employees is $' + str(current_median_salary) + '.')

The mean yearly pay for current salaried employees is $112382.98421499293.
The median yearly pay for current salaried employees is $99903.95.

current_mean_hourly = df[df['pay_rate_type'] == 'Hourly']['current_base_pay'].mean()
current_median_hourly = df[df['pay_rate_type'] == 'Hourly']['current_base_pay'].median()
The mean rate for current hourly employees at The Washington Post is $30.197119341563788.
The median rate for current hourly employees at The Washington Post is $29.23.

1.4.3 Employee gender

```
[25]: current_employee_gender = df.groupby(['gender']).agg({'current_base_pay': [np.count_nonzero]})
suppress(current_employee_gender)

[25]:

current_nonzero
gender
Female  507.00
Male  443.00

[26]: terminated_employee_gender = df2.groupby(['gender']).agg({'current_base_pay': [np.count_nonzero]})
suppress(terminated_employee_gender)

[26]:

current_nonzero
gender
Female  291.00
Male  246.00

[27]: current_median_salary_gender = df[df['pay_rate_type'] == 'Salaried'].
    .groupby(['gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_salary_gender)

[27]:
current_nonzero  median
gender
Female  370.00  91815.82
Male  337.00  109928.29

[28]: current_median_hourly_gender = df[df['pay_rate_type'] == 'Hourly'].
    .groupby(['gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_hourly_gender)

[28]:
current_nonzero  median
gender
Female  137.00  30.77
Male  106.00  25.84

[29]: current_age_gender_salaried = df[df['pay_rate_type'] == 'Salaried'].
    .groupby(['gender'])['age'].median().sort_values(ascending=False)
current_age_gender_salaried
```
gender

Male 41.00
Female 35.00
Name: age, dtype: float64

1.4.4 Employee race and ethnicity

```python
[30]: current_employee_race_ethnicity = df.groupby(['race_ethnicity']).
    agg({'current_base_pay': [np.count_nonzero]})
suppress_count(current_employee_race_ethnicity)

[30]:

count_nonzero
race_ethnicity
White (United States of America) 612.00
Black or African American (United States of Ame... 157.00
Asian (United States of America) 77.00
Hispanic or Latino (United States of America) 45.00
Two or More Races (United States of America) 18.00
Prefer Not to Disclose (United States of America) 14.00

[31]: terminated_employee_race_ethnicity = df2.groupby(['race_ethnicity']).
    agg({'current_base_pay': [np.count_nonzero]})
suppress_count(terminated_employee_race_ethnicity)

[31]:

count_nonzero
race_ethnicity
White (United States of America) 290.00
Black or African American (United States of Ame... 162.00
Asian (United States of America) 46.00
Hispanic or Latino (United States of America) 20.00
Two or More Races (United States of America) 10.00
Prefer Not to Disclose (United States of America) 7.00

[32]: current_median_salary_race = df[df['pay_rate_type'] == 'Salaried'].
    groupby(['race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.
    median]})
suppress_median(current_median_salary_race)

[32]:

count_nonzero  median
race_ethnicity
White (United States of America) 505.00 102880.00
Black or African American (United States of Ame... 62.00 91881.24
Asian (United States of America) 69.00 90780.00
Prefer Not to Disclose (United States of America) 10.00 82140.00
Hispanic or Latino (United States of America) 33.00 82000.00
Two or More Races (United States of America) 14.00 79860.00
```
current_median_hourly_race = df[df['pay_rate_type'] == 'Hourly'].
               .groupby(['race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.
               median]})
suppress_median(current_median_hourly_race)

[33]:

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>White (United States of America)</td>
<td>107.00</td>
<td>32.71</td>
</tr>
<tr>
<td>Asian (United States of America)</td>
<td>18.00</td>
<td>27.30</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>12.00</td>
<td>25.62</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>95.00</td>
<td>25.16</td>
</tr>
</tbody>
</table>

current_age_race_salaried = df[df['pay_rate_type'] == 'Salaried'].
               .groupby(['race_ethnicity'])['age'].median().sort_values(ascending=False)
current_age_race_salaried

[34]:

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>age</th>
</tr>
</thead>
<tbody>
<tr>
<td>American Indian or Alaska Native (United States of America)</td>
<td>49.50</td>
</tr>
<tr>
<td>Native Hawaiian or Other Pacific Islander (United States of America)</td>
<td>43.00</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>41.50</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>39.00</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>37.00</td>
</tr>
<tr>
<td>Asian (United States of America)</td>
<td>33.00</td>
</tr>
<tr>
<td>Prefer Not to Disclose (United States of America)</td>
<td>31.50</td>
</tr>
<tr>
<td>Two or More Races (United States of America)</td>
<td>28.00</td>
</tr>
</tbody>
</table>

Name: age, dtype: float64

current_age_race_hourly = df[df['pay_rate_type'] == 'Hourly'].
               .groupby(['race_ethnicity'])['age'].median().sort_values(ascending=False)
current_age_race_hourly

[35]:

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>age</th>
</tr>
</thead>
<tbody>
<tr>
<td>American Indian or Alaska Native (United States of America)</td>
<td>53.50</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>47.00</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>39.00</td>
</tr>
<tr>
<td>Asian (United States of America)</td>
<td>32.00</td>
</tr>
<tr>
<td>Prefer Not to Disclose (United States of America)</td>
<td>30.00</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>29.50</td>
</tr>
<tr>
<td>Two or More Races (United States of America)</td>
<td>26.50</td>
</tr>
</tbody>
</table>

Name: age, dtype: float64

1.4.5 Employee gender x race/ethnicity

[36]:

current_employee_race_gender = df.groupby(['race_ethnicity', 'gender']).
               agg({'current_base_pay': [np.count_nonzero]})
suppress(current_employee_race_gender)

[36]:

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>gender</th>
</tr>
</thead>
<tbody>
<tr>
<td>count_nonzero</td>
<td></td>
</tr>
</tbody>
</table>
Asian (United States of America)    Female  53.00
                                Male    24.00
Black or African American (United States of America)    Female  80.00
                                    Male    77.00
Hispanic or Latino (United States of America)    Female  24.00
                                      Male    21.00
Prefer Not to Disclose (United States of America)    Female   6.00
                                     Male     8.00
Two or More Races (United States of America)    Female  12.00
                                         Male    8.00
White (United States of America)    Female  318.00
                                Male    294.00

[37]:  current_salaried_race_gender = df[df['pay_rate_type'] == 'Salaried'].
groupby(['race_ethnicity', 'gender']).agg({'current_base_pay': [np.
count_nonzero]})
suppress(current_salaried_race_gender)

[37]:  count_nonzero

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>gender</th>
<th>count_nonzero</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>11.00</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>16.00</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>Female</td>
<td>11.00</td>
</tr>
<tr>
<td>Prefer Not to Disclose (United States of America)</td>
<td>Female</td>
<td>11.00</td>
</tr>
<tr>
<td>Two or More Races (United States of America)</td>
<td>Female</td>
<td>11.00</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>Female</td>
<td>16.00</td>
</tr>
</tbody>
</table>

[38]:  current_hourly_race_gender = df[df['pay_rate_type'] == 'Hourly'].
groupby(['race_ethnicity', 'gender']).agg({'current_base_pay': [np.
count_nonzero]})
suppress(current_hourly_race_gender)

[38]:  count_nonzero

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>gender</th>
<th>count_nonzero</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>11.00</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>16.00</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>Female</td>
<td>11.00</td>
</tr>
<tr>
<td>Prefer Not to Disclose (United States of America)</td>
<td>Female</td>
<td>11.00</td>
</tr>
<tr>
<td>Two or More Races (United States of America)</td>
<td>Female</td>
<td>11.00</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>Female</td>
<td>16.00</td>
</tr>
</tbody>
</table>
```python
# Calculate current median salary by race and gender for Salaried pay rate
current_median_salary_race_gender = df[df['pay_rate_type'] == 'Salaried']
  .groupby(['race_ethnicity', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_salary_race_gender)

# Calculate current median hourly rate by race and gender for Hourly pay rate
current_median_hourly_race_gender = df[df['pay_rate_type'] == 'Hourly']
  .groupby(['race_ethnicity', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_hourly_race_gender)
```

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>gender</th>
<th>count_nonzero</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>42.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>17.00</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>31.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>31.00</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>Female</td>
<td>16.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>17.00</td>
</tr>
<tr>
<td>Prefer Not to Disclose (United States of America)</td>
<td>Female</td>
<td>5.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>5.00</td>
</tr>
<tr>
<td>Two or More Races (United States of America)</td>
<td>Female</td>
<td>9.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>5.00</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>Female</td>
<td>255.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>250.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>gender</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>91115.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>90431.45</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>87808.33</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>99931.09</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>Female</td>
<td>80250.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>90780.00</td>
</tr>
<tr>
<td>Prefer Not to Disclose (United States of America)</td>
<td>Female</td>
<td>73000.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>88280.00</td>
</tr>
<tr>
<td>Two or More Races (United States of America)</td>
<td>Female</td>
<td>75000.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>94875.00</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>Female</td>
<td>95780.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>111035.50</td>
</tr>
</tbody>
</table>
```

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>gender</th>
<th>count_nonzero</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>11.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>7.00</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>49.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>46.00</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>Female</td>
<td>8.00</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>Female</td>
<td>63.00</td>
</tr>
</tbody>
</table>
Male 44.00

median

race_ethnicity  gender
Asian (United States of America)  Female  28.30
                    Male  26.30
Black or African American (United States of America)  Female  26.82
                        Male  23.20
Hispanic or Latino (United States of America)  Female  28.17
White (United States of America)  Female  33.46
                        Male  31.00

1.4.6 Employee age

[41]: current_employee_age_5 = df.groupby(['age_group_5']).agg({'current_base_pay': np.count_nonzero})
suppress(current_employee_age_5)

[41]:
  count_nonzero
  age_group_5     
  <25             59.00
  25-29           171.00
  30-34           139.00
  35-39           125.00
  40-44           98.00
  45-49           80.00
  50-54           105.00
  55-59           84.00
  60-64           56.00
  65+             33.00

[42]: terminated_employee_age_5 = df2.groupby(['age_group_5']).
  .agg({'current_base_pay': np.count_nonzero})
suppress(terminated_employee_age_5)

[42]:
  count_nonzero
  age_group_5     
  <25             7.00
  25-29           117.00
  30-34           115.00
  35-39           56.00
  40-44           52.00
  45-49           40.00
  50-54           33.00
  55-59           42.00
  60-64           29.00
  65+             44.00
```python
current_employee_age_10 = df.groupby(['age_group_10']).agg({'current_base_pay': [np.count_nonzero]})
suppress(current_employee_age_10)

count_nonzero
age_group_10
<25  59.00
25-34 310.00
35-44 223.00
45-54 185.00
55-64 140.00
65+   33.00

terminated_employee_age_10 = df2.groupby(['age_group_10']).agg({'current_base_pay': [np.count_nonzero]})
suppress(terminated_employee_age_10)

count_nonzero
age_group_10
<25  7.00
25-34 232.00
35-44 108.00
45-54  73.00
55-64  71.00
65+   44.00

current_median_salary_age_5 = df[df['pay_rate_type'] == 'Salaried'].groupby(['age_group_5']).agg({'current_base_pay': [np.median, np.count_nonzero]})
suppress(current_median_salary_age_5)

median  count_nonzero
age_group_5
<25    64640.00  34.00
25-29  80000.00  126.00
30-34  92500.00  119.00
35-39 105301.31  104.00
40-44 125924.46  72.00
45-49  99502.50  56.00
50-54 110844.65  80.00
55-59 139716.51  61.00
60-64 113134.31  38.00
65+   153061.00  17.00

current_median_hourly_age_5 = df[df['pay_rate_type'] == 'Hourly'].groupby(['age_group_5']).agg({'current_base_pay': [np.median, np.count_nonzero]})
suppress(current_median_hourly_age_5)
```
### 1.4.7 Employee department

```python
[49]: current_employee_dept = df.groupby(['dept']).agg({'current_base_pay': [np.
  ...:    count_nonzero]})
suppress_count(current_employee_dept)
```

```python
[49]: count_nonzero
depth
```
<table>
<thead>
<tr>
<th>Department</th>
<th>Count</th>
<th>Median Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>News</td>
<td>574.00</td>
<td>104669.96</td>
</tr>
<tr>
<td>Commercial</td>
<td>133.00</td>
<td>86104.69</td>
</tr>
<tr>
<td>Editorial</td>
<td>33.00</td>
<td>105000.00</td>
</tr>
<tr>
<td>News</td>
<td>541.00</td>
<td>104559.92</td>
</tr>
<tr>
<td>Finance</td>
<td>8.00</td>
<td>90575.50</td>
</tr>
<tr>
<td>WP News Media Services</td>
<td>9.00</td>
<td>86104.69</td>
</tr>
<tr>
<td>Client Solutions</td>
<td>102.00</td>
<td>85633.86</td>
</tr>
<tr>
<td>Marketing</td>
<td>7.00</td>
<td>81196.11</td>
</tr>
<tr>
<td>Production</td>
<td>5.00</td>
<td>71665.06</td>
</tr>
<tr>
<td>News</td>
<td>96.00</td>
<td>33.05</td>
</tr>
<tr>
<td>Commercial</td>
<td>147.00</td>
<td>26.27</td>
</tr>
</tbody>
</table>
1.4.8 Employee cost center

current_employee_department_hourly = df[df['pay_rate_type'] == 'Hourly'].
    groupby(['department']).agg({'current_base_pay': [np.count_nonzero, np.
    median]})
suppress_median(current_employee_department_hourly)

current_employee_desk = df.groupby(['desk']).agg({'current_base_pay': [np.
    count_nonzero]})
suppress_count(current_employee_desk)

current_employee_cost_center = df.groupby(['cost_center_current']).
    agg({'current_base_pay': [np.count_nonzero]})
suppress_count(current_employee_cost_center)
<table>
<thead>
<tr>
<th>Code</th>
<th>Department</th>
<th>Hours</th>
</tr>
</thead>
<tbody>
<tr>
<td>110652</td>
<td>News Video - General</td>
<td>50.00</td>
</tr>
<tr>
<td>110015</td>
<td>Sports Main</td>
<td>48.00</td>
</tr>
<tr>
<td>110601</td>
<td>Multiplatform Desk</td>
<td>42.00</td>
</tr>
<tr>
<td>110300</td>
<td>Style</td>
<td>39.00</td>
</tr>
<tr>
<td>119065</td>
<td>Dispatch Operations (Night Circulation)</td>
<td>39.00</td>
</tr>
<tr>
<td>115000</td>
<td>Editorial Administration</td>
<td>38.00</td>
</tr>
<tr>
<td>113210</td>
<td>Economy and Business</td>
<td>38.00</td>
</tr>
<tr>
<td>110605</td>
<td>Presentation</td>
<td>24.00</td>
</tr>
<tr>
<td>110610</td>
<td>Audience Development and Engagement</td>
<td>23.00</td>
</tr>
<tr>
<td>117682</td>
<td>Global Sales</td>
<td>22.00</td>
</tr>
<tr>
<td>110604</td>
<td>Presentation Design</td>
<td>22.00</td>
</tr>
<tr>
<td>117694</td>
<td>Digital Ad Sales - BrandStudio</td>
<td>20.00</td>
</tr>
<tr>
<td>117693</td>
<td>Digital Ad Sales - Planning</td>
<td>19.00</td>
</tr>
<tr>
<td>113205</td>
<td>National Security</td>
<td>17.00</td>
</tr>
<tr>
<td>110600</td>
<td>Universal Desk</td>
<td>16.00</td>
</tr>
<tr>
<td>110603</td>
<td>Presentation Graphics</td>
<td>15.00</td>
</tr>
<tr>
<td>126020</td>
<td>Revenue Administration</td>
<td>14.00</td>
</tr>
<tr>
<td>113215</td>
<td>News National Health &amp; Science</td>
<td>14.00</td>
</tr>
<tr>
<td>110450</td>
<td>Investigative</td>
<td>13.00</td>
</tr>
<tr>
<td>110620</td>
<td>News Audio</td>
<td>13.00</td>
</tr>
<tr>
<td>110664</td>
<td>News National Apps</td>
<td>12.00</td>
</tr>
<tr>
<td>113235</td>
<td>National America</td>
<td>12.00</td>
</tr>
<tr>
<td>116010</td>
<td>Research</td>
<td>11.00</td>
</tr>
<tr>
<td>117720</td>
<td>Health</td>
<td>10.00</td>
</tr>
<tr>
<td>117525</td>
<td>National Retailers</td>
<td>10.00</td>
</tr>
<tr>
<td>110667</td>
<td>News By The Way</td>
<td>9.00</td>
</tr>
<tr>
<td>126060</td>
<td>Circulation Accounting</td>
<td>9.00</td>
</tr>
<tr>
<td>129300</td>
<td>WP News Media Services</td>
<td>9.00</td>
</tr>
<tr>
<td>110000</td>
<td>News Operations</td>
<td>7.00</td>
</tr>
<tr>
<td>117005</td>
<td>Creative Services</td>
<td>7.00</td>
</tr>
<tr>
<td>110666</td>
<td>News Snapchat</td>
<td>6.00</td>
</tr>
<tr>
<td>120005</td>
<td>Makeup</td>
<td>6.00</td>
</tr>
<tr>
<td>117320</td>
<td>Real Estate</td>
<td>6.00</td>
</tr>
<tr>
<td>117600</td>
<td>Leadership Executive</td>
<td>6.00</td>
</tr>
<tr>
<td>117004</td>
<td>Advertising Marketing</td>
<td>6.00</td>
</tr>
<tr>
<td>114000</td>
<td>Foreign Administration</td>
<td>6.00</td>
</tr>
<tr>
<td>110001</td>
<td>News Digital Operations</td>
<td>6.00</td>
</tr>
<tr>
<td>126010</td>
<td>General Ledger</td>
<td>6.00</td>
</tr>
<tr>
<td>110495</td>
<td>Local Living</td>
<td>6.00</td>
</tr>
<tr>
<td>129100</td>
<td>Community</td>
<td>5.00</td>
</tr>
<tr>
<td>110435</td>
<td>Food</td>
<td>5.00</td>
</tr>
<tr>
<td>110460</td>
<td>Outlook</td>
<td>5.00</td>
</tr>
<tr>
<td>117210</td>
<td>Production Creative</td>
<td>5.00</td>
</tr>
<tr>
<td>119026</td>
<td>Customer Contact Center</td>
<td>5.00</td>
</tr>
<tr>
<td>118150</td>
<td>WP Live</td>
<td>5.00</td>
</tr>
<tr>
<td>117310</td>
<td>Consumer to Consumer Team I</td>
<td>5.00</td>
</tr>
<tr>
<td>117405</td>
<td>Jobs Tactical</td>
<td>5.00</td>
</tr>
</tbody>
</table>
current_employee_desk_salary = df[df['pay_rate_type'] == 'Salaried'].
groupby(['desk']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_employee_desk_salary)

desk
National 106.00 149520.50
Foreign 25.00 135000.00
Financial 38.00 133509.94
Style 45.00 107170.81
Local 65.00 105780.00
Editorial 33.00 105000.00
Graphics 15.00 100780.00
Universal Desk 8.00 100444.28
Sports 37.00 100000.00
Outlook 6.00 99937.50
Audio 7.00 92000.00
Design 45.00 88065.25
Operations 6.00 87890.00
non-newsroom 162.00 87355.95
Multiplatform 26.00 86104.00
Video 46.00 84250.00
Emerging News Products 30.00 75000.00

cost_center_current
113205 National Security 17.00 172780.00
117682 Global Sales 21.00 164984.25
113200 National Politics and Government 55.00 145980.00
113235 National America 12.00 137123.72
113215 News National Health & Science 12.00 135594.87
113210 Economy and Business 38.00 133509.94
110450 Investigative 13.00 129780.00
117600 Leadership Executive 5.00 127500.00
113240 News National Environment 5.00 126080.00
110300 Style 36.00 115177.72
112300 Local Politics and Government 65.00 105780.00
115000 Editorial Administration 33.00 105000.00
110603 Presentation Graphics 15.00 100780.00
110600 Universal Desk 8.00 100444.28
110015 Sports Main 37.00 100000.00
117525 National Retailers 8.00 99499.70
<table>
<thead>
<tr>
<th>Cost Center</th>
<th>Current Base Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>Presentation</td>
<td>24.00 96711.85</td>
</tr>
<tr>
<td>Outlook</td>
<td>5.00 94875.00</td>
</tr>
<tr>
<td>News Audio</td>
<td>7.00 92000.00</td>
</tr>
<tr>
<td>General Ledger</td>
<td>6.00 90575.50</td>
</tr>
<tr>
<td>Health</td>
<td>10.00 87924.59</td>
</tr>
<tr>
<td>News Digital Operations</td>
<td>6.00 87890.00</td>
</tr>
<tr>
<td>WP News Media Services</td>
<td>9.00 86104.69</td>
</tr>
<tr>
<td>Multiplatform Desk</td>
<td>26.00 86104.00</td>
</tr>
<tr>
<td>Digital Ad Sales - BrandStudio</td>
<td>18.00 85000.00</td>
</tr>
<tr>
<td>News Video - General</td>
<td>46.00 84250.00</td>
</tr>
<tr>
<td>Audience Development and Engagement</td>
<td>16.00 83530.00</td>
</tr>
<tr>
<td>Research</td>
<td>7.00 81196.11</td>
</tr>
<tr>
<td>News By The Way</td>
<td>9.00 80000.00</td>
</tr>
<tr>
<td>Presentation Design</td>
<td>21.00 78641.52</td>
</tr>
<tr>
<td>News Snapchat</td>
<td>6.00 76890.00</td>
</tr>
<tr>
<td>Creative Services</td>
<td>6.00 75587.35</td>
</tr>
<tr>
<td>Makeup</td>
<td>5.00 71665.06</td>
</tr>
<tr>
<td>News National Apps</td>
<td>11.00 68780.01</td>
</tr>
<tr>
<td>Digital Ad Sales - Planning</td>
<td>19.00 68000.00</td>
</tr>
</tbody>
</table>

```
[59]: current_employee_desk_hourly = df[df['pay_rate_type'] == 'Hourly'].
    .groupby(['desk']).agg({'current_base_pay': [np.count_nonzero, np.median]})
    suppress_median(current_employee_desk_hourly)

<table>
<thead>
<tr>
<th>Desk</th>
<th>Count Nonzero</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audio</td>
<td>6.00</td>
<td>39.75</td>
</tr>
<tr>
<td>Universal Desk</td>
<td>8.00</td>
<td>38.67</td>
</tr>
<tr>
<td>Multiplatform</td>
<td>16.00</td>
<td>34.09</td>
</tr>
<tr>
<td>Editorial</td>
<td>5.00</td>
<td>32.31</td>
</tr>
<tr>
<td>National</td>
<td>12.00</td>
<td>31.74</td>
</tr>
<tr>
<td>Non-newsroom</td>
<td>154.00</td>
<td>26.57</td>
</tr>
<tr>
<td>Local</td>
<td>5.00</td>
<td>26.46</td>
</tr>
<tr>
<td>Style</td>
<td>9.00</td>
<td>21.77</td>
</tr>
<tr>
<td>Sports</td>
<td>11.00</td>
<td>20.91</td>
</tr>
<tr>
<td>Operations</td>
<td>7.00</td>
<td>15.59</td>
</tr>
</tbody>
</table>

[60]: current_employee_cost_center_hourly = df[df['pay_rate_type'] == 'Hourly'].
    .groupby(['cost_center_current']).agg({'current_base_pay': [np.count_nonzero, np.median]})
    suppress_median(current_employee_cost_center_hourly)

<table>
<thead>
<tr>
<th>Cost Center</th>
<th>Count Nonzero</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>News Audio</td>
<td>6.00</td>
<td>39.75</td>
</tr>
<tr>
<td>Universal Desk</td>
<td>8.00</td>
<td>38.67</td>
</tr>
<tr>
<td>Audience Development and Engagement</td>
<td>7.00</td>
<td>37.58</td>
</tr>
<tr>
<td>Community</td>
<td>5.00</td>
<td>35.01</td>
</tr>
<tr>
<td>Multiplatform Desk</td>
<td>16.00</td>
<td>34.09</td>
</tr>
</tbody>
</table>
```
1.4.9 Employee years of service

```python
[61]: current_employee_yos = df.groupby(['years_of_service_grouped']).
    agg({'current_base_pay': [np.count_nonzero]})
suppress(current_employee_yos)

[61]:
    count_nonzero
    years_of_service_grouped   
      0         138.00
      1-2       223.00
      3-5       195.00
      6-10      109.00
     11-15      80.00
     16-20      102.00
     21-25      46.00
     25+        57.00

[62]: terminated_employee_yos = df2.groupby(['years_of_service_grouped']).
        agg({'current_base_pay': [np.count_nonzero]})
suppress(terminated_employee_yos)

[62]:
    count_nonzero
    years_of_service_grouped   
      0         8.00
      1-2       78.00
      3-5       196.00
      6-10      119.00
     11-15      51.00
     16-20      44.00
     21-25      12.00
     25+        29.00

[63]: current_employee_yos_salary = df[df['pay_rate_type'] == 'Salaried'].
        groupby(['years_of_service_grouped']).agg({'current_base_pay': [np.
        count_nonzero, np.median]})
```
<table>
<thead>
<tr>
<th>years_of_service_grouped</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>96.00</td>
<td>85000.00</td>
</tr>
<tr>
<td>1-2</td>
<td>164.00</td>
<td>91776.89</td>
</tr>
<tr>
<td>3-5</td>
<td>172.00</td>
<td>92305.85</td>
</tr>
<tr>
<td>6-10</td>
<td>75.00</td>
<td>106602.62</td>
</tr>
<tr>
<td>11-15</td>
<td>56.00</td>
<td>107685.39</td>
</tr>
<tr>
<td>16-20</td>
<td>74.00</td>
<td>125300.67</td>
</tr>
<tr>
<td>21-25</td>
<td>32.00</td>
<td>128485.24</td>
</tr>
<tr>
<td>25+</td>
<td>38.00</td>
<td>131793.39</td>
</tr>
</tbody>
</table>

```python
current_employee_yos_hourly = df[df['pay_rate_type'] == 'Hourly'].
groupby(['years_of_service_grouped']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_employee_yos_hourly)
```

<table>
<thead>
<tr>
<th>years_of_service_grouped</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>42.00</td>
<td>27.70</td>
</tr>
<tr>
<td>1-2</td>
<td>59.00</td>
<td>31.68</td>
</tr>
<tr>
<td>3-5</td>
<td>23.00</td>
<td>27.05</td>
</tr>
<tr>
<td>6-10</td>
<td>34.00</td>
<td>29.25</td>
</tr>
<tr>
<td>11-15</td>
<td>24.00</td>
<td>32.41</td>
</tr>
<tr>
<td>16-20</td>
<td>28.00</td>
<td>27.78</td>
</tr>
<tr>
<td>21-25</td>
<td>14.00</td>
<td>31.14</td>
</tr>
<tr>
<td>25+</td>
<td>19.00</td>
<td>26.82</td>
</tr>
</tbody>
</table>

```python
current_employee_yos_gender = df.groupby(['years_of_service_grouped', 'gender']).
agg({'current_base_pay': np.count_nonzero})
suppress(current_employee_yos_gender)
```

<table>
<thead>
<tr>
<th>years_of_service_grouped</th>
<th>gender</th>
<th>count_nonzero</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Female</td>
<td>82.00</td>
</tr>
<tr>
<td>0</td>
<td>Male</td>
<td>56.00</td>
</tr>
<tr>
<td>1-2</td>
<td>Female</td>
<td>132.00</td>
</tr>
<tr>
<td>1-2</td>
<td>Male</td>
<td>91.00</td>
</tr>
<tr>
<td>3-5</td>
<td>Female</td>
<td>96.00</td>
</tr>
<tr>
<td>3-5</td>
<td>Male</td>
<td>99.00</td>
</tr>
<tr>
<td>6-10</td>
<td>Female</td>
<td>51.00</td>
</tr>
<tr>
<td>6-10</td>
<td>Male</td>
<td>58.00</td>
</tr>
<tr>
<td>11-15</td>
<td>Female</td>
<td>41.00</td>
</tr>
<tr>
<td>11-15</td>
<td>Male</td>
<td>39.00</td>
</tr>
<tr>
<td>16-20</td>
<td>Female</td>
<td>48.00</td>
</tr>
<tr>
<td>16-20</td>
<td>Male</td>
<td>54.00</td>
</tr>
<tr>
<td>21-25</td>
<td>Female</td>
<td>25.00</td>
</tr>
<tr>
<td>21-25</td>
<td>Male</td>
<td>21.00</td>
</tr>
</tbody>
</table>
25+  
|   | Female | 32.00 |
|   | Male   | 25.00 |

```python
[66]:
current_employee_yos_gender_salary = df[df['pay_rate_type'] == 'Salaried'].
    groupby(['years_of_service_grouped', 'gender']).agg({'current_base_pay': [np.
    count_nonzero, np.median]})
suppress(current_employee_yos_gender_salary)
```

<table>
<thead>
<tr>
<th>years_of_service_grouped</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Female</td>
<td>61.00</td>
<td>80000.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>35.00</td>
<td>100000.00</td>
</tr>
<tr>
<td>1-2</td>
<td>Female</td>
<td>96.00</td>
<td>85780.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>68.00</td>
<td>96737.80</td>
</tr>
<tr>
<td>3-5</td>
<td>Female</td>
<td>88.00</td>
<td>89724.74</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>84.00</td>
<td>95265.36</td>
</tr>
<tr>
<td>6-10</td>
<td>Female</td>
<td>38.00</td>
<td>99499.70</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>37.00</td>
<td>117843.50</td>
</tr>
<tr>
<td>11-15</td>
<td>Female</td>
<td>28.00</td>
<td>98141.60</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>28.00</td>
<td>126910.89</td>
</tr>
<tr>
<td>16-20</td>
<td>Female</td>
<td>31.00</td>
<td>121140.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>43.00</td>
<td>127059.40</td>
</tr>
<tr>
<td>21-25</td>
<td>Female</td>
<td>13.00</td>
<td>134780.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>19.00</td>
<td>99012.31</td>
</tr>
<tr>
<td>25+</td>
<td>Female</td>
<td>15.00</td>
<td>139831.30</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>23.00</td>
<td>127476.17</td>
</tr>
</tbody>
</table>

```python
[67]:
current_employee_yos_gender_hourly = df[df['pay_rate_type'] == 'Hourly'].
    groupby(['years_of_service_grouped', 'gender']).agg({'current_base_pay': [np.
    count_nonzero, np.median]})
suppress(current_employee_yos_gender_hourly)
```

<table>
<thead>
<tr>
<th>years_of_service_grouped</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Female</td>
<td>21.00</td>
<td>29.23</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>21.00</td>
<td>22.05</td>
</tr>
<tr>
<td>1-2</td>
<td>Female</td>
<td>36.00</td>
<td>31.92</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>23.00</td>
<td>26.04</td>
</tr>
<tr>
<td>3-5</td>
<td>Female</td>
<td>8.00</td>
<td>34.77</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>15.00</td>
<td>22.98</td>
</tr>
<tr>
<td>6-10</td>
<td>Female</td>
<td>13.00</td>
<td>30.84</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>21.00</td>
<td>25.16</td>
</tr>
<tr>
<td>11-15</td>
<td>Female</td>
<td>13.00</td>
<td>34.72</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>11.00</td>
<td>29.92</td>
</tr>
<tr>
<td>16-20</td>
<td>Female</td>
<td>17.00</td>
<td>25.08</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>11.00</td>
<td>30.21</td>
</tr>
<tr>
<td>21-25</td>
<td>Female</td>
<td>12.00</td>
<td>30.25</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>17.00</td>
<td>27.69</td>
</tr>
</tbody>
</table>

36
current_employee_yos_race = df.append(groupby(['years_of_service_grouped', 'race_ethnicity']).agg({'current_base_pay': [np.count_nonzero]}))
suppress(current_employee_yos_race)

<table>
<thead>
<tr>
<th>years_of_service_grouped</th>
<th>race_ethnicity</th>
<th>count_nonzero</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Asian (United States of America)</td>
<td>15.00</td>
</tr>
<tr>
<td>1</td>
<td>Black or African American (United States of America)</td>
<td>20.00</td>
</tr>
<tr>
<td>10</td>
<td>Hispanic or Latino (United States of America)</td>
<td>10.00</td>
</tr>
<tr>
<td>8</td>
<td>Prefer Not to Disclose (United States of America)</td>
<td>8.00</td>
</tr>
<tr>
<td>6</td>
<td>Two or More Races (United States of America)</td>
<td>6.00</td>
</tr>
<tr>
<td>77</td>
<td>White (United States of America)</td>
<td>77.00</td>
</tr>
<tr>
<td>1-2</td>
<td>Asian (United States of America)</td>
<td>20.00</td>
</tr>
<tr>
<td>20</td>
<td>Black or African American (United States of America)</td>
<td>30.00</td>
</tr>
<tr>
<td>30</td>
<td>Hispanic or Latino (United States of America)</td>
<td>12.00</td>
</tr>
<tr>
<td>12</td>
<td>Two or More Races (United States of America)</td>
<td>6.00</td>
</tr>
<tr>
<td>146</td>
<td>White (United States of America)</td>
<td>146.00</td>
</tr>
<tr>
<td>3-5</td>
<td>Asian (United States of America)</td>
<td>17.00</td>
</tr>
<tr>
<td>17</td>
<td>Black or African American (United States of America)</td>
<td>20.00</td>
</tr>
<tr>
<td>20</td>
<td>Hispanic or Latino (United States of America)</td>
<td>17.00</td>
</tr>
<tr>
<td>17</td>
<td>Two or More Races (United States of America)</td>
<td>6.00</td>
</tr>
<tr>
<td>127</td>
<td>White (United States of America)</td>
<td>127.00</td>
</tr>
<tr>
<td>6-10</td>
<td>Asian (United States of America)</td>
<td>8.00</td>
</tr>
<tr>
<td>8</td>
<td>Black or African American (United States of America)</td>
<td>21.00</td>
</tr>
<tr>
<td>71</td>
<td>White (United States of America)</td>
<td>71.00</td>
</tr>
<tr>
<td>11-15</td>
<td>Asian (United States of America)</td>
<td>5.00</td>
</tr>
<tr>
<td>Race/Ethnicity</td>
<td>1-2 years of service</td>
<td>3-5 years of service</td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>----------------------</td>
<td>----------------------</td>
</tr>
<tr>
<td>Black or African American</td>
<td></td>
<td></td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>23.00</td>
<td>70.00</td>
</tr>
<tr>
<td>Asian (United States of America)</td>
<td>6.00</td>
<td>15.00</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>5.00</td>
<td>5.00</td>
</tr>
<tr>
<td>Two or More Races (United States of America)</td>
<td>5.00</td>
<td>15.00</td>
</tr>
</tbody>
</table>

```python
[69]: current_employee_yos_race_salary = df[df['pay_rate_type'] == 'Salaried'].
  .groupby(['years_of_service_grouped', 'race_ethnicity']).
  .agg({'current_base_pay': [np.count_nonzero, np.median]})
  suppress(current_employee_yos_race_salary)

[69]: count_nonzero \
  years_of_service_grouped race_ethnicity
  0 Asian (United States of America)
  11.00 Black or African American (United States of America)
  5.00 Hispanic or Latino (United States of America)
  5.00 White (United States of America)
  65.00 Asian (United States of America)
  1-2 Black or African American (United States of America)
  16.00 Hispanic or Latino (United States of America)
  12.00 Two or More Races (United States of America)
  7.00 White (United States of America)
  5.00 Asian (United States of America)
  115.00 Black or African American (United States of America)
  3-5 Asian (United States of America)
  15.00 Black or African American (United States of America)
```
<table>
<thead>
<tr>
<th>Group</th>
<th>Median</th>
<th>Race/Ethnicity</th>
</tr>
</thead>
<tbody>
<tr>
<td>15.00</td>
<td></td>
<td>Hispanic or Latino</td>
</tr>
<tr>
<td>5.00</td>
<td></td>
<td>Two or More Races</td>
</tr>
<tr>
<td>116.00</td>
<td></td>
<td>White</td>
</tr>
<tr>
<td>6-10</td>
<td>6.00</td>
<td>Asian</td>
</tr>
<tr>
<td>5.00</td>
<td></td>
<td>Black or African American</td>
</tr>
<tr>
<td>6.00</td>
<td></td>
<td>White</td>
</tr>
<tr>
<td>56.00</td>
<td></td>
<td>Black or African American</td>
</tr>
<tr>
<td>11-15</td>
<td>7.00</td>
<td>White</td>
</tr>
<tr>
<td>43.00</td>
<td></td>
<td>Black or African American</td>
</tr>
<tr>
<td>16-20</td>
<td>10.00</td>
<td>White</td>
</tr>
<tr>
<td>58.00</td>
<td></td>
<td>Black or African American</td>
</tr>
<tr>
<td>21-25</td>
<td>23.00</td>
<td>White</td>
</tr>
<tr>
<td>25+</td>
<td>29.00</td>
<td>White</td>
</tr>
<tr>
<td>median</td>
<td></td>
<td></td>
</tr>
<tr>
<td>years_of_service_grouped</td>
<td>race_ethnicity</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>77000.00</td>
<td>Asian (United States of America)</td>
</tr>
<tr>
<td></td>
<td>87000.00</td>
<td>Black or African American</td>
</tr>
<tr>
<td></td>
<td>75000.00</td>
<td>Hispanic or Latino</td>
</tr>
<tr>
<td></td>
<td>90000.00</td>
<td>White</td>
</tr>
<tr>
<td></td>
<td>1-2</td>
<td>Asian</td>
</tr>
<tr>
<td></td>
<td>87780.00</td>
<td>Black or African American</td>
</tr>
<tr>
<td></td>
<td>89780.00</td>
<td>Hispanic or Latino</td>
</tr>
<tr>
<td></td>
<td>82000.00</td>
<td>Two or More Races</td>
</tr>
<tr>
<td></td>
<td>68000.00</td>
<td>White</td>
</tr>
<tr>
<td></td>
<td>92780.00</td>
<td>White</td>
</tr>
<tr>
<td></td>
<td>3-5</td>
<td>Asian</td>
</tr>
<tr>
<td></td>
<td>92260.14</td>
<td>Hispanic or Latino</td>
</tr>
</tbody>
</table>
Black or African American (United States of America)
94662.48
Hispanic or Latino (United States of America)
81999.88
Two or More Races (United States of America)
83340.00
White (United States of America)
93279.21
Asian (United States of America)
96944.47
Black or African American (United States of America)
89196.67
White (United States of America)
112925.50
11-15
Black or African American (United States of America)
106249.68
White (United States of America)
104397.79
21-25
White (United States of America)
134697.89
25+
White (United States of America)
134957.37

[70]: current_employee_yos_race_hourly = df[df['pay_rate_type'] == 'Hourly'].

  groupby(['years_of_service_grouped', 'race_ethnicity']).

  agg({'current_base_pay': [np.count_nonzero, np.median]})

  suppress(current_employee_yos_race_hourly)

[70]:

  years_of_service_grouped  race_ethnicity
  0  Black or African American (United States of America)
      Hispanic or Latino (United States of America)
      White (United States of America)
  12.0  Black or African American (United States of America)
      Hispanic or Latino (United States of America)
      White (United States of America)
  18.0  Black or African American (United States of America)
      Hispanic or Latino (United States of America)
      White (United States of America)
  31.0  Black or African American (United States of America)
<table>
<thead>
<tr>
<th>Race/Ethnicity</th>
<th>Median years_of_service_grouped</th>
</tr>
</thead>
<tbody>
<tr>
<td>Black or African American (United States of America)</td>
<td>25.64</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>28.21</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>29.52</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>25.75</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>21.85</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>33.46</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>21.83</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>29.23</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>24.38</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>31.92</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>30.15</td>
</tr>
</tbody>
</table>

Note: The table includes median years of service grouped by race/ethnicity.
White (United States of America) 34.05
16-20 Black or African American (United States of America) 23.99
23.99
White (United States of America) 34.87
21-25 Black or African American (United States of America) 29.74
29.74
White (United States of America) 38.93
25+ Black or African American (United States of America) 24.71
24.71
White (United States of America) 32.66

### 1.4.10 Employee performance evaluations

```python
fifteen = pd.concat([fifteen1,fifteen2])
fifteenrating_gender = fifteen.groupby(['gender'])['performance_rating'].median().sort_values(ascending=False)
```

```python
gender
Male  3.40
Female 3.40
Name: performance_rating, dtype: float64
```

```python
sixteen = pd.concat([sixteen1,sixteen2])
sixteenrating_gender = sixteen.groupby(['gender'])['performance_rating'].median().sort_values(ascending=False)
```

```python
gender
Male  3.30
Female 3.30
Name: performance_rating, dtype: float64
```

```python
seventeen = pd.concat([seventeen1,seventeen2])
seventeenrating_gender = seventeen.groupby(['gender'])['performance_rating'].median().sort_values(ascending=False)
```

```python
gender
Male  3.40
Female 3.40
Name: performance_rating, dtype: float64
```

```python
eighteen = pd.concat([eighteen1,eighteen2])
```

```python
gender
Male  3.40
Female 3.40
Name: performance_rating, dtype: float64
```
eighteenrating_gender = eighteen.groupby(['gender'])['performance_rating'].median().sort_values(ascending=False)
eighteenrating_gender

[74]:
gender
  Male  3.40
  Female  3.40
Name: performance_rating, dtype: float64

[75]:
fifteenrating_race_ethnicity = fifteen.
groupby(['race_ethnicity'])['performance_rating'].median().sort_values(ascending=False)
fifteenrating_race_ethnicity

[75]:
race_ethnicity
  American Indian or Alaska Native (United States of America) 3.50
  White (United States of America) 3.40
  Asian (United States of America) 3.40
  Two or More Races (United States of America) 3.30
  Prefer Not to Disclose (United States of America) 3.30
  Native Hawaiian or Other Pacific Islander (United States of America) 3.25
  Hispanic or Latino (United States of America) 3.20
  Black or African American (United States of America) 3.20
Name: performance_rating, dtype: float64

[76]:
sixteenrating_race_ethnicity = sixteen.
groupby(['race_ethnicity'])['performance_rating'].median().sort_values(ascending=False)
sixteenrating_race_ethnicity

[76]:
race_ethnicity
  Native Hawaiian or Other Pacific Islander (United States of America) 3.70
  White (United States of America) 3.40
  Asian (United States of America) 3.35
  Prefer Not to Disclose (United States of America) 3.30
  American Indian or Alaska Native (United States of America) 3.25
  Two or More Races (United States of America) 3.20
  Black or African American (United States of America) 3.20
  Hispanic or Latino (United States of America) 3.10
Name: performance_rating, dtype: float64

[77]:
seventeenrating_race_ethnicity = seventeen.
groupby(['race_ethnicity'])['performance_rating'].median().sort_values(ascending=False)
seventeenrating_race_ethnicity

[77]:
race_ethnicity
  American Indian or Alaska Native (United States of America) 3.55
  Native Hawaiian or Other Pacific Islander (United States of America) 3.50
  White (United States of America) 3.40

43
Prefer Not to Disclose (United States of America) 3.40
Asian (United States of America) 3.40
Two or More Races (United States of America) 3.30
Hispanic or Latino (United States of America) 3.30
Black or African American (United States of America) 3.20
Name: performance_rating, dtype: float64

[78]: eighteenrating_race_ethnicity = eighteen.
groupby(['race_ethnicity'])['performance_rating'].median().sort_values(ascending=False)
eighteenrating_race_ethnicity

[78]: race_ethnicity
American Indian or Alaska Native (United States of America) 3.55
White (United States of America) 3.50
Native Hawaiian or Other Pacific Islander (United States of America) 3.40
Asian (United States of America) 3.40
Prefer Not to Disclose (United States of America) 3.35
Two or More Races (United States of America) 3.30
Hispanic or Latino (United States of America) 3.30
Black or African American (United States of America) 3.30
Name: performance_rating, dtype: float64

[79]: fifteenrating_gender_race = fifteen.
groupby(['race_ethnicity','gender'])['performance_rating'].median().sort_values(ascending=False)
fifteenrating_gender_race

[79]: race_ethnicity
gender
White (United States of America) Male
3.50
Asian (United States of America) Male
3.50
American Indian or Alaska Native (United States of America) Female
3.50
White (United States of America) Female
3.40
Asian (United States of America) Female
3.40
American Indian or Alaska Native (United States of America) Male
3.40
Two or More Races (United States of America) Female
3.30
Prefer Not to Disclose (United States of America) Female
3.30
Native Hawaiian or Other Pacific Islander (United States of America) Male
3.30
Hispanic or Latino (United States of America) Female
3.30
<table>
<thead>
<tr>
<th>Race/Ethnicity</th>
<th>Gender</th>
<th>Performance Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>Native Hawaiian or Other Pacific Islander</td>
<td>Female</td>
<td>4.10</td>
</tr>
<tr>
<td>Hispanic or Latino</td>
<td>Male</td>
<td>3.20</td>
</tr>
<tr>
<td>Black or African American</td>
<td>Female</td>
<td>3.20</td>
</tr>
<tr>
<td>Male</td>
<td></td>
<td>3.00</td>
</tr>
<tr>
<td>Two or More Races</td>
<td>Male</td>
<td>2.75</td>
</tr>
<tr>
<td>Prefer Not to Disclose</td>
<td>Female</td>
<td>3.30</td>
</tr>
<tr>
<td>American Indian or Alaska Native</td>
<td>Male</td>
<td>3.30</td>
</tr>
<tr>
<td>Asian</td>
<td>Female</td>
<td>3.30</td>
</tr>
<tr>
<td>Prefer Not to Disclose</td>
<td>Female</td>
<td>3.30</td>
</tr>
<tr>
<td>Native Hawaiian or Other Pacific Islander</td>
<td>Male</td>
<td>3.30</td>
</tr>
<tr>
<td>Asian</td>
<td>Male</td>
<td>3.30</td>
</tr>
<tr>
<td>American Indian or Alaska Native</td>
<td>Female</td>
<td>3.30</td>
</tr>
<tr>
<td>Black or African American</td>
<td>Female</td>
<td>3.25</td>
</tr>
<tr>
<td>Two or More Races</td>
<td>Female</td>
<td>3.20</td>
</tr>
<tr>
<td>American Indian or Alaska Native</td>
<td>Male</td>
<td>3.20</td>
</tr>
<tr>
<td>Hispanic or Latino</td>
<td>Female</td>
<td>3.15</td>
</tr>
<tr>
<td>Black or African American</td>
<td>Male</td>
<td>3.15</td>
</tr>
<tr>
<td>Hispanic or Latino</td>
<td>Male</td>
<td>3.10</td>
</tr>
<tr>
<td>Race/Ethnicity</td>
<td>Gender</td>
<td>Performance Rating</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>--------</td>
<td>--------------------</td>
</tr>
<tr>
<td>Native Hawaiian or Other Pacific Islander</td>
<td>Female</td>
<td>4.00</td>
</tr>
<tr>
<td>American Indian or Alaska Native</td>
<td>Female</td>
<td>3.70</td>
</tr>
<tr>
<td>Two or More Races</td>
<td>Male</td>
<td>3.50</td>
</tr>
<tr>
<td>Prefer Not to Disclose</td>
<td>Female</td>
<td>3.50</td>
</tr>
<tr>
<td>White</td>
<td>Male</td>
<td>3.40</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>3.40</td>
</tr>
<tr>
<td>Asian</td>
<td>Female</td>
<td>3.40</td>
</tr>
<tr>
<td>Hispanic or Latino</td>
<td>Male</td>
<td>3.30</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>3.30</td>
</tr>
<tr>
<td>Asian</td>
<td>Male</td>
<td>3.30</td>
</tr>
<tr>
<td>Two or More Races</td>
<td>Female</td>
<td>3.25</td>
</tr>
<tr>
<td>Prefer Not to Disclose</td>
<td>Male</td>
<td>3.20</td>
</tr>
<tr>
<td>Black or African American</td>
<td>Female</td>
<td>3.20</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>3.10</td>
</tr>
<tr>
<td>American Indian or Alaska Native</td>
<td>Male</td>
<td>3.10</td>
</tr>
<tr>
<td>Native Hawaiian or Other Pacific Islander</td>
<td>Male</td>
<td>3.00</td>
</tr>
</tbody>
</table>
```python
eighteenrating_gender_race = eighteen.
  groupby(['race_ethnicity','gender'])['performance_rating'].median().
  sort_values(ascending=False)
eighteenrating_gender_race
```

[82]: race_ethnicity                                      gender
          American Indian or Alaska Native (United States of America) Female
                3.70
          Prefer Not to Disclose (United States of America) Female
                3.55
          White (United States of America) Male
                3.50
                3.40
          Native Hawaiian or Other Pacific Islander (United States of America) Male
                3.40
          Asian (United States of America) Male
                3.40
                3.40
          Two or More Races (United States of America) Male
                3.35
                3.30
          Prefer Not to Disclose (United States of America) Male
                3.30
          Hispanic or Latino (United States of America) Male
                3.30
                3.30
          Black or African American (United States of America) Male
                3.30
                3.30
          American Indian or Alaska Native (United States of America) Male
                3.20
          Native Hawaiian or Other Pacific Islander (United States of America) Female
          nan
Name: performance_rating, dtype: float64

### 1.4.11 Employee pay changes

```python
[83]: reason_for_change = reason_for_change_combined.
  groupby(['business_process_reason']).agg({'business_process_reason': [np.
  count_nonzero]})
suppress_count(reason_for_change)
```
### Count Nonzero for Business Process Reason by Gender

```python
business_process_reason = Request Compensation Change > Adjustment > Cont...
Merit > Performance > Annual Performance Appraisal
Data Change > Data Change > Change Job Details
Transfer > Transfer > Move to another Manager
Request Compensation Change > Adjustment > Chan...
Request Compensation Change > Adjustment > Mark...
Promotion > Promotion > Promotion
Hire Employee > New Hire > Fill Vacancy
Hire Employee > New Hire > New Position
Request Compensation Change > Adjustment > Incr...
Request Compensation Change > Adjustment > Perf...
Transfer > Transfer > Transfer between companies
Hire Employee > Rehire > Fill Vacancy
Hire Employee > New Hire > Convert Contingent
Hire Employee > New Hire > Conversion
Hire Employee > Rehire > New Position

reason_for_change_gender = reason_for_change_combined.
~groupby(['business_process_reason', 'gender']).agg({'business_process_reason': ~ [np.count_nonzero]})
suppress_count(reason_for_change_gender)
```

<table>
<thead>
<tr>
<th>business_process_reason</th>
<th>gender</th>
<th>count_nonzero</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Cont...</td>
<td>Female</td>
<td>1284</td>
</tr>
<tr>
<td>Merit &gt; Performance &gt; Annual Performance Appraisal</td>
<td>Female</td>
<td>878</td>
</tr>
<tr>
<td>Data Change &gt; Data Change &gt; Change Job Details</td>
<td>Female</td>
<td>367</td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Move to another Manager</td>
<td>Male</td>
<td>1167</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Chan...</td>
<td>Female</td>
<td>288</td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Move to another Manager</td>
<td>Female</td>
<td>234</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Mark...</td>
<td>Female</td>
<td>233</td>
</tr>
<tr>
<td>Promotion &gt; Promotion &gt; Promotion</td>
<td>Female</td>
<td>228</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Mark...</td>
<td>Male</td>
<td>151</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Chan...</td>
<td>Male</td>
<td>147</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire &gt; Fill Vacancy</td>
<td>Female</td>
<td>140</td>
</tr>
<tr>
<td>Promotion &gt; Promotion &gt; Promotion</td>
<td>Male</td>
<td>131</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire &gt; Fill Vacancy</td>
<td>Male</td>
<td>113</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire &gt; New Position</td>
<td>Female</td>
<td>109</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Incr...</td>
<td>Male</td>
<td>41</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Job...</td>
<td>Female</td>
<td>33</td>
</tr>
</tbody>
</table>

48
<table>
<thead>
<tr>
<th>Business Process Reason</th>
<th>Race/Ethnicity: United States of America</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Increment</td>
<td>Female 31</td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Transfer between departments</td>
<td>Female 30</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Job Change</td>
<td>Male 27</td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Transfer between departments</td>
<td>Male 24</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Performance Change</td>
<td>Male 21</td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Transfer between companies</td>
<td>Female 21</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Performance Change</td>
<td>Female 17</td>
</tr>
<tr>
<td>Hire Employee &gt; Rehire &gt; Fill Vacancy</td>
<td>Female 9</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire &gt; Convert Contingent</td>
<td>Female 8</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire &gt; Conversion</td>
<td>Female 7</td>
</tr>
<tr>
<td>Hire Employee &gt; Rehire &gt; Fill Vacancy</td>
<td>Male 7</td>
</tr>
<tr>
<td>Hire Employee &gt; Rehire &gt; New Position</td>
<td>Female 6</td>
</tr>
</tbody>
</table>

```
[85]: reason_for_change_race = reason_for_change_combined.
     ~groupby({'business_process_reason', 'race_ethnicity'}).
     ~agg({'business_process_reason': [np.count_nonzero]})
     suppress_count(reason_for_change_race)
```

```
   count_nonzero
business_process_reason        race_ethnicity
Request Compensation Change > Adjustment > Cont... White (United States of America) 1556
Merit > Performance > Annual Performance Appraisal White (United States of America) 1109
Request Compensation Change > Adjustment > Cont... Black or African American (United States of America) 508
Data Change > Data Change > Change Job Details White (United States of America) 432
Merit > Performance > Annual Performance Appraisal Black or African American (United States of America) 347
Transfer > Transfer > Move to another Manager White (United States of America) 288
Request Compensation Change > Adjustment > Chan... White (United States of America) 266
Request Compensation Change > Adjustment > Mark... White (United States of America) 255
Promotion > Promotion > Promotion White (United States of America) 213
Request Compensation Change > Adjustment > Cont... Asian (United States of America) 195
Transfer > Transfer > Move to another Manager Black or African American (United States of America) 168
Merit > Performance > Annual Performance Appraisal Asian (United States of America) 142
Hire Employee > New Hire > Fill Vacancy White (United States of America) 133
Hire Employee > New Hire > New Position White (United States of America) 122
```
Request Compensation Change > Adjustment > Cont... Hispanic or Latino (United States of America) 97
Request Compensation Change > Adjustment > Chan... Black or African American (United States of America) 85
Data Change > Data Change > Change Job Details Black or African American (United States of America) 83
Asian (United States of America) 76
Promotion > Promotion > Promotion Black or African American (United States of America) 74
Merit > Performance > Annual Performance Appraisal Hispanic or Latino (United States of America) 65
Hire Employee > New Hire > Fill Vacancy Black or African American (United States of America) 59
Request Compensation Change > Adjustment > Incr... White (United States of America) 55
Request Compensation Change > Adjustment > Mark... Black or African American (United States of America) 43
Transfer > Transfer > Transfer between departments White (United States of America) 40
Request Compensation Change > Adjustment > Mark... Asian (United States of America) 39
Request Compensation Change > Adjustment > Job ... White (United States of America) 39
Data Change > Data Change > Change Job Details Hispanic or Latino (United States of America) 35
Promotion > Promotion > Promotion Asian (United States of America) 34
Transfer > Transfer > Move to another Manager Asian (United States of America) 33
Request Compensation Change > Adjustment > Chan... Asian (United States of America) 30
...
...
Hire Employee > New Hire > Fill Vacancy Hispanic or Latino (United States of America) 18
Promotion > Promotion > Promotion Hispanic or Latino (United States of America) 17
Request Compensation Change > Adjustment > Chan... Hispanic or Latino (United States of America) 16
Transfer > Transfer > Transfer between companies White (United States of America) 12
Merit > Performance > Annual Performance Appraisal Two or More Races (United States of America) 12
Request Compensation Change > Adjustment > Cont... American Indian or Alaska Native (United States of America) 11
Hire Employee > New Hire > Fill Vacancy Prefer Not to Disclose

50
<table>
<thead>
<tr>
<th>Event Type</th>
<th>Subtype</th>
<th>Race/Ethnicity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Promotion</td>
<td>Promotion</td>
<td>Two or More Races (United States of America)</td>
</tr>
<tr>
<td>Promotion</td>
<td>Promotion</td>
<td>Two or More Races (United States of America)</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire</td>
<td>Fill Vacancy</td>
<td>Two or More Races (United States of America)</td>
</tr>
<tr>
<td>Request Compensation Change</td>
<td>Adjustment</td>
<td>Black or African American (United States of America)</td>
</tr>
<tr>
<td>Request Compensation Change</td>
<td>Adjustment</td>
<td>Perf... Black or African American (United States of America)</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire</td>
<td>Conversion</td>
<td>White (United States of America)</td>
</tr>
<tr>
<td>Request Compensation Change</td>
<td>Adjustment</td>
<td>Mark... Two or More Races (United States of America)</td>
</tr>
<tr>
<td>Merit</td>
<td>Performance</td>
<td>Annual Performance Appraisal American Indian or Alaska Native (United States)</td>
</tr>
<tr>
<td>Request Compensation Change</td>
<td>Adjustment</td>
<td>Job ... Asian (United States of America)</td>
</tr>
<tr>
<td>Merit</td>
<td>Performance</td>
<td>Annual Performance Appraisal Prefer Not to Disclose (United States of America)</td>
</tr>
<tr>
<td>Request Compensation Change</td>
<td>Adjustment</td>
<td>Cont... Prefer Not to Disclose (United States of America)</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire</td>
<td>Convert Contingent</td>
<td>White (United States of America)</td>
</tr>
<tr>
<td>Request Compensation Change</td>
<td>Adjustment</td>
<td>Chan... Prefer Not to Disclose (United States of America)</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire</td>
<td>New Position</td>
<td>Hispanic or Latino (United States of America)</td>
</tr>
<tr>
<td>Request Compensation Change</td>
<td>Adjustment</td>
<td>Incr... Black or African American (United States of America)</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire</td>
<td>New Position</td>
<td>Two or More Races (United States of America)</td>
</tr>
<tr>
<td>Hire Employee &gt; Rehire</td>
<td>Fill Vacancy</td>
<td>White (United States of America)</td>
</tr>
<tr>
<td>Request Compensation Change</td>
<td>Adjustment</td>
<td>Cont... Native Hawaiian or Other Pacific Islander (United States of America)</td>
</tr>
<tr>
<td>Transfer</td>
<td>Transfer</td>
<td>Transfer between departments Black or African American (United States of America) Asian (United States of America)</td>
</tr>
<tr>
<td>Hire Employee &gt; Rehire</td>
<td>Fill Vacancy</td>
<td>Black or African American (United States of America)</td>
</tr>
<tr>
<td>Transfer</td>
<td>Transfer</td>
<td>Transfer between companies Prefer Not to Disclose (United States of America)</td>
</tr>
<tr>
<td>Data Change</td>
<td>Data Change</td>
<td>Change Job Details Two or More Races (United States of America)</td>
</tr>
<tr>
<td>Request Compensation Change</td>
<td>Adjustment</td>
<td>Job ... Black or African American (United States of America)</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire</td>
<td>New Position</td>
<td>Prefer Not to Disclose (United States of America)</td>
</tr>
</tbody>
</table>
```python
reason_for_change_race_gender = reason_for_change_combined.

groupby(['business_process_reason', 'race_ethnicity', 'gender']).
agg({'business_process_reason': [np.count_nonzero]})
suppress_count(reason_for_change_race_gender)
```

<table>
<thead>
<tr>
<th>business_process_reason</th>
<th>race_ethnicity</th>
<th>count_nonzero</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Cont...</td>
<td>White (United States of America)</td>
<td>Female</td>
</tr>
<tr>
<td>Merit &gt; Performance &gt; Annual Performance Appraisal</td>
<td>Male</td>
<td>762</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Cont...</td>
<td>Black or African American (United States of America)</td>
<td>Female</td>
</tr>
<tr>
<td>Data Change &gt; Data Change &gt; Change Job Details</td>
<td>Male</td>
<td>233</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Chan...</td>
<td>White (United States of America)</td>
<td>Female</td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Move to another Manager</td>
<td>Male</td>
<td>207</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Mark...</td>
<td>White (United States of America)</td>
<td>Female</td>
</tr>
<tr>
<td>Promotion &gt; Promotion &gt; Promotion</td>
<td>Female</td>
<td>178</td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Move to another Manager</td>
<td>Male</td>
<td>164</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Mark...</td>
<td>White (United States of America)</td>
<td>Female</td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Move to another Manager</td>
<td>Female</td>
<td>137</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Cont...</td>
<td>Asian (United States of America)</td>
<td>Female</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Mark...</td>
<td>White (United States of America)</td>
<td>Male</td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Move to another Manager</td>
<td>Male</td>
<td>109</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Chan...</td>
<td>White (United States of America)</td>
<td>Male</td>
</tr>
<tr>
<td>Merit &gt; Performance &gt; Annual Performance Appraisal</td>
<td>Asian (United States of America)</td>
<td>Female</td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Move to another Manager</td>
<td>Black or African American (United States of America)</td>
<td>Female</td>
</tr>
</tbody>
</table>
(United States of America) Female 76
Promotion > Promotion > Promotion White (United States of America) Male 76
Request Compensation Change > Adjustment > Cont... Asian (United States of America) Male 75
Hire Employee > New Hire > Fill Vacancy White (United States of America) Female 74
Hire Employee > New Hire > New Position White (United States of America) Female 67
Hire Employee > New Hire > Fill Vacancy White (United States of America) Male 59
Request Compensation Change > Adjustment > Chan... Black or African American (United States of America) Female 58
Data Change > Data Change > Change Job Details Black or African American (United States of America) Female 57
Merit > Performance > Annual Performance Appraisal Asian (United States of America) Male 56
Hire Employee > New Hire > New Position White (United States of America) Male 55
...
...
Hire Employee > New Hire > Fill Vacancy Hispanic or Latino (United States of America) Female 13
Transfer > Transfer > Transfer between companies White (United States of America) Female 12
Hire Employee > New Hire > New Position Black or African American (United States of America) Male 12
Request Compensation Change > Adjustment > Mark... Hispanic or Latino (United States of America) Male 12
Request Compensation Change > Adjustment > Perf... White (United States of America) Female 11
Request Compensation Change > Adjustment > Chan... Asian (United States of America) Male 11
Hire Employee > New Hire > New Position Black or African American (United States of America) Female 11
Request Compensation Change > Adjustment > Mark... Hispanic or Latino (United States of America) Female 10
Promotion > Promotion > Promotion Two or More Races (United States of America) Female 9
Merit > Performance > Annual Performance Appraisal Two or More Races (United States of America) Female 9
Request Compensation Change > Adjustment > Cont... American Indian or Alaska Native (United States of America) Female 9
Request Compensation Change > Adjustment > Mark... Asian (United States of America) Male 9
Request Compensation Change > Adjustment > Cont... Two or More Races (United States of America) Male 8
<table>
<thead>
<tr>
<th>Action Sequence</th>
<th>Event Type</th>
<th>Race/Ethnicity</th>
<th>Gender</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Change...</td>
<td>Hispanic or Latino</td>
<td>Male</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Move to another Manager</td>
<td>Asian</td>
<td>Female</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Change...</td>
<td>Hispanic or Latino</td>
<td>Female</td>
<td>8</td>
<td></td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Mark...</td>
<td>Two or More Races</td>
<td>Female</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Move to another Manager</td>
<td>Hispanic or Latino</td>
<td>Female</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>Merit &gt; Performance &gt; Annual Performance Appraisal</td>
<td>American Indian or Alaska Native</td>
<td>Female</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Change...</td>
<td>Prefer Not to Disclose</td>
<td>Female</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire &gt; Fill Vacancy</td>
<td>Two or More Races</td>
<td>Male</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Transfer between companies</td>
<td>Prefer Not to Disclose</td>
<td>Female</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire &gt; Fill Vacancy</td>
<td>Prefer Not to Disclose</td>
<td>Female</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Transfer between departments</td>
<td>Asian</td>
<td>Male</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Merit &gt; Performance &gt; Annual Performance Appraisal</td>
<td>Prefer Not to Disclose</td>
<td>Female</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Perf...</td>
<td>Black or African American</td>
<td>Male</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire &gt; Conversion</td>
<td>White</td>
<td>Female</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire &gt; Fill Vacancy</td>
<td>Asian</td>
<td>Male</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>(United States of America)</td>
<td>(United States of America)</td>
<td>Male</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>(United States of America)</td>
<td>(United States of America)</td>
<td>Male</td>
<td>5</td>
<td></td>
</tr>
</tbody>
</table>

[93 rows x 1 columns]

### 1.5 News

#### 1.5.1 Gender

```python
[87]: current_news_gender_salaried = news_salaried.groupby(['gender']).agg({'current_base_pay': [np.count_nonzero]})
suppress(current_news_gender_salaried)
```

```python
[87]: count_nonzero
gender
```
<table>
<thead>
<tr>
<th></th>
<th>Female</th>
<th>Male</th>
</tr>
</thead>
<tbody>
<tr>
<td>2016</td>
<td>284.00</td>
<td>290.00</td>
</tr>
</tbody>
</table>

**88:**
current_news_gender_hourly = news_hourly.groupby(['gender']).
→ agg({'current_base_pay': [np.count_nonzero]})
suppress(current_news_gender_hourly)

current_news_gender_hourly

<table>
<thead>
<tr>
<th></th>
<th>count_nonzero</th>
</tr>
</thead>
<tbody>
<tr>
<td>gender</td>
<td></td>
</tr>
<tr>
<td>Female</td>
<td>63.00</td>
</tr>
<tr>
<td>Male</td>
<td>33.00</td>
</tr>
</tbody>
</table>

**89:**
current_news_gender_salaried_median = news_salaried.groupby(['gender']).
→ agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_news_gender_salaried_median)

current_news_gender_salaried_median

<table>
<thead>
<tr>
<th></th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>gender</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Female</td>
<td>284.00 95595.02</td>
<td>32.75</td>
</tr>
<tr>
<td>Male</td>
<td>290.00 116064.57</td>
<td>33.33</td>
</tr>
</tbody>
</table>

**90:**
current_news_gender_hourly_median = news_hourly.groupby(['gender']).
→ agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_news_gender_hourly_median)

current_news_gender_hourly_median

<table>
<thead>
<tr>
<th></th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>gender</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Female</td>
<td>63.00 32.75</td>
<td></td>
</tr>
<tr>
<td>Male</td>
<td>33.00 33.33</td>
<td></td>
</tr>
</tbody>
</table>

**91:**
current_news_gender_age_salaried = news_salaried.groupby(['gender'])['age'].
→ median().sort_values(ascending=False)
current_news_gender_age_salaried

<table>
<thead>
<tr>
<th></th>
<th>gender</th>
</tr>
</thead>
<tbody>
<tr>
<td>Male</td>
<td>41.00</td>
</tr>
<tr>
<td>Female</td>
<td>35.00</td>
</tr>
</tbody>
</table>

**92:**
current_news_gender_age_hourly = news_hourly.groupby(['gender'])['age'].
→ median().sort_values(ascending=False)
current_news_gender_age_hourly

<table>
<thead>
<tr>
<th></th>
<th>gender</th>
</tr>
</thead>
<tbody>
<tr>
<td>Male</td>
<td>36.00</td>
</tr>
<tr>
<td>Female</td>
<td>31.00</td>
</tr>
</tbody>
</table>

**93:**
current_news_gender_age_5_salary = news_salaried.
→ groupby(['age_group_5', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_news_gender_age_5_salary)
```python
[93]:

  age_group_5  gender  count_nonzero  median
<25  Female  19.00  64280.00
      Male  5.00  72000.00
25-29 Female  60.00  80000.00
       Male  31.00  85500.00
30-34 Female  57.00  87000.00
       Male  46.00  97827.86
35-39 Female  38.00  98891.57
       Male  48.00  116030.00
40-44 Female  22.00  133200.02
       Male  41.00  125000.00
45-49 Female  20.00  117294.59
       Male  23.00  99725.00
50-54 Female  29.00  108864.49
       Male  41.00  126280.47
55-59 Female  22.00  145654.99
       Male  29.00  147780.00
60-64 Female  12.00  129324.85
       Male  16.00  156259.68
65+  Female   5.00  157095.42
       Male  10.00  156259.68

[94]:

current_news_gender_age_5_hourly = news_hourly.
      .groupby(['age_group_5', 'gender']).agg({'current_base_pay': [np.
      count_nonzero, np.median]})
suppress(current_news_gender_age_5_hourly)

[94]:

  age_group_5  gender  count_nonzero  median
<25  Female  12.00  31.38
      Male  6.00  20.96
25-29 Female  17.00  31.17
       Male  7.00  33.73
30-34 Male   7.00  33.73
35-39 Female  5.00  31.92
40-44 Female  5.00  41.43
45-49 Female  6.00  48.55
50-54 Female  5.00  38.93
55-59 Male   5.00  34.89

[95]:

  current_news_gender_age_10_salary = news_salaried.
      .groupby(['age_group_10', 'gender']).agg({'current_base_pay': [np.
      count_nonzero, np.median]})
suppress(current_news_gender_age_10_salary)

[95]:

  age_group_10 gender  count_nonzero  median
<25  Female  19.00  64280.00
      Male  5.00  72000.00
```
<table>
<thead>
<tr>
<th>Age Group</th>
<th>Gender</th>
<th>Current Base Pay</th>
<th>Hourly Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>25-34</td>
<td>Female</td>
<td>117.00</td>
<td>83146.67</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>77.00</td>
<td>92500.00</td>
</tr>
<tr>
<td>35-44</td>
<td>Female</td>
<td>60.00</td>
<td>105691.31</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>89.00</td>
<td>118785.00</td>
</tr>
<tr>
<td>45-54</td>
<td>Female</td>
<td>49.00</td>
<td>108864.49</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>64.00</td>
<td>117981.79</td>
</tr>
<tr>
<td>55-64</td>
<td>Female</td>
<td>34.00</td>
<td>140423.62</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>45.00</td>
<td>146541.57</td>
</tr>
<tr>
<td>65+</td>
<td>Female</td>
<td>5.00</td>
<td>157095.42</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>10.00</td>
<td>156259.68</td>
</tr>
</tbody>
</table>

```python
[96]:
current_news_gender_age_10_hourly = news_hourly.
    → groupby(['age_group_10', 'gender']).agg({'current_base_pay': [np.
          → count_nonzero, np.median]})
    suppress(current_news_gender_age_10_hourly)

[96]:
count_nonzero  median
age_group_10  gender
<25          Female | 12.00 | 31.38
25-34        Female | 21.00 | 31.17
              Male   | 13.00 | 30.77
35-44        Female | 10.00 | 33.12
              Male   |  7.00 | 35.90
45-54        Female | 11.00 | 41.38
              Male   |  7.00 | 33.41
55-64        Female |  5.00 | 42.14
              Male   |  7.00 | 33.41

[97]:
current_news_gender_salaried_under_40 = news_salaried[news_salaried['age'] < 40].
    → groupby(['gender']).agg({'current_base_pay': [np.count_nonzero, np.
              → median]})
    suppress(current_news_gender_salaried_under_40)

[97]:
count_nonzero  median
gender
Female | 174.00 | 84030.00
Male   | 130.00 | 95890.00

[98]:
current_news_gender_salaried_over_40 = news_salaried[news_salaried['age'] > 39].
    → groupby(['gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
    suppress(current_news_gender_salaried_over_40)

[98]:
count_nonzero  median
gender
Female | 110.00 | 126000.00
Male   | 160.00 | 127764.51

[99]:
current_news_gender_hourly_under_40 = news_hourly[news_hourly['age'] < 40].
    → groupby(['gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
    suppress(current_news_gender_hourly_under_40)
1.5.2 Race and ethnicity

```python
[101]: current_news_race_salaried = news_salaried.groupby(['race_ethnicity']).
    .agg({'current_base_pay': [np.count_nonzero]})
suppress_count(current_news_race_salaried)

[102]: current_news_race_hourly = news_hourly.groupby(['race_ethnicity']).
    .agg({'current_base_pay': [np.count_nonzero]})
suppress_count(current_news_race_hourly)

[103]: current_news_race_group_salaried = news_salaried.groupby(['race_grouping']).
    .agg({'current_base_pay': [np.count_nonzero]})
suppress_count(current_news_race_group_salaried)
```

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>count_nonzero</th>
</tr>
</thead>
<tbody>
<tr>
<td>White (United States of America)</td>
<td>406.00</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>48.00</td>
</tr>
<tr>
<td>Asian (United States of America)</td>
<td>46.00</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>28.00</td>
</tr>
<tr>
<td>Two or More Races (United States of America)</td>
<td>14.00</td>
</tr>
<tr>
<td>Prefer Not to Disclose (United States of America)</td>
<td>8.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>race_grouping</th>
<th>count_nonzero</th>
</tr>
</thead>
<tbody>
<tr>
<td>white</td>
<td>64.00</td>
</tr>
<tr>
<td>person of color</td>
<td>139.00</td>
</tr>
<tr>
<td>unknown</td>
<td>11.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>race_grouping</th>
<th>count_nonzero</th>
</tr>
</thead>
<tbody>
<tr>
<td>white</td>
<td>406.00</td>
</tr>
<tr>
<td>person of color</td>
<td>139.00</td>
</tr>
<tr>
<td>unknown</td>
<td>29.00</td>
</tr>
</tbody>
</table>
current_news_race_group_hourly = news_hourly.groupby(['race_grouping']).agg({'current_base_pay': [np.count_nonzero]})
suppress_count(current_news_race_group_hourly)

count_nonzero
race_grouping
white 	64.00
person of color 	30.00

current_news_race_median_salaried = news_salaried.groupby(['race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_news_race_median_salaried)

count_nonzero median
race_ethnicity
White (United States of America) 	406.00 106212.10
Black or African American (United States of A... 	48.00 97276.46
Asian (United States of America) 	46.00 95205.02
Hispanic or Latino (United States of America) 	28.00 82890.00
Prefer Not to Disclose (United States of America) 	8.00 82140.00
Two or More Races (United States of America) 	14.00 79860.00

current_news_race_median_hourly = news_hourly.groupby(['race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_news_race_median_hourly)

count_nonzero median
race_ethnicity
White (United States of America) 	64.00 33.59
Asian (United States of America) 	11.00 31.68
Black or African American (United States of A... 	13.00 29.37

current_news_race_group_median_salaried = news_salaried.groupby(['race_grouping']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_news_race_group_median_salaried)

count_nonzero median
race_grouping
unknown 	29.00 134780.00
white 	406.00 106212.10
person of color 	139.00 92080.00

current_news_race_group_median_hourly = news_hourly.groupby(['race_grouping']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_news_race_group_median_hourly)

count_nonzero median
race_grouping
white 	64.00 33.59
person of color 	30.00 30.07
current_news_race_age_salaried = news_salaried.
groupby(['race_ethnicity'])['age'].median().sort_values(ascending=False)

race_ethnicity
American Indian or Alaska Native (United States of America) 49.50
Native Hawaiian or Other Pacific Islander (United States of America) 43.00
White (United States of America) 40.00
Black or African American (United States of America) 39.50
Hispanic or Latino (United States of America) 37.00
Asian (United States of America) 33.00
Prefer Not to Disclose (United States of America) 30.50
Two or More Races (United States of America) 28.00
Name: age, dtype: float64

current_news_race_age_hourly = news_hourly.groupby(['race_ethnicity'])['age'].median().sort_values(ascending=False)

current_news_race_age_hourly

race_ethnicity
American Indian or Alaska Native (United States of America) 69.00
White (United States of America) 39.50
Asian (United States of America) 36.00
Black or African American (United States of America) 28.00
Hispanic or Latino (United States of America) 26.00
Prefer Not to Disclose (United States of America) 23.00
Two or More Races (United States of America) 22.50
Name: age, dtype: float64

current_news_race_age_5_salary = news_salaried.
groupby(['age_group_5','race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_news_race_age_5_salary)

count_nonzero

<table>
<thead>
<tr>
<th>age_group_5</th>
<th>race_ethnicity</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>Asian (United States of America)</td>
<td>5.00</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>12.00</td>
</tr>
<tr>
<td>25-29</td>
<td>Asian (United States of America)</td>
<td>11.00</td>
</tr>
<tr>
<td></td>
<td>Black or African American (United States of America)</td>
<td>6.00</td>
</tr>
<tr>
<td></td>
<td>Two or More Races (United States of America)</td>
<td>6.00</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>59.00</td>
</tr>
<tr>
<td>30-34</td>
<td>Asian (United States of America)</td>
<td>10.00</td>
</tr>
<tr>
<td></td>
<td>Black or African American (United States of America)</td>
<td>9.00</td>
</tr>
<tr>
<td></td>
<td>Hispanic or Latino (United States of America)</td>
<td>6.00</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>66.00</td>
</tr>
<tr>
<td>35-39</td>
<td>Asian (United States of America)</td>
<td>7.00</td>
</tr>
<tr>
<td></td>
<td>Black or African American (United States of America)</td>
<td>7.00</td>
</tr>
<tr>
<td></td>
<td>Hispanic or Latino (United States of America)</td>
<td>7.00</td>
</tr>
</tbody>
</table>
| Age Group | Race/Ethnicity                                      | Median  
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>White (United States of America)</td>
<td>61.00</td>
</tr>
<tr>
<td>25-29</td>
<td>Asian (United States of America)</td>
<td>65780.00</td>
</tr>
<tr>
<td></td>
<td>Black or African American (United States of America)</td>
<td>81000.00</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>81540.00</td>
</tr>
<tr>
<td></td>
<td>Two or More Races (United States of America)</td>
<td>79560.00</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>81756.58</td>
</tr>
<tr>
<td>30-34</td>
<td>Asian (United States of America)</td>
<td>95780.00</td>
</tr>
<tr>
<td></td>
<td>Black or African American (United States of America)</td>
<td>88132.61</td>
</tr>
<tr>
<td></td>
<td>Hispanic or Latino (United States of America)</td>
<td>80596.26</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>92460.00</td>
</tr>
<tr>
<td>35-39</td>
<td>Asian (United States of America)</td>
<td>115000.00</td>
</tr>
<tr>
<td></td>
<td>Black or African American (United States of America)</td>
<td>96147.48</td>
</tr>
<tr>
<td></td>
<td>Hispanic or Latino (United States of America)</td>
<td>79618.25</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>105780.00</td>
</tr>
<tr>
<td>40-44</td>
<td>Black or African American (United States of America)</td>
<td>122610.00</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>126080.00</td>
</tr>
<tr>
<td>45-49</td>
<td>White (United States of America)</td>
<td>104522.64</td>
</tr>
<tr>
<td>50-54</td>
<td>Asian (United States of America)</td>
<td>103150.00</td>
</tr>
<tr>
<td></td>
<td>Black or African American (United States of America)</td>
<td>106932.24</td>
</tr>
<tr>
<td></td>
<td>Hispanic or Latino (United States of America)</td>
<td>126764.81</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>120481.79</td>
</tr>
<tr>
<td>55-59</td>
<td>White (United States of America)</td>
<td>147780.00</td>
</tr>
<tr>
<td>60-64</td>
<td>White (United States of America)</td>
<td>122780.00</td>
</tr>
<tr>
<td>65+</td>
<td>White (United States of America)</td>
<td>159300.00</td>
</tr>
</tbody>
</table>

```python
### current_news_race_age_5_hourly = news_hourly.
```
<table>
<thead>
<tr>
<th>Age Group</th>
<th>Race/Ethnicity</th>
<th>Median</th>
<th>30-34</th>
<th>White (United States of America)</th>
<th>5.00</th>
<th>White (United States of America)</th>
<th>9.00</th>
<th>45-49</th>
<th>White (United States of America)</th>
<th>6.00</th>
<th>60-64</th>
<th>White (United States of America)</th>
<th>5.00</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>White (United States of America)</td>
<td>11.00</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25-29</td>
<td>Asian (United States of America)</td>
<td>18.50</td>
<td></td>
<td>White (United States of America)</td>
<td>30.77</td>
<td>White (United States of America)</td>
<td>30.15</td>
<td>50-54</td>
<td>White (United States of America)</td>
<td>38.93</td>
<td>60-64</td>
<td>White (United States of America)</td>
<td>38.82</td>
</tr>
<tr>
<td>30-34</td>
<td>Black or African American</td>
<td>33.73</td>
<td></td>
<td>White (United States of America)</td>
<td>33.37</td>
<td>White (United States of America)</td>
<td>34.72</td>
<td>55-59</td>
<td>White (United States of America)</td>
<td>33.93</td>
<td></td>
<td>White (United States of America)</td>
<td>38.82</td>
</tr>
<tr>
<td>35-39</td>
<td>Black or African American</td>
<td>34.72</td>
<td></td>
<td>White (United States of America)</td>
<td>41.43</td>
<td>White (United States of America)</td>
<td>48.55</td>
<td>60-64</td>
<td>White (United States of America)</td>
<td>33.93</td>
<td></td>
<td>White (United States of America)</td>
<td>38.82</td>
</tr>
<tr>
<td>40-44</td>
<td>Hispanic or Latino</td>
<td>41.43</td>
<td></td>
<td>White (United States of America)</td>
<td>48.55</td>
<td>White (United States of America)</td>
<td>48.55</td>
<td></td>
<td>White (United States of America)</td>
<td>38.82</td>
<td></td>
<td>White (United States of America)</td>
<td>38.82</td>
</tr>
<tr>
<td>45-49</td>
<td>Hispanic or Latino</td>
<td>48.55</td>
<td></td>
<td>White (United States of America)</td>
<td>48.55</td>
<td>White (United States of America)</td>
<td>48.55</td>
<td></td>
<td>White (United States of America)</td>
<td>38.82</td>
<td></td>
<td>White (United States of America)</td>
<td>38.82</td>
</tr>
<tr>
<td>50-54</td>
<td>Black or African American</td>
<td>48.55</td>
<td></td>
<td>White (United States of America)</td>
<td>48.55</td>
<td>White (United States of America)</td>
<td>48.55</td>
<td></td>
<td>White (United States of America)</td>
<td>38.82</td>
<td></td>
<td>White (United States of America)</td>
<td>38.82</td>
</tr>
<tr>
<td>55-59</td>
<td>Black or African American</td>
<td>5.00</td>
<td></td>
<td>White (United States of America)</td>
<td>5.00</td>
<td>White (United States of America)</td>
<td>5.00</td>
<td></td>
<td>White (United States of America)</td>
<td>5.00</td>
<td></td>
<td>White (United States of America)</td>
<td>5.00</td>
</tr>
</tbody>
</table>

```python
[113]: current_news_race_age_10_salary = news_salaried.
    groupby(['age_group_10', 'race_ethnicity']).agg({'
current_base_pay': [np.
count_nonzero, np.median]})
    suppress(current_news_race_age_10_salary)

[113]:
count_nonzero
```
<table>
<thead>
<tr>
<th>Age Group</th>
<th>Race / Ethnicity</th>
<th>Median Current Base Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>65+</td>
<td>White (United States of America)</td>
<td>159300.00</td>
</tr>
</tbody>
</table>

**Script**

```python
65+
| White (United States of America) | 13.00 |

age_group_10 race_ethnicity
<25
| Asian (United States of America)  | 65780.00 |
| White (United States of America)  | 65140.00 |

25-34
| Black or African American (United States of America) | 87000.00 |
| Hispanic or Latino (United States of America)       | 81249.94 |
| Prefer Not to Disclose (United States of America)   | 78500.00 |
| Two or More Races (United States of America)        | 76380.00 |
| White (United States of America)                    | 86000.00 |

35-44
| Asian (United States of America)                   | 108324.02 |
| Black or African American (United States of America)| 118530.00 |
| Hispanic or Latino (United States of America)      | 90390.04  |
| White (United States of America)                    | 115258.47 |

45-54
| Black or African American (United States of America)| 123541.95 |
| Hispanic or Latino (United States of America)       | 126672.40 |
| White (United States of America)                    | 116687.17 |

55-64
| Black or African American (United States of America)| 123541.95 |
| Hispanic or Latino (United States of America)       | 126672.40 |
| White (United States of America)                    | 140051.84 |

65+
| White (United States of America)                    | 159300.00 |

[114]:

```python
current_news_race_age_10_hourly = news_hourly.
  ...groupby(['age_group_10', 'race_ethnicity']).agg({'current_base_pay': [np.
  ...count_nonzero, np.median]})
suppress(current_news_race_age_10_hourly)
```

[114]:

```python
count_nonzero
  \n  age_group_10 race_ethnicity
  <25  White (United States of America)  7.00
  25-34 Black or African American (United States of America)  8.00
  35-44 White (United States of America)  20.00
  35-44 White (United States of America)  12.00
  45-54 White (United States of America)  11.00
  55-64 White (United States of America)  11.00

median

age_group_10 race_ethnicity
<25  White (United States of America)  18.50
25-34 Black or African American (United States of America)  30.15
  White (United States of America)  31.26
35-44 White (United States of America)  35.31
45-54 White (United States of America)  41.38
55-64 White (United States of America)  34.89
```

63
```python
[115]: current_news_race_group_age_5_salary = news_salaried.
groupby(['age_group_5', 'race_grouping']).agg({'current_base_pay': [np.
count_nonzero, np.median]})
suppress(current_news_race_group_age_5_salary)
```

```python
[115]:
count_nonzero  median
age_group_5  race_grouping
<25  person of color  11.00  63780.00
     white  12.00  65140.00
25-29  person of color  27.00  80000.00
       unknown  5.00  88280.00
      white  59.00  81756.58
30-34  person of color  28.00  86982.54
       unknown  9.00  108000.00
      white  66.00  92640.00
35-39  person of color  23.00  99238.50
      white  61.00  105780.00
40-44  person of color  15.00  108324.02
       unknown  5.00  145500.00
      white  43.00  126080.00
45-49  person of color  6.00  84937.50
      white  36.00  104522.64
50-54  person of color  20.00  109396.39
      white  48.00  120481.79
55-59  person of color  6.00  131686.62
      white  43.00  147780.00
60-64  white  25.00  122780.00
65+    white  13.00  159300.00
```

```python
[116]: current_news_race_group_age_5_hourly = news_hourly.
groupby(['age_group_5', 'race_grouping']).agg({'current_base_pay': [np.
count_nonzero, np.median]})
suppress(current_news_race_group_age_5_hourly)
```

```python
[116]:
count_nonzero  median
age_group_5  race_grouping
<25  person of color  6.00  29.49
     white  7.00  18.50
25-29  person of color  12.00  27.07
       white  11.00  30.77
30-34  white  9.00  33.73
35-39  white  5.00  34.72
40-44  white  7.00  41.43
45-49  white  6.00  48.55
50-54  white  5.00  38.93
55-59  white  6.00  33.93
60-64  white  5.00  38.82
```
```python
[117]: current_news_race_group_age_10_salary = news_salaried.
    → groupby(['age_group_10', 'race_grouping']).agg({'current_base_pay': [np.
    → count_nonzero, np.median]})
suppress(current_news_race_group_age_10_salary)

[117]:
<table>
<thead>
<tr>
<th>age_group_10</th>
<th>race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>person of color</td>
<td>11.00</td>
<td>63780.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>12.00</td>
<td>65140.00</td>
</tr>
<tr>
<td>25-34</td>
<td>person of color</td>
<td>55.00</td>
<td>83340.00</td>
</tr>
<tr>
<td></td>
<td>unknown</td>
<td>14.00</td>
<td>106890.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>125.00</td>
<td>86000.00</td>
</tr>
<tr>
<td>35-44</td>
<td>person of color</td>
<td>38.00</td>
<td>102890.00</td>
</tr>
<tr>
<td></td>
<td>unknown</td>
<td>7.00</td>
<td>140280.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>104.00</td>
<td>115258.47</td>
</tr>
<tr>
<td>45-54</td>
<td>person of color</td>
<td>26.00</td>
<td>106932.24</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>84.00</td>
<td>116687.17</td>
</tr>
<tr>
<td>55-64</td>
<td>person of color</td>
<td>8.00</td>
<td>140423.62</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>68.00</td>
<td>140051.84</td>
</tr>
<tr>
<td>65+</td>
<td>white</td>
<td>13.00</td>
<td>159300.00</td>
</tr>
</tbody>
</table>

[118]: current_news_race_group_age_10_hourly = news_hourly.
    → groupby(['age_group_10', 'race_grouping']).agg({'current_base_pay': [np.
    → count_nonzero, np.median]})
suppress(current_news_race_group_age_10_hourly)

[118]:
<table>
<thead>
<tr>
<th>age_group_10</th>
<th>race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>person of color</td>
<td>6.00</td>
<td>29.49</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>7.00</td>
<td>18.50</td>
</tr>
<tr>
<td>25-34</td>
<td>person of color</td>
<td>13.00</td>
<td>29.12</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>20.00</td>
<td>31.26</td>
</tr>
<tr>
<td>35-44</td>
<td>person of color</td>
<td>5.00</td>
<td>23.93</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>12.00</td>
<td>35.31</td>
</tr>
<tr>
<td>45-54</td>
<td>white</td>
<td>11.00</td>
<td>41.38</td>
</tr>
<tr>
<td>55-64</td>
<td>white</td>
<td>11.00</td>
<td>34.89</td>
</tr>
</tbody>
</table>

[119]: current_news_race_under_40_salaried = news_salaried[news_salaried['age'] < 40].
    → groupby(['race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.
    → median]})
suppress_median(current_news_race_under_40_salaried)

[119]:
<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>White (United States of America)</td>
<td>198.00</td>
<td>90780.00</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>24.00</td>
<td>87970.47</td>
</tr>
<tr>
<td>Asian (United States of America)</td>
<td>33.00</td>
<td>87000.00</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>19.00</td>
<td>79618.25</td>
</tr>
<tr>
<td>Prefer Not to Disclose (United States of America)</td>
<td>6.00</td>
<td>77750.00</td>
</tr>
</tbody>
</table>
```
Two or More Races (United States of America) 13.00 76380.00

[120]: current_news_race_over_40_salaried = news_salaried[news_salaried['age'] > 39].
  .groupby(['race_grouping']).agg({'current_base_pay': [np.count_nonzero, np.
  .median]})
  suppress_median(current_news_race_over_40_salaried)

[120]:
  count_nonzero  median
  race_grouping
  unknown        12.00 151407.91
  white          208.00 128484.46
  person of color 50.00 110844.65

[121]: current_news_race_under_40_hourly = news_hourly[news_hourly['age'] < 40].
  .groupby(['race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.
  .median]})
  suppress_median(current_news_race_under_40_hourly)

[121]:
  count_nonzero  median
  race_ethnicity
  White (United States of America) 32.00 31.96
  Black or African American (United States of Ame... 10.00 29.95
  Asian (United States of America)  7.00 25.02

[122]: current_news_race_over_40_hourly = news_hourly[news_hourly['age'] > 39].
  .groupby(['race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.
  .median]})
  suppress_median(current_news_race_over_40_hourly)

[122]:
  count_nonzero  median
  race_ethnicity
  White (United States of America) 32.00 39.86

1.5.3 Gender x race/ethnicity

[123]: current_news_race_gender_salaried = news_salaried.
  .groupby(['race_ethnicity', 'gender']).agg({'current_base_pay': [np.
  .count_nonzero]})
  suppress(current_news_race_gender_salaried)

[123]:
  count_nonzero
  race_ethnicity  gender
  Asian (United States of America) Female  34.00
                          Male   12.00
  Black or African American (United States of Ame... Female  24.00
                          Male   24.00
  Hispanic or Latino (United States of America)  Female  14.00
                          Male   14.00
  Prefer Not to Disclose (United States of America) Male   5.00
  Two or More Races (United States of America) Female  9.00
<table>
<thead>
<tr>
<th>Race/Ethnicity</th>
<th>Gender</th>
<th>Current Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>White (United States of America)</td>
<td>Male</td>
<td>5.00</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>188.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>218.00</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>Male</td>
<td>218.00</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>8.00</td>
</tr>
<tr>
<td>Asian (United States of America)</td>
<td>Male</td>
<td>5.00</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>8.00</td>
</tr>
</tbody>
</table>

```python
[124]:
current_news_race_gender_hourly = news_hourly.
    → groupby(['race_ethnicity', 'gender']).agg({'current_base_pay': [np.count_nonzero]})
suppress(current_news_race_gender_hourly)
```

```python
[124]:
count_nonzero
gender
race_ethnicity
Asian (United States of America) | Female | 8.00 |
Black or African American (United States of America) | Male   | 5.00 |
White (United States of America) | Female | 41.00 |
                                      | Male   | 23.00 |
```

```python
[125]:
current_news_race_gender_median_salaried = news_salaried.
    → groupby(['race_grouping', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_news_race_gender_median_salaried)
```

```python
[125]:
count_nonzero median
gender
race_grouping
person of color | Female | 83.00 | 86511.34 |
                | Male   | 56.00 | 101575.00 |
unknown        | Female | 13.00 | 129970.48 |
                | Male   | 16.00 | 135280.00 |
white          | Female | 188.00| 99640.00  |
                | Male   | 218.00| 117451.77 |
```

```python
[126]:
current_news_race_gender_median_hourly = news_hourly.
    → groupby(['race_ethnicity', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_news_race_gender_median_hourly)
```

```python
[126]:
count_nonzero median
gender
race_ethnicity
Asian (United States of America) | Female | 8.00 | 29.99 |
Black or African American (United States of America) | Female | 8.00 | 30.97 |
                                       | Male   | 5.00 | 20.91 |
White (United States of America) | Female | 41.00| 34.72 |
                                       | Male   | 23.00|      |
```
```python
[127]: current_news_race_gender_under_40_salaried = news_salaried[news_salaried['age'] < 40].groupby(['race_ethnicity', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_news_race_gender_under_40_salaried)

[127]:

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>25.00</td>
<td>86000.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>8.00</td>
<td>102890.00</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>16.00</td>
<td>85390.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>8.00</td>
<td>127890.00</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>Female</td>
<td>12.00</td>
<td>80059.12</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>7.00</td>
<td>75000.00</td>
</tr>
<tr>
<td>Two or More Races (United States of America)</td>
<td>Female</td>
<td>9.00</td>
<td>75000.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>7.00</td>
<td>95655.73</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>Female</td>
<td>105.00</td>
<td>85780.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>93.00</td>
<td></td>
</tr>
</tbody>
</table>

[128]: current_news_race_gender_under_40_hourly = news_hourly[news_hourly['age'] < 40].groupby(['race_ethnicity', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_news_race_gender_under_40_hourly)

[128]:

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>5.00</td>
<td>25.02</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>11.00</td>
<td>31.92</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>6.00</td>
<td>30.97</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>33.73</td>
<td></td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>Female</td>
<td>21.00</td>
<td>31.92</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td></td>
<td>33.73</td>
</tr>
</tbody>
</table>
```

Male 33.38
current_news_race_gender_over_40_salaried = news_salaried[news_salaried['age'] > 39].groupby(['race_ethnicity', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_news_race_gender_over_40_salaried)

race_ethnicity                        gender    count_nonzero    median
Asian (United States of America)       Female   9.00           111761.01
Black or African American (United States of America) Female   8.00           115002.24
                                   Male        16.00         107464.14
Hispanic or Latino (United States of America) Male        7.00           126580.00
White (United States of America)       Female   83.00          122916.97
                                   Male        125.00

current_news_race_gender_over_40_hourly = news_hourly[news_hourly['age'] > 39].groupby(['race_ethnicity', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_news_race_gender_over_40_hourly)

race_ethnicity                        gender    count_nonzero    median
White (United States of America)       Female   20.00           42.39
                                   Male        12.00           33.17

1.5.4 Years of service

current_news_yos_salary = news_salaried.groupby(['years_of_service_grouped']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_news_yos_salary)

years_of_service_grouped    count_nonzero    median
0                        65.00            90000.00
1-2                      128.00            93780.00
3-5                      146.00            92170.07
6-10                    60.00            112925.50
11-15                   50.00            110823.23
16-20                  68.00            127654.56
21-25                 24.00            143197.97
```python
[132]:
current_news_yos_hourly = news_hourly.groupby(['years_of_service_grouped']).
agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_news_yos_hourly)

[132]:
count_nonzero median
+----------------+--------+--------+
| years_of_service_grouped |       |
| 0                | 16.00  | 29.49  |
| 1-2              | 26.00  | 32.71  |
| 3-5              | 9.00   | 32.97  |
| 6-10             | 15.00  | 35.91  |
| 11-15            | 10.00  | 36.54  |
| 16-20            | 11.00  | 32.31  |
| 21-25            | 5.00   | 38.93  |

[133]:
current_news_yos_gender_salary = news_salaried.
  → groupby(['years_of_service_grouped', 'gender']).agg({'current_base_pay': [np.
  ↪ count_nonzero, np.median]})
suppress(current_news_yos_gender_salary)

[133]:
count_nonzero median
+----------------+--------+--------+
| years_of_service_grouped gender |       |
| 0 Female         | 39.00  | 80000.00 |
| 0 Male           | 26.00  | 105000.00 |
| 1-2 Female       | 70.00  | 87390.00 |
| 1-2 Male         | 58.00  | 101787.80 |
| 3-5 Female       | 72.00  | 88530.00 |
| 3-5 Male         | 74.00  | 95265.36 |
| 6-10 Female      | 26.00  | 100640.36 |
| 6-10 Male        | 34.00  | 119561.75 |
| 11-15 Female     | 25.00  | 98544.65 |
| 11-15 Male       | 25.00  | 129780.00 |
| 16-20 Female     | 28.00  | 119826.17 |
| 16-20 Male       | 40.00  | 129744.80 |
| 21-25 Female     | 11.00  | 134780.00 |
| 21-25 Male       | 13.00  | 148416.62 |
| 25+ Female       | 13.00  | 142280.00 |
| 25+ Male         | 20.00  | 131793.39 |

[134]:
current_news_yos_gender_hourly = news_hourly.
  → groupby(['years_of_service_grouped', 'gender']).agg({'current_base_pay': [np.
  ↪ count_nonzero, np.median]})
suppress(current_news_yos_gender_hourly)

[134]:
count_nonzero median
+----------------+--------+--------+
| years_of_service_grouped gender |       |
| 0 Female         | 11.00  | 28.21  |
| 0 Male           | 5.00   | 30.77  |
| 1-2 Female       | 18.00  | 32.36  |
```
<table>
<thead>
<tr>
<th>Years of Service</th>
<th>Race/Ethnicity</th>
<th>Current Base Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>3-5</td>
<td>Male</td>
<td>8.00 33.35</td>
</tr>
<tr>
<td>3-5</td>
<td>Female</td>
<td>6.00 32.47</td>
</tr>
<tr>
<td>6-10</td>
<td>Male</td>
<td>8.00 31.38</td>
</tr>
<tr>
<td>6-10</td>
<td>Female</td>
<td>7.00 36.70</td>
</tr>
<tr>
<td>11-15</td>
<td>Male</td>
<td>9.00 38.36</td>
</tr>
<tr>
<td>11-15</td>
<td>Female</td>
<td>7.00 42.14</td>
</tr>
</tbody>
</table>

```python
[135]: current_news_yos_race_salary = news_salaried.
      ~groupby(['years_of_service_grouped', 'race_ethnicity']).
      ~agg({'current_base_pay': [np.count_nonzero, np.median]})
      suppress(current_news_yos_race_salary)
```

```python
[135]: count_nonzero \\
     years_of_service_grouped race_ethnicity  
     0  Asian (United States of America)    
     7.00  White (United States of America)  
     42.00  Asian (United States of America)  
     1-2  Black or African American (United States of America)  
     13.00  Hispanic or Latino (United States of America)  
     10.00  Two or More Races (United States of America)  
     6.00  White (United States of America)  
     5.00  Asian (United States of America)  
     85.00  Black or African American (United States of America)  
     3-5  Hispanic or Latino (United States of America)  
     12.00  Two or More Races (United States of America)  
     12.00  White (United States of America)  
     14.00  Black or African American (United States of America)  
     5.00  Hispanic or Latino (United States of America)  
     5.00  Two or More Races (United States of America)  
     5.00  White (United States of America)  
     97.00  Black or African American (United States of America)  
     6-10  White (United States of America)  
     45.00  Black or African American (United States of America)  
     11-15  White (United States of America)  
     5.00  Black or African American (United States of America)  
     40.00  White (United States of America)  
     16-20  Black or African American (United States of America)  
     10.00  White (United States of America)  
     53.00  White (United States of America)  
```
<table>
<thead>
<tr>
<th>years_of_service_grouped</th>
<th>median</th>
<th>race_ethnicity</th>
</tr>
</thead>
<tbody>
<tr>
<td>21-25</td>
<td>16.00</td>
<td>White (United States of America)</td>
</tr>
<tr>
<td>17.00</td>
<td></td>
<td>White (United States of America)</td>
</tr>
<tr>
<td>25+</td>
<td>27.00</td>
<td>White (United States of America)</td>
</tr>
<tr>
<td>27.00</td>
<td></td>
<td>White (United States of America)</td>
</tr>
</tbody>
</table>

Median years of service grouped race/ethnicity:

- 0: Asian (United States of America)
- 100000.00: White (United States of America)
- 1-2: Asian (United States of America)
- 84780.00: Black or African American (United States of America)
- 89780.00: Hispanic or Latino (United States of America)
- 82890.00: Two or More Races (United States of America)
- 68000.00: White (United States of America)
- 95780.00: White (United States of America)
- 3-5: Asian (United States of America)
- 93630.07: Black or African American (United States of America)
- 97276.46: Hispanic or Latino (United States of America)
- 80809.07: Two or More Races (United States of America)
- 83340.00: White (United States of America)
- 91687.46: White (United States of America)
- 6-10: White (United States of America)
- 115236.94: Black or African American (United States of America)
- 11-15: Black or African American (United States of America)
- 124080.00: White (United States of America)
- 107685.39: Black or African American (United States of America)
- 16-20: Black or African American (United States of America)
- 104397.79: White (United States of America)
- 134848.85: White (United States of America)
- 21-25: White (United States of America)
- 134780.00: White (United States of America)
- 25+: White (United States of America)
- 135869.69: White (United States of America)
current_news_yos_race_hourly = news_hourly.
   -> groupby(['years_of_service_grouped','race_ethnicity']).
   -> agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_news_yos_race_hourly)

years_of_service_grouped  race_ethnicity  count_nonzero  median
0                         White (United States of America)  6.00  29.49
1-2                       White (United States of America) 18.00  32.84
3-5                       White (United States of America)  6.00  32.47
6-10                      White (United States of America)  9.00  35.91
11-15                     White (United States of America)  8.00  39.87
16-20                     White (United States of America)  9.00  42.14
21-25                     White (United States of America)  5.00  38.93

current_news_yos_race_gender_salary = news_salaried.
   -> groupby(['years_of_service_grouped','race_grouping','gender']).
   -> agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_news_yos_race_gender_salary)

years_of_service_grouped  race_grouping  gender  count_nonzero  median
0  person of color  Female  12.00  76000.00
    Male  6.00  93500.00
    white Female  25.00  85000.00
        Male  17.00  110000.00
1-2 person of color  Female  25.00  82000.00
    Male  9.00  113280.00
    unknown Male  5.00  117780.00
    white Female  41.00  90780.00
        Male  44.00  99780.00
3-5 person of color  Female  25.00  86965.08
    Male  18.00  92764.52
    white Female  43.00  88780.00
        Male  54.00  97690.36
6-10 person of color  Female  5.00  79160.51
    Male  6.00  98630.00
    white Female  20.00  105206.00
Male 25.00 121280.00
11-15 person of color Female 5.00 95410.05
white Female 20.00 98898.12
Male 20.00 128916.43
16-20 person of color Female 6.00 113688.41
Male 8.00 104929.69
white Female 21.00 128399.50
Male 32.00 137948.68
21-25 white Female 8.00 130890.00
Male 9.00 148416.62
25+ white Female 10.00 139074.85
Male 17.00 134957.37

[138]: current_news_yos_race_gender_hourly = news_hourly.
groupby(['years_of_service_grouped', 'race_grouping', 'gender']).
agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_news_yos_race_gender_hourly)

[138]:
\[
\begin{array}{ccc}
\text{years_of_service_grouped} & \text{race_grouping} & \text{gender} \\
0 & \text{person of color} & \text{Female} \\
1-2 & \text{person of color} & \text{Female} \\
& & \text{white} \\
6-10 & \text{white} & \text{Male} \\
11-15 & \text{white} & \text{Female} \\
16-20 & \text{white} & \text{Female} \\
\end{array}
\]

\[
\begin{array}{ccc}
\text{count_nonzero} & \text{median} \\
6.00 & 29.49 \\
6.00 & 31.59 \\
12.00 & 32.71 \\
6.00 & 33.35 \\
5.00 & 35.91 \\
7.00 & 41.38 \\
6.00 & 42.39 \\
\end{array}
\]

1.5.5 Age

[139]: current_median_news_age_5_salaried = news_salaried.groupby(['age_group_5']).
agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_news_age_5_salaried)

[139]:
\[
\begin{array}{ccc}
\text{age_group_5} & \text{count_nonzero} & \text{median} \\
<25 & 24.00 & 64640.00 \\
25-29 & 91.00 & 80500.00 \\
30-34 & 103.00 & 90780.00 \\
35-39 & 86.00 & 105691.31 \\
40-44 & 63.00 & 125768.93 \\
45-49 & 43.00 & 102795.60 \\
50-54 & 70.00 & 115769.96 \\
55-59 & 51.00 & 147780.00 \\
60-64 & 28.00 & 131216.77 \\
65+ & 15.00 & 157095.42 \\
\end{array}
\]
current_median_news_age_5_hourly = news_hourly.groupby(['age_group_5']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_news_age_5_hourly)

<table>
<thead>
<tr>
<th>age_group_5</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>14.00</td>
<td>29.49</td>
</tr>
<tr>
<td>25-29</td>
<td>23.00</td>
<td>30.77</td>
</tr>
<tr>
<td>30-34</td>
<td>11.00</td>
<td>33.73</td>
</tr>
<tr>
<td>35-39</td>
<td>8.00</td>
<td>33.92</td>
</tr>
<tr>
<td>40-44</td>
<td>9.00</td>
<td>33.13</td>
</tr>
<tr>
<td>45-49</td>
<td>7.00</td>
<td>50.38</td>
</tr>
<tr>
<td>50-54</td>
<td>7.00</td>
<td>33.38</td>
</tr>
<tr>
<td>55-59</td>
<td>7.00</td>
<td>34.89</td>
</tr>
<tr>
<td>60-64</td>
<td>5.00</td>
<td>38.82</td>
</tr>
<tr>
<td>65+</td>
<td>5.00</td>
<td>42.64</td>
</tr>
</tbody>
</table>

current_median_news_age_10_salaried = news_salaried.groupby(['age_group_10']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_news_age_10_salaried)

<table>
<thead>
<tr>
<th>age_group_10</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>24.00</td>
<td>64640.00</td>
</tr>
<tr>
<td>25-34</td>
<td>194.00</td>
<td>85890.00</td>
</tr>
<tr>
<td>35-44</td>
<td>149.00</td>
<td>115236.94</td>
</tr>
<tr>
<td>45-54</td>
<td>113.00</td>
<td>114803.00</td>
</tr>
<tr>
<td>55-64</td>
<td>79.00</td>
<td>141015.94</td>
</tr>
<tr>
<td>65+</td>
<td>15.00</td>
<td>157095.42</td>
</tr>
</tbody>
</table>

current_median_news_age_10_hourly = news_hourly.groupby(['age_group_10']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_news_age_10_hourly)

<table>
<thead>
<tr>
<th>age_group_10</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>14.00</td>
<td>29.49</td>
</tr>
<tr>
<td>25-34</td>
<td>34.00</td>
<td>31.01</td>
</tr>
<tr>
<td>35-44</td>
<td>17.00</td>
<td>33.13</td>
</tr>
<tr>
<td>45-54</td>
<td>14.00</td>
<td>41.09</td>
</tr>
<tr>
<td>55-64</td>
<td>12.00</td>
<td>35.80</td>
</tr>
<tr>
<td>65+</td>
<td>5.00</td>
<td>42.64</td>
</tr>
</tbody>
</table>

current_news_age_5_yos_salary = news_salaried.groupby(['age_group_5', 'years_of_service_grouped']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_news_age_5_yos_salary)

<table>
<thead>
<tr>
<th>age_group_5 years_of_service_grouped</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age Group</td>
<td>Less Than 25</td>
<td>25-29</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
<td>-------</td>
</tr>
<tr>
<td></td>
<td>&lt;25</td>
<td>1-2</td>
</tr>
<tr>
<td>Base Pay</td>
<td>9.00</td>
<td>13.00</td>
</tr>
<tr>
<td>Median</td>
<td>66000.00</td>
<td>63780.00</td>
</tr>
</tbody>
</table>

```python
[144]: current_news_age_5_yos_hourly = news_hourly.
    -> groupby(['age_group_5', 'years_of_service_grouped']).agg({'current_base_pay': [np.count_nonzero, np.median]})
    suppress(current_news_age_5_yos_hourly)
```

```
<table>
<thead>
<tr>
<th>age_group_5</th>
<th>years_of_service_grouped</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td></td>
<td>0</td>
<td>6.00</td>
</tr>
<tr>
<td>1-2</td>
<td></td>
<td>6.00</td>
<td>24.11</td>
</tr>
</tbody>
</table>
```

76
1-2  8.00  32.00
25-29  0  8.00  29.49
  1-2  12.00  32.20

[145]: current_news_age_10_yos_salary = news_salaried.
groupby(['age_group_10', 'years_of_service_grouped']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_news_age_10_yos_salary)

[145]:

<table>
<thead>
<tr>
<th>age_group_10</th>
<th>years_of_service_grouped</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td></td>
<td>0</td>
<td>9.00</td>
</tr>
<tr>
<td></td>
<td>1-2</td>
<td>13.00</td>
<td>63780.00</td>
</tr>
<tr>
<td>25-34</td>
<td></td>
<td>0</td>
<td>32.00</td>
</tr>
<tr>
<td></td>
<td>1-2</td>
<td>58.00</td>
<td>86280.00</td>
</tr>
<tr>
<td></td>
<td>3-5</td>
<td>84.00</td>
<td>85890.00</td>
</tr>
<tr>
<td></td>
<td>6-10</td>
<td>16.00</td>
<td>94675.93</td>
</tr>
<tr>
<td>35-44</td>
<td></td>
<td>0</td>
<td>38.00</td>
</tr>
<tr>
<td></td>
<td>1-2</td>
<td>16.00</td>
<td>125000.00</td>
</tr>
<tr>
<td></td>
<td>3-5</td>
<td>36.00</td>
<td>116530.00</td>
</tr>
<tr>
<td></td>
<td>6-10</td>
<td>38.00</td>
<td>110934.68</td>
</tr>
<tr>
<td></td>
<td>11-15</td>
<td>25.00</td>
<td>115236.94</td>
</tr>
<tr>
<td></td>
<td>16-20</td>
<td>22.00</td>
<td>111892.50</td>
</tr>
<tr>
<td></td>
<td></td>
<td>11.00</td>
<td>118512.33</td>
</tr>
<tr>
<td>45-54</td>
<td></td>
<td>0</td>
<td>6.00</td>
</tr>
<tr>
<td></td>
<td>1-2</td>
<td>15.00</td>
<td>129000.00</td>
</tr>
<tr>
<td></td>
<td>3-5</td>
<td>15.00</td>
<td>94875.00</td>
</tr>
<tr>
<td></td>
<td>6-10</td>
<td>12.00</td>
<td>105762.84</td>
</tr>
<tr>
<td></td>
<td>11-15</td>
<td>17.00</td>
<td>111761.01</td>
</tr>
<tr>
<td></td>
<td>16-20</td>
<td>32.00</td>
<td>112559.53</td>
</tr>
<tr>
<td></td>
<td>21-25</td>
<td>32.00</td>
<td>112954.67</td>
</tr>
<tr>
<td></td>
<td>25+</td>
<td>22.00</td>
<td>140051.84</td>
</tr>
<tr>
<td></td>
<td></td>
<td>10.00</td>
<td>145022.96</td>
</tr>
<tr>
<td></td>
<td></td>
<td>23.00</td>
<td>142280.00</td>
</tr>
<tr>
<td>55-64</td>
<td>1-2</td>
<td>6.00</td>
<td>138696.68</td>
</tr>
<tr>
<td></td>
<td>3-5</td>
<td>6.00</td>
<td>137467.86</td>
</tr>
<tr>
<td></td>
<td>11-15</td>
<td>6.00</td>
<td>106232.39</td>
</tr>
<tr>
<td></td>
<td>16-20</td>
<td>22.00</td>
<td>140051.84</td>
</tr>
<tr>
<td></td>
<td>21-25</td>
<td>10.00</td>
<td>145022.96</td>
</tr>
<tr>
<td></td>
<td>25+</td>
<td>23.00</td>
<td>142280.00</td>
</tr>
</tbody>
</table>

[146]: current_news_age_10_yos_hourly = news_hourly.
groupby(['age_group_10', 'years_of_service_grouped']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_news_age_10_yos_hourly)

[146]:

<table>
<thead>
<tr>
<th>age_group_10</th>
<th>years_of_service_grouped</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td></td>
<td>0</td>
<td>6.00</td>
</tr>
<tr>
<td></td>
<td>1-2</td>
<td>8.00</td>
<td>24.11</td>
</tr>
<tr>
<td>25-34</td>
<td></td>
<td>0</td>
<td>9.00</td>
</tr>
<tr>
<td></td>
<td>1-2</td>
<td>16.00</td>
<td>32.71</td>
</tr>
</tbody>
</table>
3-5 6.00 29.99
35-44 11-15 6.00 33.92

```
[147]: current_median_news_age_5_gender_salaried = news_salaried.
    groupby(["age_group_5", 'gender']).agg({'current_base_pay': [np.
    count_nonzero, np.median]})
suppress(current_median_news_age_5_gender_salaried)

[147]:

<table>
<thead>
<tr>
<th>age_group_5</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>Female</td>
<td>19.00</td>
<td>64280.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>5.00</td>
<td>72000.00</td>
</tr>
<tr>
<td>25-29</td>
<td>Female</td>
<td>60.00</td>
<td>80000.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>31.00</td>
<td>85500.00</td>
</tr>
<tr>
<td>30-34</td>
<td>Female</td>
<td>57.00</td>
<td>87000.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>46.00</td>
<td>97827.86</td>
</tr>
<tr>
<td>35-39</td>
<td>Female</td>
<td>38.00</td>
<td>98891.57</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>48.00</td>
<td>116030.00</td>
</tr>
<tr>
<td>40-44</td>
<td>Female</td>
<td>22.00</td>
<td>133200.02</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>41.00</td>
<td>125000.00</td>
</tr>
<tr>
<td>45-49</td>
<td>Female</td>
<td>20.00</td>
<td>117294.59</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>23.00</td>
<td>99725.00</td>
</tr>
<tr>
<td>50-54</td>
<td>Female</td>
<td>29.00</td>
<td>108864.49</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>41.00</td>
<td>126280.47</td>
</tr>
<tr>
<td>55-59</td>
<td>Female</td>
<td>22.00</td>
<td>145654.99</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>29.00</td>
<td>147780.00</td>
</tr>
<tr>
<td>60-64</td>
<td>Female</td>
<td>12.00</td>
<td>129324.85</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>16.00</td>
<td>131216.77</td>
</tr>
<tr>
<td>65+</td>
<td>Female</td>
<td>5.00</td>
<td>157095.42</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>10.00</td>
<td>156259.68</td>
</tr>
</tbody>
</table>
```

```
[148]: current_median_news_age_5_gender_hourly = news_hourly.
    groupby(["age_group_5", 'gender']).agg({'current_base_pay': [np.
    count_nonzero, np.median]})
suppress(current_median_news_age_5_gender_hourly)

[148]:

<table>
<thead>
<tr>
<th>age_group_5</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>Female</td>
<td>12.00</td>
<td>31.38</td>
</tr>
<tr>
<td>25-29</td>
<td>Female</td>
<td>17.00</td>
<td>31.17</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>6.00</td>
<td>20.96</td>
</tr>
<tr>
<td>30-34</td>
<td>Male</td>
<td>7.00</td>
<td>33.73</td>
</tr>
<tr>
<td>35-39</td>
<td>Female</td>
<td>5.00</td>
<td>31.92</td>
</tr>
<tr>
<td>40-44</td>
<td>Female</td>
<td>5.00</td>
<td>41.23</td>
</tr>
<tr>
<td>45-49</td>
<td>Female</td>
<td>6.00</td>
<td>48.55</td>
</tr>
<tr>
<td>50-54</td>
<td>Female</td>
<td>5.00</td>
<td>38.93</td>
</tr>
<tr>
<td>55-59</td>
<td>Male</td>
<td>5.00</td>
<td>34.89</td>
</tr>
</tbody>
</table>
```
current_median_news_age_10_gender_salaried = news_salaried.
    .groupby(['age_group_10', 'gender']).agg({'current_base_pay': [np.
        count_nonzero, np.median]})
suppress(current_median_news_age_10_gender_salaried)

count_nonzero  median  
age_group_10  gender  
<25  Female  19.00  64280.00  
     Male  5.00  72000.00  
25-34 Female  117.00  83146.67  
     Male  77.00  92500.00  
35-44 Female  60.00  105691.31  
     Male  89.00  118785.00  
45-54 Female  49.00  108864.49  
     Male  64.00  117981.79  
55-64 Female  34.00  140423.62  
     Male  45.00  146541.57  
65+  Female  5.00  157095.42  
     Male  10.00  156259.68  

current_median_news_age_10_gender_hourly = news_hourly.
    .groupby(['age_group_10', 'gender']).agg({'current_base_pay': [np.
        count_nonzero, np.median]})
suppress(current_median_news_age_10_gender_hourly)

count_nonzero  median  
age_group_10  gender  
<25  Female  12.00  31.38  
     Male  13.00  30.77  
25-34 Female  21.00  31.17  
     Male  10.00  33.12  
35-44 Female  10.00  33.12  
     Male  7.00  35.90  
45-54 Female  11.00  41.38  
     Male  5.00  42.14  
55-64 Female  5.00  42.14  
     Male  7.00  33.41  

current_median_news_age_5_race_salaried = news_salaried.
    .groupby(['age_group_5', 'race_ethnicity']).agg({'current_base_pay': [np.
        count_nonzero, np.median]})
suppress(current_median_news_age_5_race_salaried)

count_nonzero  \ 
age_group_5 race_ethnicity  
<25  Asian (United States of America)  5.00  
     White (United States of America)  12.00  
25-29 Asian (United States of America)  11.00  
     Black or African American (United States of America)  6.00  
     Two or More Races (United States of America)  6.00  
     White (United States of America)  59.00  

79
<table>
<thead>
<tr>
<th>Age Group</th>
<th>Race/Ethnicity</th>
<th>Median Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>30-34</td>
<td>Asian (United States of America)</td>
<td>10.00</td>
</tr>
<tr>
<td></td>
<td>Black or African American (United States of America)</td>
<td>9.00</td>
</tr>
<tr>
<td></td>
<td>Hispanic or Latino (United States of America)</td>
<td>6.00</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>66.00</td>
</tr>
<tr>
<td>35-39</td>
<td>Asian (United States of America)</td>
<td>7.00</td>
</tr>
<tr>
<td></td>
<td>Black or African American (United States of America)</td>
<td>7.00</td>
</tr>
<tr>
<td></td>
<td>Hispanic or Latino (United States of America)</td>
<td>7.00</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>61.00</td>
</tr>
<tr>
<td>40-44</td>
<td>Black or African American (United States of America)</td>
<td>6.00</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>43.00</td>
</tr>
<tr>
<td>45-49</td>
<td>White (United States of America)</td>
<td>36.00</td>
</tr>
<tr>
<td>50-54</td>
<td>Asian (United States of America)</td>
<td>5.00</td>
</tr>
<tr>
<td></td>
<td>Black or African American (United States of America)</td>
<td>10.00</td>
</tr>
<tr>
<td></td>
<td>Hispanic or Latino (United States of America)</td>
<td>5.00</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>48.00</td>
</tr>
<tr>
<td>55-59</td>
<td>White (United States of America)</td>
<td>43.00</td>
</tr>
<tr>
<td>60-64</td>
<td>White (United States of America)</td>
<td>25.00</td>
</tr>
<tr>
<td>65+</td>
<td>White (United States of America)</td>
<td>13.00</td>
</tr>
</tbody>
</table>

age_group_5 race_ethnicity
<25  Asian (United States of America)  | 65780.00  |
     White (United States of America)  | 65140.00  |
25-29 | Asian (United States of America)     | 77000.00  |
     Black or African American (United States of America) | 81000.00  |
     Two or More Races (United States of America)  | 75690.00  |
     White (United States of America)  | 81756.58  |
30-34 | Asian (United States of America)     | 95780.00  |
     Black or African American (United States of America) | 88132.61  |
     Hispanic or Latino (United States of America) | 80596.26  |
     White (United States of America)  | 92640.00  |
35-39 | Asian (United States of America)     | 115000.00 |
     Black or African American (United States of America) | 96147.48  |
     Hispanic or Latino (United States of America) | 79618.25  |
     White (United States of America)  | 105780.00 |
40-44 | Black or African American (United States of America) | 122610.00 |
     White (United States of America)  | 126080.00 |
45-49 | White (United States of America)     | 104522.64 |
50-54 | Asian (United States of America)     | 103150.00 |
     Black or African American (United States of America) | 106932.24 |
     Hispanic or Latino (United States of America) | 126764.81 |
     White (United States of America)  | 120481.79 |
55-59 | White (United States of America)     | 147780.00 |
60-64 | White (United States of America)     | 122780.00 |
65+  | White (United States of America)     | 159300.00 |

80
current_median_news_age_5_race_hourly = news_hourly.
groupby(['age_group_5', 'race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_news_age_5_race_hourly)

current_median_news_age_5_race_group_salaried = news_salaried.
groupby(['age_group_5', 'race_grouping']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_news_age_5_race_group_salaried)
40-44  person of color  15.00  108324.02
        unknown       5.00  145500.00
        white         43.00  126080.00
45-49  person of color  6.00  84937.50
        white         36.00  104522.64
50-54  person of color  20.00  109396.39
        white         48.00  120481.79
55-59  person of color  6.00  131686.62
        white         43.00  147780.00
60-64  white           25.00  122780.00
65+    white           13.00  159300.00

[154]: current_median_news_age_5_race_group_hourly = news_hourly.
         .groupby(['age_group_5','race_grouping']).agg({'current_base_pay': [np.
         .count_nonzero, np.median]})
    suppress(current_median_news_age_5_race_group_hourly)

[154]:
   age_group_5  race_grouping     count_nonzero  median
   <25          person of color    6.00        29.49
                white            7.00        18.50
   25-29        person of color    12.00       27.07
                white           11.00       30.77
   30-34        white             9.00        33.73
   35-39        white             5.00        34.72
   40-44        white             7.00        41.43
   45-49        white             6.00        48.55
   50-54        white             5.00        38.93
   55-59        white             6.00        33.93
   60-64        white             5.00        38.82

[155]: current_median_news_age_10_race_salaried = news_salaried.
         .groupby(['age_group_10','race_ethnicity']).agg({'current_base_pay': [np.
         .count_nonzero, np.median]})
    suppress(current_median_news_age_10_race_salaried)

[155]:
   age_group_10  race_ethnicity  count_nonzero
   <25          Asian (United States of America)  5.00
                White (United States of America)  12.00
   25-34        Asian (United States of America) 21.00
                Black or African American (United States of Ame...  15.00
                Hispanic or Latino (United States of America)  10.00
                Prefer Not to Disclose (United States of America)  5.00
                Two or More Races (United States of America)  9.00
                White (United States of America)  125.00
   35-44        Asian (United States of America)  11.00
                Black or African American (United States of Ame...  13.00

82
<table>
<thead>
<tr>
<th>Age Group</th>
<th>Race/Ethnicity</th>
<th>Median Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>45-54</td>
<td>Hispanic or Latino</td>
<td>10.00</td>
</tr>
<tr>
<td></td>
<td>White</td>
<td>104.00</td>
</tr>
<tr>
<td></td>
<td>Asian</td>
<td>7.00</td>
</tr>
<tr>
<td></td>
<td>Black or African American</td>
<td>12.00</td>
</tr>
<tr>
<td></td>
<td>Hispanic or Latino</td>
<td>6.00</td>
</tr>
<tr>
<td></td>
<td>White</td>
<td>84.00</td>
</tr>
<tr>
<td>55-64</td>
<td>Black or African American</td>
<td>5.00</td>
</tr>
<tr>
<td></td>
<td>White</td>
<td>68.00</td>
</tr>
<tr>
<td>65+</td>
<td>White</td>
<td>13.00</td>
</tr>
</tbody>
</table>

```python
# Count non-zero values
count_nonzero = Suppress(current_median_news_age_10_race_hourly)

# Median pay by age group and race/ethnicity
median_pay =
```

<table>
<thead>
<tr>
<th>Age Group</th>
<th>Race/Ethnicity</th>
<th>Median Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>Asian</td>
<td>65780.00</td>
</tr>
<tr>
<td></td>
<td>White</td>
<td>65140.00</td>
</tr>
<tr>
<td>25-34</td>
<td>Asian</td>
<td>86000.00</td>
</tr>
<tr>
<td></td>
<td>Black or African American</td>
<td>87000.00</td>
</tr>
<tr>
<td></td>
<td>Hispanic or Latino</td>
<td>81249.94</td>
</tr>
<tr>
<td></td>
<td>Prefer Not to Disclose</td>
<td>78500.00</td>
</tr>
<tr>
<td></td>
<td>Two or More Races</td>
<td>76380.00</td>
</tr>
<tr>
<td></td>
<td>White</td>
<td>86000.00</td>
</tr>
<tr>
<td>35-44</td>
<td>Asian</td>
<td>108324.02</td>
</tr>
<tr>
<td></td>
<td>Black or African American</td>
<td>118530.00</td>
</tr>
<tr>
<td></td>
<td>Hispanic or Latino</td>
<td>90390.04</td>
</tr>
<tr>
<td></td>
<td>White</td>
<td>115258.47</td>
</tr>
<tr>
<td>45-54</td>
<td>Asian</td>
<td>111761.01</td>
</tr>
<tr>
<td></td>
<td>Black or African American</td>
<td>102465.54</td>
</tr>
<tr>
<td></td>
<td>Hispanic or Latino</td>
<td>126672.40</td>
</tr>
<tr>
<td></td>
<td>White</td>
<td>116687.17</td>
</tr>
<tr>
<td>55-64</td>
<td>Black or African American</td>
<td>123541.95</td>
</tr>
<tr>
<td></td>
<td>White</td>
<td>140051.84</td>
</tr>
<tr>
<td>65+</td>
<td>White</td>
<td>159300.00</td>
</tr>
</tbody>
</table>
```

```python
# Suppress count_nonzero values
suppress(current_median_news_age_10_race_hourly)
```

83
<table>
<thead>
<tr>
<th>age_group 10</th>
<th>race_ethnicity</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>White (United States of America)</td>
<td>18.50</td>
</tr>
<tr>
<td>25-34</td>
<td>Black or African American (United States of America)</td>
<td>30.15</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>31.26</td>
</tr>
<tr>
<td>35-44</td>
<td>White (United States of America)</td>
<td>35.31</td>
</tr>
<tr>
<td>45-54</td>
<td>White (United States of America)</td>
<td>41.38</td>
</tr>
<tr>
<td>55-64</td>
<td>White (United States of America)</td>
<td>34.89</td>
</tr>
</tbody>
</table>

```python
[157]: current_median_news_age_10_race_group_salaried = news_salaried.
groupby(['age_group_10', 'race_grouping']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_news_age_10_race_group_salaried)
```

<table>
<thead>
<tr>
<th>age_group 10</th>
<th>race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>person of color</td>
<td>11.00</td>
<td>63780.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>12.00</td>
<td>65140.00</td>
</tr>
<tr>
<td>25-34</td>
<td>person of color</td>
<td>55.00</td>
<td>83340.00</td>
</tr>
<tr>
<td></td>
<td>unknown</td>
<td>14.00</td>
<td>106890.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>125.00</td>
<td>86000.00</td>
</tr>
<tr>
<td>35-44</td>
<td>person of color</td>
<td>38.00</td>
<td>102890.00</td>
</tr>
<tr>
<td></td>
<td>unknown</td>
<td>7.00</td>
<td>140280.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>104.00</td>
<td>115258.47</td>
</tr>
<tr>
<td>45-54</td>
<td>person of color</td>
<td>26.00</td>
<td>106932.24</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>84.00</td>
<td>116687.17</td>
</tr>
<tr>
<td>55-64</td>
<td>person of color</td>
<td>8.00</td>
<td>140423.62</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>68.00</td>
<td>140051.84</td>
</tr>
<tr>
<td>65+</td>
<td>white</td>
<td>13.00</td>
<td>159300.00</td>
</tr>
</tbody>
</table>

```python
[158]: current_median_news_age_10_race_group_hourly = news_hourly.
groupby(['age_group_10', 'race_grouping']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_news_age_10_race_group_hourly)
```

<table>
<thead>
<tr>
<th>age_group 10</th>
<th>race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>person of color</td>
<td>6.00</td>
<td>29.49</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>7.00</td>
<td>18.50</td>
</tr>
<tr>
<td>25-34</td>
<td>person of color</td>
<td>13.00</td>
<td>29.12</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>20.00</td>
<td>31.26</td>
</tr>
<tr>
<td>35-44</td>
<td>person of color</td>
<td>5.00</td>
<td>23.93</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>12.00</td>
<td>35.31</td>
</tr>
<tr>
<td>45-54</td>
<td>white</td>
<td>11.00</td>
<td>41.38</td>
</tr>
<tr>
<td>55-64</td>
<td>white</td>
<td>11.00</td>
<td>34.89</td>
</tr>
</tbody>
</table>

```python
[159]: current_median_news_age_5_race_gender_salaried = news_salaried.
groupby(['age_group_5', 'race_ethnicity', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_news_age_5_race_gender_salaried)
```
<table>
<thead>
<tr>
<th>Age Group</th>
<th>Race/Ethnicity</th>
<th>Gender</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>5.00</td>
</tr>
<tr>
<td>25-29</td>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>9.00</td>
</tr>
<tr>
<td>25-29</td>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>5.00</td>
</tr>
<tr>
<td>30-34</td>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>8.00</td>
</tr>
<tr>
<td>30-34</td>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>5.00</td>
</tr>
<tr>
<td>30-34</td>
<td>Hispanic or Latino (United States of America)</td>
<td>Female</td>
<td>6.00</td>
</tr>
<tr>
<td>35-39</td>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>5.00</td>
</tr>
<tr>
<td>35-39</td>
<td>Hispanic or Latino (United States of America)</td>
<td>Male</td>
<td>5.00</td>
</tr>
<tr>
<td>35-39</td>
<td>White (United States of America)</td>
<td>Female</td>
<td>32.00</td>
</tr>
<tr>
<td>35-39</td>
<td>White (United States of America)</td>
<td>Male</td>
<td>35.00</td>
</tr>
<tr>
<td>40-44</td>
<td>Black or African American (United States of America)</td>
<td>Male</td>
<td>5.00</td>
</tr>
<tr>
<td>40-44</td>
<td>White (United States of America)</td>
<td>Female</td>
<td>11.00</td>
</tr>
<tr>
<td>45-49</td>
<td>White (United States of America)</td>
<td>Female</td>
<td>45.00</td>
</tr>
<tr>
<td>45-49</td>
<td>White (United States of America)</td>
<td>Male</td>
<td>32.00</td>
</tr>
<tr>
<td>50-54</td>
<td>Black or African American (United States of America)</td>
<td>Male</td>
<td>5.00</td>
</tr>
<tr>
<td>50-54</td>
<td>Hispanic or Latino (United States of America)</td>
<td>Male</td>
<td>6.00</td>
</tr>
<tr>
<td>50-54</td>
<td>White (United States of America)</td>
<td>Female</td>
<td>5.00</td>
</tr>
<tr>
<td>Age Group</td>
<td>Race/Ethnicity</td>
<td>Gender</td>
<td>Median Age</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------------------</td>
<td>--------</td>
<td>------------</td>
</tr>
<tr>
<td>&lt;25</td>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>65780.00</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>Female</td>
<td>64280.00</td>
</tr>
<tr>
<td>25-29</td>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>77000.00</td>
</tr>
<tr>
<td></td>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>80000.00</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>Female</td>
<td>81878.29</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>76780.00</td>
</tr>
<tr>
<td>30-34</td>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>100780.00</td>
</tr>
<tr>
<td></td>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>85780.00</td>
</tr>
<tr>
<td></td>
<td>Hispanic or Latino (United States of America)</td>
<td>Female</td>
<td>80596.26</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>Female</td>
<td>87660.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>94280.00</td>
</tr>
<tr>
<td>35-39</td>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>87808.33</td>
</tr>
<tr>
<td></td>
<td>Hispanic or Latino (United States of America)</td>
<td>Male</td>
<td>75000.00</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>Female</td>
<td>99272.32</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>115280.00</td>
</tr>
<tr>
<td>40-44</td>
<td>Black or African American (United States of America)</td>
<td>Male</td>
<td>124080.00</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>Female</td>
<td>140000.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>125384.46</td>
</tr>
<tr>
<td>45-49</td>
<td>White (United States of America)</td>
<td>Female</td>
<td>106249.68</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>102795.60</td>
</tr>
<tr>
<td>50-54</td>
<td>Black or African American (United States of America)</td>
<td>Male</td>
<td>107464.14</td>
</tr>
<tr>
<td></td>
<td>Hispanic or Latino (United States of America)</td>
<td>Male</td>
<td>126764.81</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>Female</td>
<td>114803.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>128052.85</td>
</tr>
<tr>
<td>55-59</td>
<td>White (United States of America)</td>
<td>Female</td>
<td>138564.42</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>153922.58</td>
</tr>
<tr>
<td>60-64</td>
<td>White (United States of America)</td>
<td>Female</td>
<td>121896.57</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>127476.17</td>
</tr>
<tr>
<td>65+</td>
<td>White (United States of America)</td>
<td>Male</td>
<td>159458.37</td>
</tr>
</tbody>
</table>
```python
current_median_news_age_5_race_gender_hourly = news_hourly.
groupby(['age_group_5', 'race_ethnicity', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_news_age_5_race_gender_hourly)

<table>
<thead>
<tr>
<th>age_group_5</th>
<th>race_ethnicity</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>White (United States of America)</td>
<td>Female</td>
<td>5.00</td>
<td>32.00</td>
</tr>
<tr>
<td>25-29</td>
<td>White (United States of America)</td>
<td>Female</td>
<td>10.00</td>
<td>31.23</td>
</tr>
<tr>
<td>30-34</td>
<td>White (United States of America)</td>
<td>Male</td>
<td>6.00</td>
<td>34.43</td>
</tr>
<tr>
<td>45-49</td>
<td>White (United States of America)</td>
<td>Female</td>
<td>6.00</td>
<td>48.55</td>
</tr>
<tr>
<td>55-59</td>
<td>White (United States of America)</td>
<td>Male</td>
<td>5.00</td>
<td>34.89</td>
</tr>
</tbody>
</table>

current_median_news_age_5_race_group_gender_salaried = news_salaried.
groupby(['age_group_5', 'race_grouping', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_news_age_5_race_group_gender_salaried)

<table>
<thead>
<tr>
<th>age_group_5</th>
<th>race_grouping</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>person of color</td>
<td>Female</td>
<td>10.00</td>
<td>64390.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>9.00</td>
<td>64280.00</td>
</tr>
<tr>
<td>25-29</td>
<td>person of color</td>
<td>Female</td>
<td>19.00</td>
<td>77000.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td></td>
<td>8.00</td>
<td>88540.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>38.00</td>
<td>81878.29</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td></td>
<td>21.00</td>
<td>76780.00</td>
</tr>
<tr>
<td>30-34</td>
<td>person of color</td>
<td>Female</td>
<td>22.00</td>
<td>86372.54</td>
</tr>
<tr>
<td></td>
<td>unknown</td>
<td>Male</td>
<td>6.00</td>
<td>106000.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>32.00</td>
<td>87660.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td></td>
<td>34.00</td>
<td>94280.00</td>
</tr>
<tr>
<td>35-39</td>
<td>person of color</td>
<td>Female</td>
<td>11.00</td>
<td>96147.48</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td></td>
<td>12.00</td>
<td>115530.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>26.00</td>
<td>99272.32</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td></td>
<td>35.00</td>
<td>115280.00</td>
</tr>
<tr>
<td>40-44</td>
<td>person of color</td>
<td>Female</td>
<td>8.00</td>
<td>113418.18</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td></td>
<td>7.00</td>
<td>94643.69</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>11.00</td>
<td>140000.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td></td>
<td>32.00</td>
<td>125384.46</td>
</tr>
<tr>
<td>45-49</td>
<td>person of color</td>
<td>Male</td>
<td>5.00</td>
<td>75000.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>19.00</td>
<td>106249.68</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td></td>
<td>17.00</td>
<td>102795.60</td>
</tr>
<tr>
<td>50-54</td>
<td>person of color</td>
<td>Female</td>
<td>6.00</td>
<td>102904.48</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td></td>
<td>14.00</td>
<td>113138.72</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>23.00</td>
<td>114803.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td></td>
<td>25.00</td>
<td>128052.85</td>
</tr>
<tr>
<td>55-59</td>
<td>white</td>
<td>Female</td>
<td>16.00</td>
<td>138564.42</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td></td>
<td>27.00</td>
<td>153922.58</td>
</tr>
</tbody>
</table>
```
60-64  white  Female  10.00  121896.57
        Male  15.00  127476.17
65+    white  Male  9.00  159458.37

[162]: current_median_news_age_5_race_group_gender_hourly = news_hourly.
groupby(['age_group_5', 'race_grouping', 'gender']).agg({'current_base_pay':
            [np.count_nonzero, np.median]})
suppress(current_median_news_age_5_race_group_gender_hourly)

[162]:
count_nonzero  median
age_group_5  race_grouping  gender
<25          person of color Female  6.00  29.49
            white  Female  5.00  32.00
25-29        person of color Female  7.00  31.17
            Male  5.00  20.91
            white  Female  10.00  31.23
30-34        white  Male  6.00  34.43
45-49        white  Female  6.00  48.55
55-59        white  Male  5.00  34.89

[163]: current_median_news_age_10_race_gender_salaried = news_salaried.
groupby(['age_group_10', 'race_ethnicity', 'gender']).agg({'current_base_pay':
            [np.count_nonzero, np.median]})
suppress(current_median_news_age_10_race_gender_salaried)

[163]:
count_nonzero  
age_group_10  race_ethnicity  
            gender
<25  Asian (United States of America)  Female  5.00
     White (United States of America)  Female  9.00
     17.00
     Black or African American (United States of Ame... Female  10.00
     Male  5.00
     Hispanic or Latino (United States of America)  Female  8.00
     Two or More Races (United States of America)  Female  6.00
     White (United States of America)  Female  70.00
     Male  55.00
     35-44  Asian (United States of America)  Female  7.00
     Black or African American (United States of Ame... Female  6.00
<table>
<thead>
<tr>
<th>Age Group</th>
<th>Median Income</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>Male 65780.00</td>
</tr>
<tr>
<td></td>
<td>Female 64280.00</td>
</tr>
<tr>
<td>25-34</td>
<td>Male 87000.00</td>
</tr>
<tr>
<td></td>
<td>Female 81000.00</td>
</tr>
<tr>
<td></td>
<td>Male 140000.00</td>
</tr>
<tr>
<td>35-44</td>
<td>Male 125000.00</td>
</tr>
<tr>
<td></td>
<td>Female 91977.90</td>
</tr>
<tr>
<td>45-54</td>
<td>Male 102465.54</td>
</tr>
<tr>
<td></td>
<td>Female 126672.40</td>
</tr>
<tr>
<td>55-64</td>
<td>Male 123530.24</td>
</tr>
<tr>
<td></td>
<td>Female 130924.43</td>
</tr>
<tr>
<td>65+</td>
<td>Male 159458.37</td>
</tr>
</tbody>
</table>

**Notes:**
- Male and Female refer to the gender of the individual.
- Median income refers to the median household income in each specified age group, race/ethnicity, and gender combination.
- The data is sorted by age group, then race/ethnicity, and finally gender.
### current_median_news_age_10_race_gender_hourly

```python
current_median_news_age_10_race_gender_hourly = news_hourly.
groupby(['age_group_10', 'race_ethnicity', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_news_age_10_race_gender_hourly)
```

<table>
<thead>
<tr>
<th>age_group_10</th>
<th>race_ethnicity</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>White (United States of America)</td>
<td>Female</td>
<td>5.00</td>
<td>32.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>7.00</td>
<td>33.73</td>
</tr>
<tr>
<td>25-34</td>
<td>White (United States of America)</td>
<td>Female</td>
<td>13.00</td>
<td>30.84</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>7.00</td>
<td>33.73</td>
</tr>
<tr>
<td>35-44</td>
<td>White (United States of America)</td>
<td>Female</td>
<td>7.00</td>
<td>34.72</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>5.00</td>
<td>35.90</td>
</tr>
<tr>
<td>45-54</td>
<td>White (United States of America)</td>
<td>Female</td>
<td>9.00</td>
<td>44.46</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>7.00</td>
<td>33.41</td>
</tr>
</tbody>
</table>

### current_median_news_age_10_race_group_gender_salaried

```python
current_median_news_age_10_race_group_gender_salaried = news_salaried.
groupby(['age_group_10', 'race_grouping', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_news_age_10_race_group_gender_salaried)
```

<table>
<thead>
<tr>
<th>age_group_10</th>
<th>race_grouping</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>person of color</td>
<td>Female</td>
<td>10.00</td>
<td>64390.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Female</td>
<td>9.00</td>
<td>64280.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>41.00</td>
<td>81999.88</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>14.00</td>
<td>89540.00</td>
</tr>
<tr>
<td></td>
<td>unknown</td>
<td>Female</td>
<td>6.00</td>
<td>92140.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>8.00</td>
<td>120390.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>70.00</td>
<td>84640.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>55.00</td>
<td>90780.00</td>
</tr>
<tr>
<td>35-44</td>
<td>person of color</td>
<td>Female</td>
<td>19.00</td>
<td>100000.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>19.00</td>
<td>113280.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>37.00</td>
<td>105000.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>67.00</td>
<td>120780.00</td>
</tr>
<tr>
<td>45-54</td>
<td>person of color</td>
<td>Female</td>
<td>7.00</td>
<td>108864.49</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>19.00</td>
<td>105000.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>42.00</td>
<td>111589.34</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>42.00</td>
<td>123530.24</td>
</tr>
<tr>
<td>55-64</td>
<td>person of color</td>
<td>Female</td>
<td>6.00</td>
<td>142688.10</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>26.00</td>
<td>130924.43</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>42.00</td>
<td>147160.79</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>9.00</td>
<td>159458.37</td>
</tr>
</tbody>
</table>

### current_median_news_age_10_race_group_gender_hourly

```python
current_median_news_age_10_race_group_gender_hourly = news_hourly.
groupby(['age_group_10', 'race_grouping', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_news_age_10_race_group_gender_hourly)
```
<table>
<thead>
<tr>
<th>age_group_10</th>
<th>race_grouping</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>person of color</td>
<td>Female</td>
<td>6.00</td>
<td>29.49</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>5.00</td>
<td>32.00</td>
</tr>
<tr>
<td>25–34</td>
<td>person of color</td>
<td>Female</td>
<td>7.00</td>
<td>31.17</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Male</td>
<td>6.00</td>
<td>20.96</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>Female</td>
<td>13.00</td>
<td>30.84</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>7.00</td>
<td>33.73</td>
</tr>
<tr>
<td>35–44</td>
<td>white</td>
<td>Female</td>
<td>7.00</td>
<td>34.72</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>5.00</td>
<td>35.90</td>
</tr>
<tr>
<td>45–54</td>
<td>white</td>
<td>Female</td>
<td>9.00</td>
<td>44.46</td>
</tr>
<tr>
<td>55–64</td>
<td>white</td>
<td>Male</td>
<td>7.00</td>
<td>33.41</td>
</tr>
</tbody>
</table>

### 1.5.6 Desks

```python
[167]: current_news_median_desk_salaried = news_salaried.groupby(['desk']).agg({'current_base_pay': [np.count_nonzero, np.median]}).supress_median(current_news_median_desk_salaried)
```

<table>
<thead>
<tr>
<th>desk</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>National</td>
<td>106.00</td>
<td>149520.50</td>
</tr>
<tr>
<td>Foreign</td>
<td>25.00</td>
<td>135000.00</td>
</tr>
<tr>
<td>Financial</td>
<td>38.00</td>
<td>133509.94</td>
</tr>
<tr>
<td>Style</td>
<td>45.00</td>
<td>107170.81</td>
</tr>
<tr>
<td>Local</td>
<td>65.00</td>
<td>105780.00</td>
</tr>
<tr>
<td>Editorial</td>
<td>33.00</td>
<td>105000.00</td>
</tr>
<tr>
<td>Graphics</td>
<td>15.00</td>
<td>100780.00</td>
</tr>
<tr>
<td>Universal Desk</td>
<td>8.00</td>
<td>100444.28</td>
</tr>
<tr>
<td>Sports</td>
<td>37.00</td>
<td>100000.00</td>
</tr>
<tr>
<td>Outlook</td>
<td>6.00</td>
<td>99937.50</td>
</tr>
<tr>
<td>non-newsroom</td>
<td>29.00</td>
<td>95780.00</td>
</tr>
<tr>
<td>Audio</td>
<td>7.00</td>
<td>92000.00</td>
</tr>
<tr>
<td>Design</td>
<td>45.00</td>
<td>88065.25</td>
</tr>
<tr>
<td>Operations</td>
<td>6.00</td>
<td>87890.00</td>
</tr>
<tr>
<td>Multiplatform</td>
<td>26.00</td>
<td>86104.00</td>
</tr>
<tr>
<td>Video</td>
<td>46.00</td>
<td>84250.00</td>
</tr>
<tr>
<td>Emerging News Products</td>
<td>30.00</td>
<td>75000.00</td>
</tr>
</tbody>
</table>

```python
[168]: current_news_median_desk_hourly = news_hourly.groupby(['desk']).agg({'current_base_pay': [np.count_nonzero, np.median]}).supress_median(current_news_median_desk_hourly)
```

<table>
<thead>
<tr>
<th>desk</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audio</td>
<td>6.00</td>
<td>39.75</td>
</tr>
<tr>
<td>Universal Desk</td>
<td>8.00</td>
<td>38.67</td>
</tr>
<tr>
<td>non-newsroom</td>
<td>7.00</td>
<td>37.58</td>
</tr>
</tbody>
</table>
Multiplatform | 16.00 | 34.09
Editorial     |  5.00 | 32.31
National      | 12.00 | 31.74
Local         |  5.00 | 26.46
Style         |  9.00 | 21.77
Sports        | 11.00 | 20.91
Operations    |  7.00 | 15.59

[169]:
current_news_median_desk_gender_salaried = news_salaried.
        .groupby(['desk', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_news_median_desk_gender_salaried)

[169]:
count_nonzero  median
desk              gender
National       Male      57.00  169780.00
Foreign        Male      14.00  145390.00
Editorial      Male      18.00  140271.26
National       Female    49.00  139780.00
Financial      Male      25.00  136467.50
Foreign        Female    11.00  129970.48
Financial      Female    13.00  125000.00
Local          Male      31.00  118850.00
Style          Male      20.00  115036.81
Sports         Female    9.00   115000.00
Graphics       Male      8.00   106925.50
Style          Female    25.00  106602.62
non-newsroom  Male      16.00  102890.00
Local          Female    34.00  100390.00
Sports         Male      28.00   99862.50
Editorial      Female    15.00   98405.45
Universal Desk Female    5.00   96944.47
Graphics       Female    7.00   95780.00
Design         Male      24.00   95211.85
non-newsroom  Female    13.00   95000.00
Audio          Female    5.00    92000.00
Multiplatform  Male      11.00  88151.74
Video          Male      18.00  88130.00
Operations     Female    5.00   85000.00
Multiplatform  Female    15.00   84780.00
Video          Female    28.00   79250.00
Design         Female    21.00   78641.52
Emerging News Products Female    21.00   77000.00
                Male      9.00    73172.23

[170]:
current_news_median_desk_gender_hourly = news_hourly.groupby(['desk', 'gender']).
        .agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_news_median_desk_gender_hourly)
```python
[170]:
count_nonzero median
desk gender
Audio Female 5.00 41.03
Universal Desk Female 5.00 35.90
Multiplatform Female 13.00 34.72
Sports Male 8.00 32.97
National Female 8.00 32.71
Style Female 8.00 26.73

[171]:
current_news_median_desk_race_salaried = news_salaried.
    .groupby(["desk", "race_grouping"]).agg("
current_base_pay": [np.
    .count_nonzero, np.median])
suppress_median(current_news_median_desk_race_salaried)

[171]:
count_nonzero median
desk race_grouping
National white 84.00 168780.00
Foreign unknown 20.00 137500.00
Financial white 29.00 136467.50
National person of color 21.00 130780.00
Editorial white 27.00 120000.27
Financial person of color 6.00 115570.00
Style white 38.00 112371.03
Local white 46.00 107707.84
Sports person of color 7.00 105000.00
Universal Desk white 5.00 104393.45
non-newsroom white 22.00 101390.00
Graphics white 9.00 100780.00
Sports white 30.00 99862.50
Editorial person of color 5.00 98405.45
Graphics person of color 6.00 97280.00
Style person of color 7.00 96147.48
non-newsroom person of color 7.00 95780.00
Audio white 5.00 92000.00
Local person of color 19.00 91450.00
Design white 27.00 90280.00
Video white 28.00 88000.00
Multiplatform white 22.00 87289.87
Design person of color 17.00 82000.00
Video person of color 16.00 76390.00
Emerging News Products person of color 10.00 76000.00
white 20.00 75000.00

[172]:
current_news_median_desk_race_hourly = news_hourly.
    .groupby(["desk", "race_ethnicity"]).agg("
current_base_pay": [np.
    .count_nonzero, np.median])
suppress_median(current_news_median_desk_race_hourly)
```

93
count_nonzero  median
desk  race_ethnicity
Style  White (United States of America)  5.00  38.93
Universal Desk  White (United States of America)  6.00  38.67
Multiplatform  White (United States of America)  12.00  36.54
Sports  White (United States of America)  9.00  32.97
National  White (United States of America)  9.00  32.71

current_news_median_desk_race_gender_salaried = news_salaried.
groupby(['desk','race_ethnicity','gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_news_median_desk_race_gender_salaried)

count_nonzero  desk  race_ethnicity  gender
National  White (United States of America)  Male  46.00
Financial  White (United States of America)  Male  21.00
Editorial  White (United States of America)  Male  16.00
National  White (United States of America)  Female  38.00
Black or African American (United States of America)  Male  8.00
Asian (United States of America)  Female  8.00
Sports  White (United States of America)  Female  6.00
Financial  White (United States of America)  Female  8.00
Local  White (United States of America)  Male  25.00
non-newsroom  White (United States of America)  Male  12.00
Style  White (United States of America)  Male  18.00
Design  White (United States of America)  Male  20.00
Graphics  White (United States of America)  Male  12.00
Editorial  White (United States of America)  Female  5.00
Local  White (United States of America)  Female  11.00
Sports  White (United States of America)  Male  21.00
### Median Salary Data

<table>
<thead>
<tr>
<th>Desk</th>
<th>Race/Ethnicity</th>
<th>Gender</th>
</tr>
</thead>
<tbody>
<tr>
<td>National</td>
<td>White (United States of America)</td>
<td>Male</td>
</tr>
<tr>
<td>175374.24</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Financial</td>
<td>White (United States of America)</td>
<td>Male</td>
</tr>
<tr>
<td>140387.17</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Editorial</td>
<td>White (United States of America)</td>
<td>Male</td>
</tr>
<tr>
<td>140271.26</td>
<td></td>
<td></td>
</tr>
<tr>
<td>National</td>
<td>White (United States of America)</td>
<td>Female</td>
</tr>
<tr>
<td>139733.72</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Emerging News Products</td>
<td>White (United States of America)</td>
<td>Male</td>
</tr>
<tr>
<td>135390.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sports</td>
<td>White (United States of America)</td>
<td>Female</td>
</tr>
<tr>
<td>132780.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Financial</td>
<td>White (United States of America)</td>
<td>Female</td>
</tr>
<tr>
<td>132014.99</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Local</td>
<td>White (United States of America)</td>
<td>Male</td>
</tr>
<tr>
<td>119553.20</td>
<td></td>
<td></td>
</tr>
<tr>
<td>non-newsroom</td>
<td>White (United States of America)</td>
<td>Male</td>
</tr>
<tr>
<td>115640.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Style</td>
<td>White (United States of America)</td>
<td>Male</td>
</tr>
<tr>
<td>115036.81</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Black or African American (United States of America)</td>
<td>Male</td>
</tr>
<tr>
<td>24.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design</td>
<td>Black or African American (United States of America)</td>
<td>Male</td>
</tr>
<tr>
<td>6.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Video</td>
<td>White (United States of America)</td>
<td>Male</td>
</tr>
<tr>
<td>13.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Multiplatform</td>
<td>White (United States of America)</td>
<td>Male</td>
</tr>
<tr>
<td>9.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>non-newsroom</td>
<td>White (United States of America)</td>
<td>Female</td>
</tr>
<tr>
<td>10.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Video</td>
<td>White (United States of America)</td>
<td>Female</td>
</tr>
<tr>
<td>15.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Local</td>
<td>Hispanic or Latino (United States of America)</td>
<td>Female</td>
</tr>
<tr>
<td>6.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Multiplatform</td>
<td>White (United States of America)</td>
<td>Female</td>
</tr>
<tr>
<td>13.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Design</td>
<td>White (United States of America)</td>
<td>Female</td>
</tr>
<tr>
<td>15.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Emerging News Products</td>
<td>White (United States of America)</td>
<td>Male</td>
</tr>
<tr>
<td>7.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Design  White (United States of America)  Male  109841.38
Graphics  White (United States of America)  Male  103330.48
Editorial  White (United States of America)  Female  102780.00
Local  White (United States of America)  Female  100780.00
Sports  White (United States of America)  Male  98393.66
Design  Black or African American (United States of America)  Male  93910.58
Video  White (United States of America)  Male  90780.00
Multiplatform  White (United States of America)  Male  88151.74
non-newsroom  White (United States of America)  Female  86160.00
Video  White (United States of America)  Female  86000.00
Local  Hispanic or Latino (United States of America)  Female  85372.54
Multiplatform  White (United States of America)  Female  84780.00
Design  White (United States of America)  Female  79140.00
Emerging News Products  White (United States of America)  Male  75000.00
Emerging News Products  White (United States of America)  Female  75000.00

[174]: current_news_median_desk_race_gender_hourly = news_hourly.
groupby(['desk', 'race_ethnicity', 'gender']).agg({'current_base_pay': [np.
count_nonzero, np.median]})
suppress_median(current_news_median_desk_race_gender_hourly)

[174]:

count_nonzero median
desk race_ethnicity gender
Style  White (United States of America)  Female  5.00  38.93
Multiplatform  White (United States of America)  Female  9.00  38.36
Sports  White (United States of America)  Male  7.00  32.97
National  White (United States of America)  Female  6.00  32.71

[175]:
current_news_median_desk_race_group_gender_salaried = news_salaried.
groupby(['desk', 'race_grouping', 'gender']).agg({'current_base_pay': [np.
count_nonzero, np.median]})
suppress_median(current_news_median_desk_race_group_gender_salaried)
<table>
<thead>
<tr>
<th>desk</th>
<th>race_grouping</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>National</td>
<td>white</td>
<td>Male</td>
<td>46.00</td>
<td>175374.24</td>
</tr>
<tr>
<td>Financial</td>
<td>white</td>
<td>Male</td>
<td>21.00</td>
<td>140387.17</td>
</tr>
<tr>
<td>Editorial</td>
<td>white</td>
<td>Male</td>
<td>16.00</td>
<td>140271.26</td>
</tr>
<tr>
<td>Foreign</td>
<td>unknown</td>
<td>Male</td>
<td>11.00</td>
<td>140000.00</td>
</tr>
<tr>
<td>National</td>
<td>white</td>
<td>Female</td>
<td>38.00</td>
<td>139733.72</td>
</tr>
<tr>
<td>Foreign</td>
<td>unknown</td>
<td>Female</td>
<td>9.00</td>
<td>135000.00</td>
</tr>
<tr>
<td>National</td>
<td>person of color</td>
<td>Female</td>
<td>10.00</td>
<td>132780.00</td>
</tr>
<tr>
<td>Sports</td>
<td>white</td>
<td>Female</td>
<td>6.00</td>
<td>132014.99</td>
</tr>
<tr>
<td>National</td>
<td>person of color</td>
<td>Male</td>
<td>11.00</td>
<td>130780.00</td>
</tr>
<tr>
<td>Financial</td>
<td>white</td>
<td>Female</td>
<td>8.00</td>
<td>130390.00</td>
</tr>
<tr>
<td>person of color</td>
<td>Female</td>
<td>5.00</td>
<td>121140.00</td>
<td></td>
</tr>
<tr>
<td>Local</td>
<td>white</td>
<td>Male</td>
<td>25.00</td>
<td>119553.20</td>
</tr>
<tr>
<td>non-newsroom</td>
<td>white</td>
<td>Male</td>
<td>12.00</td>
<td>115640.00</td>
</tr>
<tr>
<td>Style</td>
<td>white</td>
<td>Male</td>
<td>18.00</td>
<td>115036.81</td>
</tr>
<tr>
<td>Editorial</td>
<td>white</td>
<td>Female</td>
<td>20.00</td>
<td>109841.38</td>
</tr>
<tr>
<td>Local</td>
<td>person of color</td>
<td>Male</td>
<td>6.00</td>
<td>101674.57</td>
</tr>
<tr>
<td>Style</td>
<td>person of color</td>
<td>Female</td>
<td>5.00</td>
<td>96147.48</td>
</tr>
<tr>
<td>Local</td>
<td>person of color</td>
<td>Male</td>
<td>11.00</td>
<td>90780.00</td>
</tr>
<tr>
<td>Sports</td>
<td>white</td>
<td>Female</td>
<td>21.00</td>
<td>100780.00</td>
</tr>
<tr>
<td>Multiplatform</td>
<td>white</td>
<td>Male</td>
<td>24.00</td>
<td>98393.66</td>
</tr>
<tr>
<td>non-newsroom</td>
<td>white</td>
<td>Female</td>
<td>10.00</td>
<td>86160.00</td>
</tr>
<tr>
<td>Video</td>
<td>white</td>
<td>Male</td>
<td>13.00</td>
<td>90780.00</td>
</tr>
<tr>
<td>Multiplatform</td>
<td>white</td>
<td>Female</td>
<td>15.00</td>
<td>86000.00</td>
</tr>
<tr>
<td>Design</td>
<td>white</td>
<td>Male</td>
<td>13.00</td>
<td>88151.74</td>
</tr>
<tr>
<td>Design</td>
<td>person of color</td>
<td>Female</td>
<td>8.00</td>
<td>79140.00</td>
</tr>
<tr>
<td>Emerging News Products</td>
<td>person of color</td>
<td>Female</td>
<td>8.00</td>
<td>78500.00</td>
</tr>
<tr>
<td>Video</td>
<td>person of color</td>
<td>Female</td>
<td>12.00</td>
<td>76390.00</td>
</tr>
<tr>
<td>Emerging News Products</td>
<td>white</td>
<td>Male</td>
<td>7.00</td>
<td>75000.00</td>
</tr>
<tr>
<td>Design</td>
<td>person of color</td>
<td>Female</td>
<td>6.00</td>
<td>73500.00</td>
</tr>
</tbody>
</table>

```python
[176]:
current_news_median_desk_race_group_gender_hourly = news_hourly.
current_news_median_desk_race_group_gender_hourly = current_news_median_desk_race_group_gender_hourly.groupby(['desk', 'race_grouping', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_news_median_desk_race_group_gender_hourly)
```

```python
[176]:
count_nonzero median
desk          race_grouping gender
Style         white Female  5.00  38.93
Multiplatform white Female  9.00  38.36
Sports        white Male    7.00  32.97
```
Current news median desk race gender age5 salaried = news_salaried.
  groupby(['desk', 'race_ethnicity', 'gender', 'age_group_5']).
  agg({'current_base_pay': [np.count_nonzero, np.median]})

suppress_median(current_news_median_desk_race_gender_age5_salaried)

count_nonzero \\
<table>
<thead>
<tr>
<th>desk</th>
<th>race_ethnicity</th>
<th>gender</th>
<th>age_group_5</th>
</tr>
</thead>
<tbody>
<tr>
<td>National</td>
<td>White (United States of America) Male</td>
<td>40-44</td>
<td>30-34</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>9.00</td>
<td>Female 50-54</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5.00</td>
<td>55-59</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6.00</td>
<td>40-44</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5.00</td>
<td>Male 35-39</td>
</tr>
<tr>
<td></td>
<td>Sports</td>
<td>White (United States of America) Male</td>
<td>35-39</td>
</tr>
<tr>
<td></td>
<td>7.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Financial</td>
<td>White (United States of America) Male</td>
<td>35-39</td>
</tr>
<tr>
<td></td>
<td>5.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Local</td>
<td>White (United States of America) Male</td>
<td>55-59</td>
</tr>
<tr>
<td></td>
<td>6.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>National</td>
<td>White (United States of America) Female</td>
<td>25-29</td>
</tr>
<tr>
<td></td>
<td>5.00</td>
<td></td>
<td>35-39</td>
</tr>
<tr>
<td></td>
<td>6.00</td>
<td></td>
<td>Female 30-34</td>
</tr>
<tr>
<td></td>
<td>Video</td>
<td>White (United States of America) Female</td>
<td>30-34</td>
</tr>
<tr>
<td></td>
<td>5.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Sports</td>
<td>White (United States of America) Male</td>
<td>45-49</td>
</tr>
<tr>
<td></td>
<td>5.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Emerging News Products</td>
<td>White (United States of America) Female</td>
<td>25-29</td>
</tr>
<tr>
<td></td>
<td>7.00</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

median desk race_ethnicity gender age_group_5

<table>
<thead>
<tr>
<th>National</th>
<th>White (United States of America) Male</th>
<th>40-44</th>
</tr>
</thead>
<tbody>
<tr>
<td>170000.00</td>
<td></td>
<td>30-34</td>
</tr>
<tr>
<td>169780.00</td>
<td></td>
<td>Female 50-54</td>
</tr>
<tr>
<td>167780.00</td>
<td></td>
<td>55-59</td>
</tr>
<tr>
<td>162854.23</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
40-44
160000.00
Male 35-39
148640.00
Sports White (United States of America) Male 35-39
147300.00
Financial White (United States of America) Male 35-39
144755.00
Local White (United States of America) Male 55-59
127654.56
National White (United States of America) Male 35-39
125000.00

35-39
109390.00
Video White (United States of America) Female 30-34
88000.00
Sports White (United States of America) Male 45-49
87277.77
Emerging News Products White (United States of America) Female 25-29
70000.00

[178]: current_news_median_desk_race_gender_age5_hourly = news_hourly.
    → groupby(['desk', 'race_ethnicity', 'gender', 'age_group_5']).
    → agg({'current_base_pay': [np.count_nonzero, np.median]})
    suppress_median(current_news_median_desk_race_gender_age5_hourly)

[178]: Empty DataFrame
Columns: [count_nonzero, median]
Index: []

[179]: current_news_median_desk_race_group_gender_age5_salaried = news_salaried.
    → groupby(['desk', 'race_grouping', 'gender', 'age_group_5']).
    → agg({'current_base_pay': [np.count_nonzero, np.median]})
    suppress_median(current_news_median_desk_race_group_gender_age5_salaried)

[179]:
count_nonzero
desk           race_grouping gender age_group_5
National       white    Male  40-44  9.00
              Male  30-34  9.00
              Female  50-54  5.00
              Female  55-59  6.00
              Male  40-44  5.00
              Male  35-39 10.00
Sports         white    Male  35-39  7.00
Financial      white    Male  35-39  5.00
Local          white    Male  55-59  6.00
Foreign        unknown Male  30-34  5.00
National       white    Female 25-29  5.00
              Female  35-39  6.00
Video  white  Female 30-34  5.00  
Sports  white  Male 45-49  5.00  
Video  person of color  Female 25-29  8.00  
Emerging News Products  white  Female 25-29  7.00

desk  race_grouping  gender  age_group_5  median
National  white  Male  40-44  170000.00  
30-34  169780.00  
Female  50-54  167780.00  
55-59  162854.23  
40-44  160000.00  
Male  35-39  148640.00  
Sports  white  Male  35-39  147300.00  
Financial  white  Male  35-39  144755.00  
Local  white  Male  55-59  127654.56  
Foreign  unknown  Male  30-34  125000.00  
National  white  Female  25-29  125000.00  
35-39  109390.00  
Video  white  Female  30-34  88000.00  
Sports  white  Male  45-49  87277.77  
Video  person of color  Female  25-29  76390.00  
Emerging News Products  white  Female  25-29  70000.00

[180]: current_news_medianDesk_race_group_gender_age5_hourly = news_hourly.
    .groupby(['desk','race_grouping','gender','age_group_5']).
    .agg({'current_base_pay': [np.count_nonzero, np.median]})
    suppress_median(current_news_medianDesk_race_group_gender_age5_hourly)

[180]: Empty DataFrame
Columns: [count_nonzero, median]
Index: []

[181]: current_news_medianDesk_tier_salaried = news_salaried.groupby(['tier']).
    .agg({'current_base_pay': [np.count_nonzero, np.median]})
    suppress_median(current_news_medianDesk_tier_salaried)

[181]:
count_nonzero  median
tier
Tier 1  169.00  140387.17
Tier 2  209.00  105000.00
other  29.00  95780.00
Tier 3  131.00  86000.00
Tier 4  36.00  75000.00

[182]: current_news_medianDesk_tier_gender_salaried = news_salaried.
    .groupby(['tier','gender']).agg({'current_base_pay': [np.count_nonzero, np.
    median]})
suppress_median(current_news_medianDesk_tier_gender_salaried)
<table>
<thead>
<tr>
<th>Tier</th>
<th>Male Gender</th>
<th>Count Nonzero</th>
<th>Median Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tier 1</td>
<td>Male</td>
<td>96.00</td>
<td>152115.94</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>73.00</td>
<td>135320.05</td>
</tr>
<tr>
<td>Tier 2</td>
<td>Male</td>
<td>112.00</td>
<td>112755.06</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>16.00</td>
<td>102890.00</td>
</tr>
<tr>
<td>Tier 2</td>
<td>Female</td>
<td>97.00</td>
<td>99251.60</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>13.00</td>
<td>95000.00</td>
</tr>
<tr>
<td>Tier 3</td>
<td>Male</td>
<td>56.00</td>
<td>90780.00</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>75.00</td>
<td>81999.88</td>
</tr>
<tr>
<td>Tier 4</td>
<td>Female</td>
<td>26.00</td>
<td>75000.00</td>
</tr>
</tbody>
</table>

```python
import numpy as np

current_news_median_desk_tier_race_salaried = news_salaried.groupby(['tier', 'race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_news_median_desk_tier_race_salaried)
```

<table>
<thead>
<tr>
<th>Tier</th>
<th>Male Race Ethnicity</th>
<th>Count Nonzero</th>
<th>Median Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tier 1 White</td>
<td></td>
<td>116.00</td>
<td>159150.00</td>
</tr>
<tr>
<td></td>
<td>Black or African American</td>
<td>11.00</td>
<td>140000.00</td>
</tr>
<tr>
<td></td>
<td>Asian (United States of America)</td>
<td>15.00</td>
<td>125780.00</td>
</tr>
<tr>
<td>Tier 2 White</td>
<td></td>
<td>159.00</td>
<td>107170.81</td>
</tr>
<tr>
<td></td>
<td>Black or African American (United States of America)</td>
<td>16.00</td>
<td>101702.73</td>
</tr>
<tr>
<td>other White</td>
<td></td>
<td>22.00</td>
<td></td>
</tr>
<tr>
<td>Tier 2 Asian</td>
<td></td>
<td>14.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Hispanic or Latino (United States of America)</td>
<td>11.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Two or More Races (United States of America)</td>
<td>6.00</td>
<td></td>
</tr>
<tr>
<td>Tier 3 White</td>
<td></td>
<td>86.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Black or African American (United States of America)</td>
<td>16.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Hispanic or Latino (United States of America)</td>
<td>10.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Asian (United States of America)</td>
<td>12.00</td>
<td></td>
</tr>
<tr>
<td>Tier 4 White</td>
<td></td>
<td>23.00</td>
<td></td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>81249.94</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Asian (United States of America)</td>
<td>75500.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tier 4 White (United States of America)</td>
<td>75000.00</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```python
[184]: current_news_median_desk_tier_race_gender_salaried = news_salaried.
    ↬ groupby(['tier', 'race_ethnicity', 'gender']).agg({'
    ↬ current_base_pay': [np.
    ↬ count_nonzero, np.median]})}}

suppress_median(current_news_median_desk_tier_race_gender_salaried)
```

<table>
<thead>
<tr>
<th>tier race_ethnicity</th>
<th>gender</th>
<th>count_nonzero</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tier 1 White (United States of America)</td>
<td>Male</td>
<td>68.00</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>48.00</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>Male</td>
<td>8.00</td>
</tr>
<tr>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>10.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>5.00</td>
</tr>
<tr>
<td>Tier 2 White (United States of America)</td>
<td>Male</td>
<td>93.00</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>Male</td>
<td>5.00</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>Male</td>
<td>7.00</td>
</tr>
<tr>
<td>other White (United States of America)</td>
<td>Male</td>
<td>12.00</td>
</tr>
<tr>
<td>Tier 2 White (United States of America)</td>
<td>Female</td>
<td>66.00</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>9.00</td>
</tr>
<tr>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>10.00</td>
</tr>
<tr>
<td>Tier 3 Black or African American (United States of America)</td>
<td>Male</td>
<td>7.00</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>Male</td>
<td>37.00</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>Male</td>
<td>6.00</td>
</tr>
<tr>
<td>other White (United States of America)</td>
<td>Female</td>
<td>10.00</td>
</tr>
<tr>
<td>Tier 3 White (United States of America)</td>
<td>Female</td>
<td>49.00</td>
</tr>
<tr>
<td>Tier 2 Hispanic or Latino (United States of America)</td>
<td>Female</td>
<td>6.00</td>
</tr>
<tr>
<td>Tier 3 Black or African American (United States of America)</td>
<td>Female</td>
<td>9.00</td>
</tr>
<tr>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>10.00</td>
</tr>
<tr>
<td>Tier 4 White (United States of America)</td>
<td>Male</td>
<td>8.00</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>15.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>tier race_ethnicity</th>
<th>gender</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tier 1 White (United States of America)</td>
<td>Male</td>
<td>169870.29</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>135824.85</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>Male</td>
<td>135390.00</td>
</tr>
<tr>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>128430.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>125000.00</td>
</tr>
<tr>
<td>Tier 2 White (United States of America)</td>
<td>Male</td>
<td>117843.50</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>Male</td>
<td>117780.00</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>Male</td>
<td>116349.15</td>
</tr>
<tr>
<td>other White (United States of America)</td>
<td>Male</td>
<td>115640.00</td>
</tr>
<tr>
<td>Tier 2 White (United States of America)</td>
<td>Female</td>
<td>102423.86</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>96147.48</td>
</tr>
</tbody>
</table>

102
<table>
<thead>
<tr>
<th>Race/Ethnicity</th>
<th>Gender</th>
<th>Median Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>93835.10</td>
</tr>
<tr>
<td>Tier 3 Black or African American (United States of America)</td>
<td>Male</td>
<td>93177.48</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>Male</td>
<td>92980.00</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>Male</td>
<td>90390.04</td>
</tr>
<tr>
<td>Other White (United States of America)</td>
<td>Female</td>
<td>86160.00</td>
</tr>
<tr>
<td>Tier 3 White (United States of America)</td>
<td>Female</td>
<td>85780.00</td>
</tr>
<tr>
<td>Tier 2 Hispanic or Latino (United States of America)</td>
<td>Female</td>
<td>85372.54</td>
</tr>
<tr>
<td>Tier 3 Black or African American (United States of America)</td>
<td>Female</td>
<td>85000.00</td>
</tr>
<tr>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>75500.00</td>
</tr>
<tr>
<td>Tier 4 White (United States of America)</td>
<td>Male</td>
<td>75500.00</td>
</tr>
<tr>
<td>Tier 4 White (United States of America)</td>
<td>Female</td>
<td>75000.00</td>
</tr>
</tbody>
</table>

```python
[185]:
current_news_median_desk_tier_race_group_gender_salaried = news_salaried.
      groupby(['tier', 'race_grouping', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_news_median_desk_tier_race_group_gender_salaried)
```

```python
[185]:
tier race_grouping gender count_nonzero median
Tier 1 white Male 68.00 169870.29
    unknown Male 14.00 137890.00
    white Female 10.00 137640.00
    person of color Male 14.00 135390.00
        Female 15.00 125780.00
Tier 2 white Male 93.00 117843.50
    other white Male 12.00 115640.00
    person of color Male 19.00 105000.00
        white Female 66.00 102423.86
        person of color Female 30.00 93020.07
Tier 3 white Male 37.00 92980.00
    other white Female 10.00 86160.00
Tier 3 White (United States of America) Male 60-64 5.00
    40-44 13.00
    Female 45-49 5.00
```

```python
[186]:
current_news_median_desk_tier_race_gender_age5_salaried = news_salaried.
      groupby(['tier', 'race_ethnicity', 'gender', 'age_group_5']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_news_median_desk_tier_race_gender_age5_salaried)
```

```python
[186]:
tier race_ethnicity gender age_group_5 count_nonzero median
Tier 1 White (United States of America) Male 60-64 5.00
    40-44 13.00
    Female 45-49 5.00
```
| Tier 1 White (United States of America) | Male 35-39 | 15.00 |
| Tier 1 White (United States of America) | Female 50-54 | 7.00 |
| Tier 2 White (United States of America) | Male 30-34 | 13.00 |

| Tier 2 White (United States of America) | Male 50-54 | 15.00 |
| Tier 2 White (United States of America) | Female 60-64 | 6.00 |

| Tier 1 White (United States of America) | Female 35-39 | 9.00 |
| Tier 2 White (United States of America) | Male 60-64 | 6.00 |

| Tier 1 White (United States of America) | Female 25-29 | 6.00 |
| Tier 2 White (United States of America) | Female 50-54 | 10.00 |
| Tier 3 White (United States of America) | Male 40-44 | 6.00 |
| Tier 2 White (United States of America) | Male 40-44 | 11.00 |
| Tier 3 White (United States of America) | Female 50-54 | 5.00 |
| Tier 2 White (United States of America) | Male 45-49 | 10.00 |
| Tier 2 White (United States of America) | Female 30-34 | 11.00 |
| Tier 3 White (United States of America) | Male 30-34 | 7.00 |
| Tier 3 White (United States of America) | Female 35-39 | 9.00 |
| Tier 3 White (United States of America) | Male 30-34 | 7.00 |
| Tier 3 White (United States of America) | Female 30-34 | 10.00 |
| Tier 3 White (United States of America) | Male 55-59 | 5.00 |
| Tier 3 White (United States of America) | Male 25-29 | 6.00 |
| Tier 4 White (United States of America) | Female 25-29 | 6.00 |
| Tier 4 White (United States of America) | Female 25-29 | 11.00 |
| Tier 4 White (United States of America) | Female 25-29 | 8.00 |

<p>| Tier 1 White (United States of America) | Male 60-64 | 174968.48 |
| Tier 1 White (United States of America) | Female 45-49 | 170000.00 |
| Tier 1 White (United States of America) | Male 55-59 | 162890.00 |
| Tier 1 White (United States of America) | Female 55-59 | 162854.23 |
| Tier 2 White (United States of America) | Female 55-59 | 149029.98 |
| Tier 2 White (United States of America) | Male 65+ | 147473.21 |</p>
<table>
<thead>
<tr>
<th>Tier 1 White (United States of America)</th>
<th>Male 35-39</th>
<th>144755.00</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Female 50-54</td>
<td>134780.00</td>
</tr>
<tr>
<td></td>
<td>Male 30-34</td>
<td>128780.00</td>
</tr>
<tr>
<td>Tier 2 White (United States of America)</td>
<td>Male 50-54</td>
<td>128052.85</td>
</tr>
<tr>
<td></td>
<td>35-39</td>
<td>124120.00</td>
</tr>
<tr>
<td>Tier 1 White (United States of America)</td>
<td>Female 25-29</td>
<td>112500.00</td>
</tr>
<tr>
<td>Tier 2 White (United States of America)</td>
<td>Male 60-64</td>
<td>115891.66</td>
</tr>
<tr>
<td>Tier 3 White (United States of America)</td>
<td>Male 40-44</td>
<td>107773.24</td>
</tr>
<tr>
<td>Tier 2 White (United States of America)</td>
<td>Male 40-44</td>
<td>105000.00</td>
</tr>
<tr>
<td>Tier 3 White (United States of America)</td>
<td>Male 45-49</td>
<td>101260.30</td>
</tr>
<tr>
<td>Tier 2 White (United States of America)</td>
<td>Male 45-49</td>
<td>99280.00</td>
</tr>
<tr>
<td></td>
<td>Female 30-34</td>
<td>100780.00</td>
</tr>
<tr>
<td>Tier 2 White (United States of America)</td>
<td>Female 40-44</td>
<td>105000.00</td>
</tr>
<tr>
<td>Tier 3 White (United States of America)</td>
<td>Female 35-39</td>
<td>103990.60</td>
</tr>
<tr>
<td>Tier 2 White (United States of America)</td>
<td>Male 30-34</td>
<td>86000.00</td>
</tr>
<tr>
<td></td>
<td>Female 30-34</td>
<td>84750.00</td>
</tr>
<tr>
<td></td>
<td>55-59</td>
<td>81108.52</td>
</tr>
<tr>
<td>Tier 3 White (United States of America)</td>
<td>Male 55-59</td>
<td>93030.00</td>
</tr>
<tr>
<td>Tier 2 White (United States of America)</td>
<td>Male 25-29</td>
<td>91282.50</td>
</tr>
<tr>
<td>Tier 3 White (United States of America)</td>
<td>Female 25-29</td>
<td>91000.00</td>
</tr>
<tr>
<td>Tier 4 White (United States of America)</td>
<td>Female 25-29</td>
<td>74780.00</td>
</tr>
<tr>
<td>Tier 4 White (United States of America)</td>
<td>Female 25-29</td>
<td>69890.00</td>
</tr>
</tbody>
</table>

### Code

```python
[187]: current_news_median_desk_tier_race_group_gender_age5_salaried = news_salaried.
          ↪ groupby(['tier', 'race_grouping', 'gender', 'age_group_5']).
          ↪ agg({'current_base_pay': [np.count_nonzero, np.median]})
          suppress_median(current_news_median_desk_tier_race_group_gender_age5_salaried)
```

```plaintext
<table>
<thead>
<tr>
<th>tier</th>
<th>race_grouping</th>
<th>gender</th>
<th>age_group_5</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tier 1</td>
<td>white</td>
<td>Male</td>
<td>60-64</td>
<td>5.00</td>
<td>174968.48</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>40-44</td>
<td>13.00</td>
<td>170000.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Female</td>
<td>45-49</td>
<td>5.00</td>
<td>165000.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>55-59</td>
<td>8.00</td>
<td>162890.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Female</td>
<td>55-59</td>
<td>6.00</td>
<td>162854.23</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>40-44</td>
<td>5.00</td>
<td>160000.00</td>
</tr>
<tr>
<td>Tier 2</td>
<td>white</td>
<td>Female</td>
<td>55-59</td>
<td>5.00</td>
<td>149029.98</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>65+</td>
<td>6.00</td>
<td>147473.21</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>55-59</td>
<td>16.00</td>
<td>147160.79</td>
</tr>
</tbody>
</table>
```
<table>
<thead>
<tr>
<th>Tier 1 white</th>
<th>Male 35-39</th>
<th>15.00</th>
<th>144755.00</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Female 50-54</td>
<td>7.00</td>
<td>134780.00</td>
</tr>
<tr>
<td></td>
<td>Male 30-34</td>
<td>13.00</td>
<td>128780.00</td>
</tr>
<tr>
<td>Tier 2 white</td>
<td>Male 50-54</td>
<td>15.00</td>
<td>128052.85</td>
</tr>
<tr>
<td>Tier 1 person of color</td>
<td>Male 35-39</td>
<td>5.00</td>
<td>125000.00</td>
</tr>
<tr>
<td>Tier 2 white</td>
<td>Male 35-39</td>
<td>13.00</td>
<td>124120.00</td>
</tr>
<tr>
<td></td>
<td>Female 60-64</td>
<td>6.00</td>
<td>121896.57</td>
</tr>
<tr>
<td>Tier 1 white</td>
<td>Female 35-39</td>
<td>9.00</td>
<td>120880.00</td>
</tr>
<tr>
<td>unknown</td>
<td>Male 30-34</td>
<td>6.00</td>
<td>120390.00</td>
</tr>
<tr>
<td>Tier 2 person of color</td>
<td>Male 50-54</td>
<td>8.00</td>
<td>118932.52</td>
</tr>
<tr>
<td></td>
<td>Female 60-64</td>
<td>6.00</td>
<td>115891.66</td>
</tr>
<tr>
<td>Tier 1 white</td>
<td>Female 25-29</td>
<td>6.00</td>
<td>112500.00</td>
</tr>
<tr>
<td>Tier 2 white</td>
<td>Female 50-54</td>
<td>10.00</td>
<td>107773.24</td>
</tr>
<tr>
<td>Tier 3 white</td>
<td>Male 40-44</td>
<td>6.00</td>
<td>106500.00</td>
</tr>
<tr>
<td>Tier 2 white</td>
<td>Male 40-44</td>
<td>11.00</td>
<td>105000.00</td>
</tr>
<tr>
<td>Tier 3 white</td>
<td>Female 50-54</td>
<td>5.00</td>
<td>103990.60</td>
</tr>
<tr>
<td>Tier 2 white</td>
<td>Male 45-49</td>
<td>10.00</td>
<td>101260.30</td>
</tr>
<tr>
<td></td>
<td>Female 30-34</td>
<td>11.00</td>
<td>100780.00</td>
</tr>
<tr>
<td>Tier 3 person of color</td>
<td>Male 50-54</td>
<td>5.00</td>
<td>99931.09</td>
</tr>
<tr>
<td>Tier 2 white</td>
<td>Female 45-49</td>
<td>9.00</td>
<td>99280.00</td>
</tr>
<tr>
<td></td>
<td>Male 30-34</td>
<td>7.00</td>
<td>95655.73</td>
</tr>
<tr>
<td></td>
<td>Female 35-39</td>
<td>9.00</td>
<td>95000.00</td>
</tr>
<tr>
<td>Tier 3 white</td>
<td>Male 35-39</td>
<td>6.00</td>
<td>93030.00</td>
</tr>
<tr>
<td>Tier 2 person of color</td>
<td>Female 35-39</td>
<td>5.00</td>
<td>91450.00</td>
</tr>
<tr>
<td></td>
<td>Male 25-29</td>
<td>6.00</td>
<td>91282.50</td>
</tr>
<tr>
<td></td>
<td>Female 25-29</td>
<td>9.00</td>
<td>91000.00</td>
</tr>
<tr>
<td>Tier 3 white</td>
<td>Female 35-39</td>
<td>7.00</td>
<td>90780.00</td>
</tr>
<tr>
<td></td>
<td>45-49</td>
<td>5.00</td>
<td>90780.00</td>
</tr>
<tr>
<td>Tier 2 person of color</td>
<td>Female 30-34</td>
<td>8.00</td>
<td>87548.85</td>
</tr>
<tr>
<td>Tier 3 white</td>
<td>Male 30-34</td>
<td>7.00</td>
<td>86000.00</td>
</tr>
<tr>
<td></td>
<td>Female 30-34</td>
<td>10.00</td>
<td>84750.00</td>
</tr>
<tr>
<td></td>
<td>person of color Female 30-34</td>
<td>7.00</td>
<td>81999.88</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>5.00</td>
<td>81108.52</td>
</tr>
<tr>
<td></td>
<td>Female 55-59</td>
<td>5.00</td>
<td>80250.00</td>
</tr>
<tr>
<td></td>
<td>Male 25-29</td>
<td>6.00</td>
<td>80250.00</td>
</tr>
<tr>
<td></td>
<td>person of color Female 25-29</td>
<td>11.00</td>
<td>77000.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>11.00</td>
<td>74780.00</td>
</tr>
<tr>
<td>Tier 4 white</td>
<td>Female 25-29</td>
<td>8.00</td>
<td>69890.00</td>
</tr>
</tbody>
</table>

### 1.5.7 Job profiles

```python
[188]: current_news_median_job_salaried = news_salaried.

    → groupby(['job_profile_current']).agg({'current_base_pay': [np.count_nonzero,
    np.median]})

[188]: suppress_median(current_news_median_job_salaried)
```

```python
[188]:

    current_base_pay

    count_nonzero  median

    job_profile_current
```

106
<table>
<thead>
<tr>
<th>Job Profile Current</th>
<th>Hourly Pay</th>
<th>Median Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>Columnist</td>
<td>19.00</td>
<td>170496.80</td>
</tr>
<tr>
<td>Columnist - Editorial</td>
<td>7.00</td>
<td>151896.27</td>
</tr>
<tr>
<td>Critic</td>
<td>9.00</td>
<td>150962.35</td>
</tr>
<tr>
<td>Editorial Writer</td>
<td>7.00</td>
<td>129236.03</td>
</tr>
<tr>
<td>Staff Writer</td>
<td>306.00</td>
<td>124040.00</td>
</tr>
<tr>
<td>Graphics Editor</td>
<td>7.00</td>
<td>111071.00</td>
</tr>
<tr>
<td>Operations Editor</td>
<td>7.00</td>
<td>90780.00</td>
</tr>
<tr>
<td>Video Journalist</td>
<td>20.00</td>
<td>89240.00</td>
</tr>
<tr>
<td>Video Graphics Editor</td>
<td>8.00</td>
<td>87280.00</td>
</tr>
<tr>
<td>Assistant Editor</td>
<td>23.00</td>
<td>87000.00</td>
</tr>
<tr>
<td>Multiplatform Editor</td>
<td>53.00</td>
<td>83146.67</td>
</tr>
<tr>
<td>Designer</td>
<td>29.00</td>
<td>76000.00</td>
</tr>
<tr>
<td>Photo Editor</td>
<td>8.00</td>
<td>74961.88</td>
</tr>
<tr>
<td>Digital Video Editor</td>
<td>22.00</td>
<td>74500.00</td>
</tr>
<tr>
<td>News Intern - 2 Year</td>
<td>5.00</td>
<td>65780.00</td>
</tr>
</tbody>
</table>

[189]: current_news_median_job_hourly = news_hourly.groupby(['job_profile_current']).agg({'current_base_pay': [np.count_nonzero, np.median]}).suppress_median

<table>
<thead>
<tr>
<th>Job Profile Current</th>
<th>Count Nonzero</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>280225 - Producer</td>
<td>18.00</td>
<td>36.74</td>
</tr>
<tr>
<td>400151 - Administrative Aide</td>
<td>6.00</td>
<td>35.30</td>
</tr>
<tr>
<td>397110 - Multiplatform Editor (PT/PTOC)</td>
<td>23.00</td>
<td>34.72</td>
</tr>
<tr>
<td>380117 - Research Assistant</td>
<td>6.00</td>
<td>31.23</td>
</tr>
<tr>
<td>410251 - Editorial Aide</td>
<td>12.00</td>
<td>21.45</td>
</tr>
<tr>
<td>430117 - News Aide</td>
<td>8.00</td>
<td>17.06</td>
</tr>
<tr>
<td>440116 - Copy Aide</td>
<td>5.00</td>
<td>15.19</td>
</tr>
</tbody>
</table>

[190]: current_news_median_job_gender_salaried = news_salaried.groupby(['job_profile_current', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]}).suppress_median

<table>
<thead>
<tr>
<th>Job Profile Current</th>
<th>Count Nonzero</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>300113 - Columnist</td>
<td>8.00</td>
<td>175984.43</td>
</tr>
<tr>
<td>330113 - Editorial Writer</td>
<td>5.00</td>
<td>164899.53</td>
</tr>
<tr>
<td>320113 - Critic</td>
<td>5.00</td>
<td>160780.00</td>
</tr>
<tr>
<td>300113 - Columnist</td>
<td>11.00</td>
<td>154780.00</td>
</tr>
<tr>
<td>300313 - Columnist - Editorial</td>
<td>5.00</td>
<td>151896.27</td>
</tr>
<tr>
<td>280212 - Staff Writer</td>
<td>170.00</td>
<td>128439.57</td>
</tr>
<tr>
<td>390510 - Graphics Editor</td>
<td>5.00</td>
<td>111071.00</td>
</tr>
<tr>
<td>360114 - Photographer</td>
<td>11.00</td>
<td>109928.29</td>
</tr>
<tr>
<td>Job Title</td>
<td>Gender</td>
<td>Hourly Pay</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>--------</td>
<td>------------</td>
</tr>
<tr>
<td>Video Journalist</td>
<td>Male</td>
<td>8.00</td>
</tr>
<tr>
<td>Graphics Reporter</td>
<td>Female</td>
<td>5.00</td>
</tr>
<tr>
<td>Assistant Editor</td>
<td>Male</td>
<td>9.00</td>
</tr>
<tr>
<td>Operations Editor</td>
<td>Female</td>
<td>5.00</td>
</tr>
<tr>
<td>Designer</td>
<td>Male</td>
<td>11.00</td>
</tr>
<tr>
<td>Photographer</td>
<td>Female</td>
<td>5.00</td>
</tr>
<tr>
<td>Video Journalist</td>
<td>Female</td>
<td>12.00</td>
</tr>
<tr>
<td>Video Graphics Editor</td>
<td>Male</td>
<td>5.00</td>
</tr>
<tr>
<td>Assistant Editor</td>
<td>Female</td>
<td>14.00</td>
</tr>
<tr>
<td>Multiplatform Editor</td>
<td>Male</td>
<td>26.00</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>27.00</td>
</tr>
<tr>
<td>Digital Video Editor</td>
<td>Female</td>
<td>17.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>5.00</td>
</tr>
<tr>
<td>Designer</td>
<td>Female</td>
<td>18.00</td>
</tr>
</tbody>
</table>

[191]:
```
current_news_median_job_gender_hourly = news_hourly.
    → groupby(['job_profile_current', 'gender']).agg({'current_base_pay': [np.
    → count_nonzero, np.median]})
suppress_median(current_news_median_job_gender_hourly)
```

[191]:
```
count_nonzero median
job_profile_current  gender
280225 - Producer     Male  6.00  36.74
397110 - Multiplatform Editor (PT/PTOC) Female 14.00  36.54
280225 - Producer     Female 12.00  36.35
400151 - Administrative Aide Female 6.00  35.30
397110 - Multiplatform Editor (PT/PTOC) Male 9.00  33.41
380117 - Research Assistant Female 5.00  31.68
410251 - Editorial Aide Female 8.00  21.45
```

[192]:
```
current_news_median_job_race_salaried = news_salaried.
    → groupby(['job_profile_current', 'race_ethnicity']).agg({'current_base_pay': [np.
    → count_nonzero, np.median]})
suppress_median(current_news_median_job_race_salaried)
```

[192]:
```
count_nonzero race_ethnicity
job_profile_current
300313 - Columnist - Editorial White (United States of America) 6.00
300113 - Columnist White (United States of America) 13.00
Ame... Black or African American (United States of
5.00
320113 - Critic White (United States of America) 8.00
330113 - Editorial Writer White (United States of America) 6.00
280212 - Staff Writer White (United States of America) 223.00
```
Black or African American (United States of America)... 18.00 Asian (United States of America)
24.00
390510 - Graphics Editor White (United States of America)
5.00
360114 - Photographer White (United States of America)
12.00
280226 - Video Journalist White (United States of America)
13.00
120202 - Assistant Editor White (United States of America)
16.00
390310 - Video Graphics Editor White (United States of America)
6.00
390110 - Multiplatform Editor Black or African American (United States of America)
5.00
280212 - Staff Writer Hispanic or Latino (United States of America)
10.00
390110 - Multiplatform Editor White (United States of America)
42.00
280228 - Designer Hispanic or Latino (United States of America)
5.00
126202 - Photo Editor White (United States of America)
6.00
280228 - Designer White (United States of America)
16.00
390410 - Digital Video Editor White (United States of America)
10.00

median
job_profile_current race_ethnicity
300313 - Columnist - Editorial White (United States of America)
190948.14
300113 - Columnist White (United States of America)
176780.00
Black or African American (United States of America)
Ame... 153061.00
320113 - Critic White (United States of America)
149371.17
330113 - Editorial Writer White (United States of America)
127118.49
280212 - Staff Writer White (United States of America)
125000.00
Black or African American (United States of America)
Ame... 122340.98
116892.50
Asian (United States of America)
<table>
<thead>
<tr>
<th>Job Profile</th>
<th>Race/Ethnicity</th>
<th>Hourly Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>390510</td>
<td>Graphics Editor</td>
<td>111071.00</td>
</tr>
<tr>
<td>360114</td>
<td>Photographer</td>
<td>106014.84</td>
</tr>
<tr>
<td>280226</td>
<td>Video Journalist</td>
<td>103000.00</td>
</tr>
<tr>
<td>120202</td>
<td>Assistant Editor</td>
<td>91280.02</td>
</tr>
<tr>
<td>390310</td>
<td>Video Graphics Editor</td>
<td>89780.00</td>
</tr>
<tr>
<td>390110</td>
<td>Multiplatform Editor</td>
<td>85692.50</td>
</tr>
<tr>
<td>280212</td>
<td>Staff Writer</td>
<td>85372.54</td>
</tr>
<tr>
<td>390110</td>
<td>Multiplatform Editor</td>
<td>83398.19</td>
</tr>
<tr>
<td>280228</td>
<td>Designer</td>
<td>81999.88</td>
</tr>
<tr>
<td>126202</td>
<td>Photo Editor</td>
<td>77070.00</td>
</tr>
<tr>
<td>280228</td>
<td>Designer</td>
<td>75500.00</td>
</tr>
<tr>
<td>390410</td>
<td>Digital Video Editor</td>
<td>71240.00</td>
</tr>
</tbody>
</table>

```python
[193]:

```current_news_median_job_race_hourly = news_hourly.
  groupby(['job_profile_current', 'race_ethnicity']).agg({'current_base_pay': np.array([np.count_nonzero, np.median])})
suppress_median(current_news_median_job_race_hourly)
```
of Ame... 37.58
35.91
397110 - Multiplatform Editor (PT/PTOC) White (United States of America)
34.80
380117 - Research Assistant White (United States of America)
31.68
410251 - Editorial Aide White (United States of America)
21.12
430117 - News Aide White (United States of America)
16.50

```
[194]: current_news_median_job_race_gender_salaried = news_salaried.
    ↪ groupby(['job_profile_current','race_ethnicity','gender']).
    ↪ agg({'current_base_pay': [np.count_nonzero, np.median]})
    ↪ suppress_median(current_news_median_job_race_gender_salaried)
```

<table>
<thead>
<tr>
<th>job_profile_current</th>
<th>race_ethnicity</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Columnist</td>
<td>White (United States of America)</td>
<td>7.00</td>
<td>6.00</td>
</tr>
<tr>
<td>Critic</td>
<td>White (United States of America)</td>
<td>5.00</td>
<td></td>
</tr>
<tr>
<td>Staff Writer</td>
<td>White (United States of America)</td>
<td>130.00</td>
<td></td>
</tr>
<tr>
<td>Staff Writer</td>
<td>Black or African American (United States of America)</td>
<td>13.00</td>
<td></td>
</tr>
<tr>
<td>Staff Writer</td>
<td>Asian (United States of America)</td>
<td>9.00</td>
<td>15.00</td>
</tr>
<tr>
<td>Photographer</td>
<td>White (United States of America)</td>
<td>93.00</td>
<td></td>
</tr>
<tr>
<td>Staff Writer</td>
<td>Black or African American (United States of America)</td>
<td>7.00</td>
<td></td>
</tr>
<tr>
<td>Video Journalist</td>
<td>White (United States of America)</td>
<td>6.00</td>
<td></td>
</tr>
<tr>
<td>Assistant Editor</td>
<td>White (United States of America)</td>
<td>8.00</td>
<td></td>
</tr>
<tr>
<td>Designer</td>
<td>White (United States of America)</td>
<td>5.00</td>
<td></td>
</tr>
<tr>
<td>Photographer</td>
<td>White (United States of America)</td>
<td>5.00</td>
<td></td>
</tr>
<tr>
<td>Video Journalist</td>
<td>White (United States of America)</td>
<td>7.00</td>
<td></td>
</tr>
</tbody>
</table>
120202 - Assistant Editor White (United States of America)
   Female 8.00
390110 - Multiplatform Editor White (United States of America)
   Male 21.00
   Female 21.00
280212 - Staff Writer Hispanic or Latino (United States of America)
   Female 6.00
280228 - Designer White (United States of America)
   Female 11.00
390410 - Digital Video Editor White (United States of America)
   Female 7.00

median

job_profile_current   race_ethnicity
gender
300113 - Columnist White (United States of America)
   Female 224460.51
   Male 175984.43
320113 - Critic White (United States of America)
   Male 160780.00
280212 - Staff Writer White (United States of America)
   Male 129280.00
       Black or African American (United States of America)
       Male 125000.00
       Asian (United States of America)
       Male 118785.00
       Female 115000.00
   Female 115000.00
360114 - Photographer White (United States of America)
   Male 113756.68
280212 - Staff Writer Black or African American (United States of America)
   Female 108864.49
280226 - Video Journalist White (United States of America)
   Male 106500.00
120202 - Assistant Editor White (United States of America)
   Male 92528.23
280228 - Designer White (United States of America)
   Male 90280.00
360114 - Photographer White (United States of America)
   Female 88065.25
280226 - Video Journalist White (United States of America)
   Female 88000.00
120202 - Assistant Editor White (United States of America)
   Female 87890.02
390110 - Multiplatform Editor White (United States of America)
   Male 83649.71
Female  83146.67
280212 - Staff Writer  Hispanic or Latino (United States of America)
Female  82890.00
280228 - Designer  White (United States of America)
Female  72000.00
390410 - Digital Video Editor  White (United States of America)
Female  71500.00

[195]: current_news_median_job_race_gender_hourly = news_hourly.
groupby(['job_profile_current', 'race_ethnicity', 'gender']).
agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_news_median_job_race_gender_hourly)

[195]:

                   count_nonzero
job_profile_current race_ethnicity gender
397110 - Multiplatform Editor (PT/PTOC) White (United States of America) Female 10.00
280225 - Producer White (United States of America) Female 5.00
397110 - Multiplatform Editor (PT/PTOC) White (United States of America) Male 8.00
410251 - Editorial Aide White (United States of America) Female 5.00

                   median
job_profile_current race_ethnicity gender
397110 - Multiplatform Editor (PT/PTOC) White (United States of America) Female 39.87
280225 - Producer White (United States of America) Female 34.24
397110 - Multiplatform Editor (PT/PTOC) White (United States of America) Male 33.39
410251 - Editorial Aide White (United States of America) Female 21.12

[196]: current_news_median_job_race_group_gender_salaried = news_salaried.
groupby(['desk', 'race_grouping', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_news_median_job_race_group_gender_salaried)

[196]:

                      count_nonzero  median
desk  race_grouping  gender          
National  white  Male         46.00  175374.24
Financial white  Male         21.00  140387.17
Editorial white  Male         16.00  140271.26
Foreign  unknown  Male         11.00  140000.00
National  white  Female        38.00  139733.72
Foreign  unknown  Female        9.00  135000.00
National  person of color Female  10.00  132780.00
Current News Median Job Race Group Gender Hourly = news_hourly.

```python
groupby(['job_profile_current', 'race_grouping', 'gender']).
agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_news_median_job_race_group_gender_hourly)
```

<table>
<thead>
<tr>
<th>Job Profile Current</th>
<th>Race Grouping</th>
<th>Gender</th>
<th>Current Base Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multiplatform Editor (PT/PTOC) white</td>
<td>Female</td>
<td>39.87</td>
<td></td>
</tr>
<tr>
<td>Producer person of color</td>
<td>Female</td>
<td>35.90</td>
<td></td>
</tr>
<tr>
<td>Multiplatform Editor (PT/PTOC) white</td>
<td>Male</td>
<td>33.39</td>
<td></td>
</tr>
<tr>
<td>Editorial Aide white</td>
<td>Female</td>
<td>34.24</td>
<td></td>
</tr>
<tr>
<td>Producer person of color</td>
<td>Female</td>
<td>33.92</td>
<td></td>
</tr>
</tbody>
</table>
410251 - Editorial Aide  white  Female  21.12

```python
current_news_median_job_race_gender_age5_salaried = news_salaried.
.groupby(['job_profile_current', 'race_ethnicity', 'gender', 'age_group_5']).
.agg({'current_base_pay': [np.count_nonzero, np.median]})
supress_median(current_news_median_job_race_gender_age5_salaried)
```

<table>
<thead>
<tr>
<th>job_profile_current</th>
<th>race_ethnicity</th>
<th>gender</th>
<th>age_group_5</th>
<th>current_base_pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>280212 - Staff Writer</td>
<td>White (United States of America) Male  65+</td>
<td></td>
<td>55-59</td>
<td>17.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Female</td>
<td>55-59</td>
<td>7.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>45-49</td>
<td>10.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>40-44</td>
<td>9.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>60-64</td>
<td>11.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>40-44</td>
<td>20.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>50-54</td>
<td>14.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Female</td>
<td>60-64</td>
<td>9.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>45-49</td>
<td>6.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>35-39</td>
<td>25.00</td>
</tr>
<tr>
<td></td>
<td>Asian (United States of America) Female 30-34</td>
<td></td>
<td>5.00</td>
<td>11.00</td>
</tr>
<tr>
<td></td>
<td>White (United States of America) Female 50-54</td>
<td></td>
<td></td>
<td>21.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>30-34</td>
<td>17.00</td>
</tr>
<tr>
<td>390110 - Multiplatform Editor</td>
<td>White (United States of America) Female 50-54</td>
<td></td>
<td>5.00</td>
<td>17.00</td>
</tr>
<tr>
<td>280212 - Staff Writer</td>
<td>White (United States of America) Female 30-34</td>
<td></td>
<td>13.00</td>
<td>14.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Female</td>
<td>25-29</td>
<td>13.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>45-49</td>
<td>14.00</td>
</tr>
<tr>
<td>390110 - Multiplatform Editor</td>
<td>White (United States of America) Male 45-49</td>
<td></td>
<td>5.00</td>
<td>17.00</td>
</tr>
</tbody>
</table>
120202 - Assistant Editor  White (United States of America) Female 25-29 5.00
280212 - Staff Writer  White (United States of America) Male 25-29 6.00
390110 - Multiplatform Editor White (United States of America) Male 30-34 5.00
5.00
5.00
Female 25-29
5.00

median
job_profile_current  race_ethnicity  gender
age_group_5
280212 - Staff Writer  White (United States of America) Male 65+ 159458.37
55-59
153922.58
Female 55-59
153780.00
45-49
144559.75
40-44
140000.00
Male 60-64
134957.37
40-44
132980.42
50-54
132273.46
45-49
130845.00
Female 60-64
128441.42
Male 35-39
126280.00
Asian (United States of America) Female 30-34
125000.00
White (United States of America) Female 50-54
125000.00
Male 30-34
121280.00
Female 35-39
105000.00
390110 - Multiplatform Editor White (United States of America) Female 50-54 102234.81
280212 - Staff Writer  White (United States of America) Female 30-34
100780.00
25-29
91030.00
390110 - Multiplatform Editor White (United States of America) Male 45-49
90090.00
120202 - Assistant Editor White (United States of America) Female 25-29
84280.00
280212 - Staff Writer White (United States of America) Male 25-29
78208.20
390110 - Multiplatform Editor White (United States of America) Male 30-34
76055.50
71500.00
Female 25-29
68421.60

[199]: current_news_median_job_race_gender_age5_hourly = news_hourly.
   → groupby(['job_profile_current', 'race_ethnicity', 'gender', 'age_group_5']).
   → agg({'current_base_pay': [np.count_nonzero, np.median]})
   suppress_median(current_news_median_job_race_gender_age5_hourly)

[199]: Empty DataFrame
   Columns: [count_nonzero, median]
   Index: []

[200]: current_news_median_job_race_group_gender_age5_salaried = news_salaried.
   → groupby(['job_profile_current', 'race_grouping', 'gender', 'age_group_5']).
   → agg({'current_base_pay': [np.count_nonzero, np.median]})
   suppress_median(current_news_median_job_race_group_gender_age5_salaried)

[200]:

<table>
<thead>
<tr>
<th>job_profile_current</th>
<th>race_grouping</th>
<th>gender</th>
<th>age_group_5</th>
<th>current_base_pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>280212 - Staff Writer</td>
<td>white</td>
<td>Male</td>
<td>65+</td>
<td>5.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>55-59</td>
<td>17.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Female</td>
<td>55-59</td>
<td>7.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>45-49</td>
<td>10.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>40-44</td>
<td>9.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>60-64</td>
<td>11.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>40-44</td>
<td>20.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>50-54</td>
<td>14.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>45-49</td>
<td>9.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Female</td>
<td>60-64</td>
<td>6.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>35-39</td>
<td>25.00</td>
</tr>
<tr>
<td></td>
<td>person of color</td>
<td>Male</td>
<td>35-39</td>
<td>7.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>50-54</td>
<td>11.00</td>
</tr>
<tr>
<td></td>
<td>person of color</td>
<td>Male</td>
<td>50-54</td>
<td>6.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Male</td>
<td>30-34</td>
<td>21.00</td>
</tr>
<tr>
<td></td>
<td>unknown</td>
<td>Male</td>
<td>30-34</td>
<td>6.00</td>
</tr>
<tr>
<td>Job Profile</td>
<td>Race Grouping</td>
<td>Gender</td>
<td>Age Group</td>
<td>Median Salary</td>
</tr>
<tr>
<td>-------------</td>
<td>---------------</td>
<td>--------</td>
<td>-----------</td>
<td>----------------------</td>
</tr>
<tr>
<td>390110 - Multiplatform Editor</td>
<td>white</td>
<td>Female 35-39</td>
<td>17.00</td>
<td></td>
</tr>
<tr>
<td>280212 - Staff Writer</td>
<td>white</td>
<td>Female 30-34</td>
<td>13.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>person of color</td>
<td>Female 35-39</td>
<td>7.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female 25-29</td>
<td>14.00</td>
<td></td>
</tr>
<tr>
<td>390110 - Multiplatform Editor</td>
<td>white</td>
<td>Male 45-49</td>
<td>5.00</td>
<td></td>
</tr>
<tr>
<td>280212 - Staff Writer</td>
<td>person of color</td>
<td>Female 30-34</td>
<td>10.00</td>
<td></td>
</tr>
<tr>
<td>120202 - Assistant Editor</td>
<td>white</td>
<td>Female 25-29</td>
<td>5.00</td>
<td></td>
</tr>
<tr>
<td>280212 - Staff Writer</td>
<td>white</td>
<td>Male 25-29</td>
<td>6.00</td>
<td></td>
</tr>
<tr>
<td>390110 - Multiplatform Editor</td>
<td>white</td>
<td>Male 30-34</td>
<td>5.00</td>
<td></td>
</tr>
<tr>
<td>390410 - Digital Video Editor</td>
<td>person of color</td>
<td>Female 25-29</td>
<td>6.00</td>
<td></td>
</tr>
<tr>
<td>390110 - Multiplatform Editor</td>
<td>white</td>
<td>Male 25-29</td>
<td>5.00</td>
<td></td>
</tr>
</tbody>
</table>

job_profile_current | race_grouping | gender | age_group_5 | median |
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>280212 - Staff Writer</td>
<td>white</td>
<td>Male 65+</td>
<td>159458.37</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>55-59</td>
<td>153922.58</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Female 55-59</td>
<td>153780.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>45-49</td>
<td>144559.75</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>40-44</td>
<td>140000.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Male 60-64</td>
<td>134957.37</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>40-44</td>
<td>132980.42</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>50-54</td>
<td>132273.46</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>45-49</td>
<td>130845.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Female 60-64</td>
<td>128441.42</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Male 35-39</td>
<td>126280.00</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>person of color</td>
<td>Male 35-39</td>
<td>125000.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female 50-54</td>
<td>125000.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>person of color</td>
<td>Male 50-54</td>
<td>124140.35</td>
<td></td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Male 30-34</td>
<td>121280.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>unknown</td>
<td>Male 30-34</td>
<td>120390.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female 35-39</td>
<td>105000.00</td>
<td></td>
</tr>
<tr>
<td>390110 - Multiplatform Editor</td>
<td>white</td>
<td>Female 50-54</td>
<td>102234.81</td>
<td></td>
</tr>
<tr>
<td>280212 - Staff Writer</td>
<td>white</td>
<td>Female 30-34</td>
<td>100780.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>person of color</td>
<td>Female 35-39</td>
<td>99238.50</td>
<td></td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female 25-29</td>
<td>91030.00</td>
<td></td>
</tr>
<tr>
<td>390110 - Multiplatform Editor</td>
<td>white</td>
<td>Male 45-49</td>
<td>90090.00</td>
<td></td>
</tr>
<tr>
<td>280212 - Staff Writer</td>
<td>person of color</td>
<td>Female 30-34</td>
<td>85372.54</td>
<td></td>
</tr>
<tr>
<td>120202 - Assistant Editor</td>
<td>white</td>
<td>Female 25-29</td>
<td>84280.00</td>
<td></td>
</tr>
<tr>
<td>280212 - Staff Writer</td>
<td>white</td>
<td>Male 25-29</td>
<td>78208.20</td>
<td></td>
</tr>
<tr>
<td>390110 - Multiplatform Editor</td>
<td>white</td>
<td>Male 30-34</td>
<td>76055.50</td>
<td></td>
</tr>
<tr>
<td>390410 - Digital Video Editor</td>
<td>person of color</td>
<td>Female 25-29</td>
<td>75390.00</td>
<td></td>
</tr>
<tr>
<td>390110 - Multiplatform Editor</td>
<td>white</td>
<td>Male 25-29</td>
<td>71500.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Female 25-29</td>
<td>68421.60</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
current_news_median_job_race_group_gender_age5_hourly = news_hourly.
  groupby(['job_profile_current', 'race_grouping', 'gender', 'age_group_5']).
  agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_news_median_job_race_group_gender_age5_hourly)

Empty DataFrame
Columns: [count_nonzero, median]
Index: []

1.5.8 Performance evaluations

news_ratings = ratings_combined[ratings_combined['dept'] == 'News']

news_ratings_gender = news_ratings.groupby(['gender']).
  agg({'performance_rating': [np.count_nonzero, np.median]})
suppress_median(news_ratings_gender)

gender
count_nonzero median
Female 1892.00  3.40
Male  1772.00  3.40

news_ratings_race = news_ratings.groupby(['race_ethnicity']).
  agg({'performance_rating': [np.count_nonzero, np.median]})
suppress_median(news_ratings_race)

race_ethnicity
count_nonzero median
American Indian or Alaska Native (United States)... 12.00  3.60
White (United States of America)           2516.00  3.50
Asian (United States of America)           324.00  3.40
Prefer Not to Disclose (United States of America) 56.00  3.40
Black or African American (United States of Ame... 416.00  3.30
Hispanic or Latino (United States of America) 164.00  3.30
Native Hawaiian or Other Pacific Islander (Unit... 8.00  3.30
Two or More Races (United States of America)   80.00  3.20

news_ratings_race_gender = news_ratings.groupby(['race_ethnicity', 'gender']).
  agg({'performance_rating': [np.count_nonzero, np.median]})
suppress(news_ratings_race_gender)

race_ethnicity    gender
count_nonzero \
American Indian or Alaska Native (United States)... Female  8.00
Asian (United States of America)                Female 232.00
                                         Male   92.00
Black or African American (United States of Ame... Female 224.00
                                         Male   192.00
Hispanic or Latino (United States of America)   Female  80.00
                                         Male   84.00
<table>
<thead>
<tr>
<th>Race/Ethnicity</th>
<th>Female</th>
<th>Male</th>
</tr>
</thead>
<tbody>
<tr>
<td>American Indian or Alaska Native</td>
<td>3.70</td>
<td>3.30</td>
</tr>
<tr>
<td>Asian</td>
<td>3.40</td>
<td>3.40</td>
</tr>
<tr>
<td>Black or African American</td>
<td>3.25</td>
<td>3.30</td>
</tr>
<tr>
<td>Hispanic or Latino</td>
<td>3.30</td>
<td>3.30</td>
</tr>
<tr>
<td>Native Hawaiian or Other Pacific Islander</td>
<td>3.30</td>
<td>3.30</td>
</tr>
<tr>
<td>Prefer Not to Disclose</td>
<td>3.50</td>
<td>3.30</td>
</tr>
<tr>
<td>Two or More Races</td>
<td>3.20</td>
<td>3.20</td>
</tr>
<tr>
<td>White</td>
<td>3.40</td>
<td>3.50</td>
</tr>
</tbody>
</table>

```python
suppress(news_ratings_race_gender_under3)
```

<table>
<thead>
<tr>
<th>Race/Ethnicity</th>
<th>Female</th>
<th>Male</th>
</tr>
</thead>
<tbody>
<tr>
<td>person of color</td>
<td>57.00</td>
<td>49.00</td>
</tr>
<tr>
<td>white</td>
<td>92.00</td>
<td>80.00</td>
</tr>
</tbody>
</table>

```python
[207]: news_ratings_race_gender_over4 = news_ratings[news_ratings['performance_rating'] > 3.9].groupby(['race_grouping', 'gender']).agg({'performance_rating': [np.count_nonzero, np.median]})
suppress(news_ratings_race_gender_over4)
```

<table>
<thead>
<tr>
<th>Race/Ethnicity</th>
<th>Female</th>
<th>Male</th>
</tr>
</thead>
<tbody>
<tr>
<td>person of color</td>
<td>13.00</td>
<td>5.00</td>
</tr>
<tr>
<td>unknown</td>
<td>5.00</td>
<td>5.00</td>
</tr>
</tbody>
</table>
1.5.9 Pay changes

```
[208]: news_change = reason_for_change_combined[reason_for_change_combined['dept'] == 'News']

[209]: news_change_gender = news_change.groupby(['business_process_reason', 'gender']).
agg({'business_process_reason': [np.count_nonzero]}))
suppress_count(news_change_gender)
```

```
<table>
<thead>
<tr>
<th>business_process_reason</th>
<th>gender</th>
<th>count_nonzero</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Cont...</td>
<td>Male</td>
<td>813</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>809</td>
</tr>
<tr>
<td>Merit &gt; Performance &gt; Annual Performance Appraisal</td>
<td>Male</td>
<td>623</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>583</td>
</tr>
<tr>
<td>Data Change &gt; Data Change &gt; Change Job Details</td>
<td>Female</td>
<td>282</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>245</td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Move to another Manager</td>
<td>Male</td>
<td>185</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>169</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Mark...</td>
<td>Female</td>
<td>131</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td></td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Move to another Manager</td>
<td>Female</td>
<td>111</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Chan...</td>
<td>Female</td>
<td>90</td>
</tr>
<tr>
<td>Promotion &gt; Promotion &gt; Promotion</td>
<td>Female</td>
<td>84</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>79</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire &gt; New Position</td>
<td>Female</td>
<td>78</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire &gt; Fill Vacancy</td>
<td>Female</td>
<td>70</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Chan...</td>
<td>Male</td>
<td>62</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire &gt; New Position</td>
<td>Male</td>
<td>58</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire &gt; Fill Vacancy</td>
<td>Male</td>
<td>55</td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Transfer between departments</td>
<td>Female</td>
<td>27</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Incr...</td>
<td>Male</td>
<td>26</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Job ...</td>
<td>Male</td>
<td>24</td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Transfer between departments</td>
<td>Female</td>
<td>24</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Job ...</td>
<td>Female</td>
<td>22</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Incr...</td>
<td>Female</td>
<td>20</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Perf...</td>
<td>Male</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>11</td>
</tr>
<tr>
<td>Hire Employee &gt; Rehire &gt; New Position</td>
<td>Female</td>
<td>6</td>
</tr>
</tbody>
</table>
```

```
[210]: news_change_race = news_change.
      ~groupby(['business_process_reason', 'race_ethnicity']).
      ~agg({'business_process_reason': [np.count_nonzero]})
suppress_count(news_change_race)
```
count_nonzero

business_process_reason  race_ethnicity
Request Compensation Change > Adjustment > Cont... White (United States of America) 1164
Merit > Performance > Annual Performance Appraisal White (United States of America) 889
Data Change > Data Change > Change Job Details White (United States of America) 345
Transfer > Transfer > Move to another Manager White (United States of America) 201
Request Compensation Change > Adjustment > Mark... White (United States of America) 198
Request Compensation Change > Adjustment > Cont... Black or African American (United States of America) 169
Asian (United States of America) 138
Merit > Performance > Annual Performance Appraisal Black or African American (United States of America) 108
Asian (United States of America) 106
Promotion > Promotion > Promotion White (United States of America) 104
Hire Employee > New Hire > New Position White (United States of America) 93
Request Compensation Change > Adjustment > Chan... White (United States of America) 87
Hire Employee > New Hire > Fill Vacancy White (United States of America) 77
Request Compensation Change > Adjustment > Cont... Hispanic or Latino (United States of America) 71
Data Change > Data Change > Change Job Details Black or African American (United States of America) 55
Asian (United States of America) 54
Transfer > Transfer > Move to another Manager Black or African American (United States of America) 52
Merit > Performance > Annual Performance Appraisal Hispanic or Latino (United States of America) 46
Transfer > Transfer > Transfer between departments White (United States of America) 40
Request Compensation Change > Adjustment > Incr... White (United States of America) 34
Request Compensation Change > Adjustment > Mark... Asian (United States of America) 31
Request Compensation Change > Adjustment > Job ... White (United States of America) 31
Request Compensation Change > Adjustment > Mark... Black or African American
1.5.10 Performance evaluations x merit raises

```python
[211]: reason_for_change_combined['merit_raises'] = (~reason_for_change_combined['business_process_reason'].str.contains('Merit', ignore_case=True))

[212]: twenty14 = np.datetime64('2016-04-01')
twenty15 = np.datetime64('2017-04-01')
twenty16 = np.datetime64('2018-04-01')
twenty17 = np.datetime64('2019-04-01')
twenty18 = np.datetime64('2020-04-01')

def raise_time(row):
    if row['effective_date'] < twenty14:
        return 'before 2015'
    if row['effective_date'] < twenty15:
        return '2015'
    if row['effective_date'] < twenty16:
        return '2016'
    if row['effective_date'] < twenty17:
        return '2017'
    if row['effective_date'] < twenty18:
        return '2018'
    return 'unknown'

reason_for_change_combined['raise_after'] = reason_for_change_combined.apply(lambda row: raise_time(row), axis=1)
```
```python
merit_raises_news_gender_salaried = ~reason_for_change_combined[(reason_for_change_combined['merit_raises'] == True) & (reason_for_change_combined['dept'] == 'News') & (reason_for_change_combined['pay_rate_type'] == 'Salaried')].groupby([gender]).agg({ 'base_pay_change': [np.count_nonzero, np.median] })
suppress(merit_raises_news_gender_salaried)
```

<table>
<thead>
<tr>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>431.00</td>
<td>3000.00</td>
</tr>
<tr>
<td>Male</td>
<td>494.00</td>
<td>3000.00</td>
</tr>
</tbody>
</table>

```python
merit_raises_news_gender_hourly = ~reason_for_change_combined[(reason_for_change_combined['merit_raises'] == True) & (reason_for_change_combined['dept'] == 'News') & (reason_for_change_combined['pay_rate_type'] == 'Hourly')].groupby([gender]).agg({ 'base_pay_change': [np.count_nonzero, np.median] })
suppress(merit_raises_news_gender_hourly)
```

<table>
<thead>
<tr>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>78.00</td>
<td>1.27</td>
</tr>
<tr>
<td>Male</td>
<td>51.00</td>
<td>1.03</td>
</tr>
</tbody>
</table>

```python
merit_raises_news_race_salaried = ~reason_for_change_combined[(reason_for_change_combined['merit_raises'] == True) & (reason_for_change_combined['dept'] == 'News') & (reason_for_change_combined['pay_rate_type'] == 'Salaried')].groupby([race_ethnicity]).agg({ 'base_pay_change': [np.count_nonzero, np.median] })
suppress_median(merit_raises_news_race_salaried)
```

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>American Indian or Alaska Native (United States)</td>
<td>5.00</td>
<td>3500.00</td>
</tr>
<tr>
<td>Two or More Races (United States of America)</td>
<td>7.00</td>
<td>3500.00</td>
</tr>
<tr>
<td>Asian (United States of America)</td>
<td>69.00</td>
<td>3000.00</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>82.00</td>
<td>3000.00</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>707.00</td>
<td>3000.00</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>36.00</td>
<td>2500.00</td>
</tr>
</tbody>
</table>

```python
merit_raises_news_race_hourly = ~reason_for_change_combined[(reason_for_change_combined['merit_raises'] == True) & (reason_for_change_combined['dept'] == 'News') & (reason_for_change_combined['pay_rate_type'] == 'Hourly')].groupby([race_ethnicity]).agg({ 'base_pay_change': [np.count_nonzero, np.median] })
suppress_median(merit_raises_news_race_hourly)
```

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>American Indian or Alaska Native (United States)</td>
<td>5.00</td>
<td>3500.00</td>
</tr>
<tr>
<td>Two or More Races (United States of America)</td>
<td>7.00</td>
<td>3500.00</td>
</tr>
<tr>
<td>Asian (United States of America)</td>
<td>69.00</td>
<td>3000.00</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>82.00</td>
<td>3000.00</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>707.00</td>
<td>3000.00</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>36.00</td>
<td>2500.00</td>
</tr>
</tbody>
</table>
```
count_nonzero median
race_ethnicity
White (United States of America) 91.00 1.28
Black or African American (United States of America) 16.00 1.25
Asian (United States of America) 18.00 1.03

merit_raises_news_race_group_salaried =
~reason_for_change_combined['merit_raises'] == True &
~(reason_for_change_combined['dept'] == 'News') &
~(reason_for_change_combined['pay_rate_type'] == 'Salaried')]
~groupby(['race_grouping']).agg({'base_pay_change': [np.count_nonzero, np.median]})
suppress_median(merit_raises_news_race_group_salaried)

count_nonzero median
race_grouping
person of color 200.00 3000.00
white 707.00 3000.00
unknown 18.00 2860.00

merit_raises_news_race_group_hourly =
~reason_for_change_combined['merit_raises'] == True &
~(reason_for_change_combined['dept'] == 'News') &
~(reason_for_change_combined['pay_rate_type'] == 'Hourly')]
~groupby(['race_grouping']).agg({'base_pay_change': [np.count_nonzero, np.median]})
suppress_median(merit_raises_news_race_group_hourly)

count_nonzero median
race_grouping
white 91.00 1.28
person of color 38.00 1.03

merit_raises_news_gender_race_group_salaried =
~reason_for_change_combined['merit_raises'] == True &
~(reason_for_change_combined['dept'] == 'News') &
~(reason_for_change_combined['pay_rate_type'] == 'Salaried')]
~groupby(['gender', 'race_grouping']).agg({'base_pay_change': [np.count_nonzero, np.median]})
suppress_median(merit_raises_news_gender_race_group_salaried)

count_nonzero median
gender race_grouping
Female unknown 10.00 3500.00
    person of color 112.00 3000.00
    white 309.00 3000.00
Male white 398.00 3000.00
    person of color 88.00 2900.00
    unknown 8.00 2457.50
merit_raises_news_gender_race_group_hourly =
(reason_for_change_combined['merit_raises'] == True) &
(reason_for_change_combined['dept'] == 'News') &
(reason_for_change_combined['pay_rate_type'] == 'Hourly').
groupby(['gender', 'race_grouping']).agg({'base_pay_change': [np.count_nonzero, np.median]})
suppress_median(merit_raises_news_gender_race_group_hourly)

gender race_grouping count_nonzero median
Female white 59.00 1.28
person of color 19.00 1.26
Male person of color 19.00 1.03
white 32.00 1.02

fifteen_raises_amount =
(reason_for_change_combined['merit_raises'] == True) &
(reason_for_change_combined['dept'] == 'News') &
(reason_for_change_combined['pay_rate_type'] == 'Salaried') &
(reason_for_change_combined['raise_after'] == '2015').
groupby(['gender', 'race_grouping']).agg({'base_pay_change': [np.count_nonzero, np.median], '2015_annual_performance_rating': [np.count_nonzero, np.median]})
suppress(fifteen_raises_amount)

gender race_grouping count_nonzero median
Female person of color 17.00 2888.00
white 44.00 2500.00
Male person of color 10.00 2162.50
white 64.00 3000.00

fifteen_raises_score =
(reason_for_change_combined['merit_raises'] == True) &
(reason_for_change_combined['dept'] == 'News') &
(reason_for_change_combined['pay_rate_type'] == 'Salaried') &
(reason_for_change_combined['raise_after'] == '2015').
[np.count_nonzero, np.median])
suppress(fifteen_raises_score)

gender race_grouping count_nonzero median
Female person of color 17.00 3.40
white 44.00 3.70
Male person of color 10.00 3.50
white 64.00 3.65
sixteen_raises_amount =
~reason_for_change_combined[(reason_for_change_combined['merit_raises'] == True) & (reason_for_change_combined['dept'] == 'News') & ~reason_for_change_combined['pay_rate_type'] == 'Salaried') & ~reason_for_change_combined['raise_after'] == '2016'].
~groupby(['gender', 'race_grouping']).agg({'base_pay_change': [np.count_nonzero, np.median], '2016_annual_performance_rating': [np.count_nonzero, np.median]})
suppress(sixteen_raises_amount)

[223]:

<table>
<thead>
<tr>
<th>gender race_grouping</th>
<th>count_zero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female person of color white</td>
<td>26.00</td>
<td>3000.00</td>
</tr>
<tr>
<td>Male person of color white</td>
<td>17.00</td>
<td>3000.00</td>
</tr>
</tbody>
</table>

[224]:

<table>
<thead>
<tr>
<th>gender race_grouping</th>
<th>count_zero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female person of color white</td>
<td>26.00</td>
<td>3.40</td>
</tr>
<tr>
<td>Male person of color white</td>
<td>17.00</td>
<td>3.40</td>
</tr>
</tbody>
</table>

[225]:

seveneteen_raises_amount =
~reason_for_change_combined[(reason_for_change_combined['merit_raises'] == True) & (reason_for_change_combined['dept'] == 'News') & ~reason_forchange_combined['pay_rate_type'] == 'Salaried') & ~reason_forchange_combined['raise_after'] == '2017'].
~groupby(['gender', 'race_grouping']).agg({'base_pay_change': [np.count_nonzero, np.median], '2017_annual_performance_rating': [np.count_nonzero, np.median]})
suppress(seventeen_raises_amount)

[225]:

<table>
<thead>
<tr>
<th>gender race_grouping</th>
<th>count_zero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female person of color white</td>
<td>25.00</td>
<td>3000.00</td>
</tr>
<tr>
<td>Male person of color white</td>
<td>25.00</td>
<td>3000.00</td>
</tr>
</tbody>
</table>
seventeen_raises_score =
(reason_for_change_combined['merit_raises'] == True) & (reason_for_change_combined['dept'] == 'News') &
(reason_for_change_combined['pay_rate_type'] == 'Salaried') &
(reason_for_change_combined['raise_after'] == '2017').
groupby(['gender', 'race_grouping']).agg({'2017_annual_performance_rating': [np.count_nonzero, np.median]})
suppress(seventeen_raises_score)

gender race_grouping
count_nonzero median
Female person of color 25.00 3.50
   white 59.00 3.40
Male   person of color 25.00 3.40
   white 89.00 3.60

eighteen_raises_amount =
(reason_for_change_combined['merit_raises'] == True) & (reason_for_change_combined['dept'] == 'News') &
(reason_for_change_combined['pay_rate_type'] == 'Salaried') &
(reason_for_change_combined['raise_after'] == '2018').
groupby(['gender', 'race_grouping']).agg({'base_pay_change': [np.count_nonzero, np.median], '2018_annual_performance_rating': [np.count_nonzero, np.median]})
suppress(eighteen_raises_amount)

gender race_grouping
count_nonzero median
Female person of color 28.00 3000.00
   white 104.00 3000.00
Male   person of color 26.00 2500.00
   white 120.00 3000.00

eighteen_raises_score =
(reason_for_change_combined['merit_raises'] == True) & (reason_for_change_combined['dept'] == 'News') &
(reason_for_change_combined['pay_rate_type'] == 'Salaried') &
(reason_for_change_combined['raise_after'] == '2018').
groupby(['gender', 'race_grouping']).agg({'2018_annual_performance_rating': [np.count_nonzero, np.median]})
suppress(eighteen_raises_score)

gender race_grouping
count_nonzero median
Female person of color 28.00 3.50
   white 104.00 3.50
Male   person of color 26.00 3.40
   white 120.00 3.60
merit_raises_15 =
  reason_for_change_combined[(reason_for_change_combined['raise_after'] == '2015') & (reason_for_change_combined['merit_raises'] == True)]
merit_raises_16 =
  reason_for_change_combined[(reason_for_change_combined['raise_after'] == '2016') & (reason_for_change_combined['merit_raises'] == True)]
merit_raises_17 =
  reason_for_change_combined[(reason_for_change_combined['raise_after'] == '2017') & (reason_for_change_combined['merit_raises'] == True)]
merit_raises_18 =
  reason_for_change_combined[(reason_for_change_combined['raise_after'] == '2018') & (reason_for_change_combined['merit_raises'] == True)]

merit_raises_15 =
  merit_raises_15[['base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping']
  rename(columns={'2015_annual_performance_rating': 'performance_rating'})
merit_raises_16 =
  merit_raises_16[['base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping']
  rename(columns={'2016_annual_performance_rating': 'performance_rating'})
merit_raises_17 =
  merit_raises_17[['base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping']
  rename(columns={'2017_annual_performance_rating': 'performance_rating'})
merit_raises_18 =
  merit_raises_18[['base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping']
  rename(columns={'2018_annual_performance_rating': 'performance_rating'})

merit_raises_15 = pd.DataFrame(merit_raises_15)
merit_raises_16 = pd.DataFrame(merit_raises_16)
merit_raises_17 = pd.DataFrame(merit_raises_17)
merit_raises_18 = pd.DataFrame(merit_raises_18)

merit_raises_combined = pd.concat([merit_raises_15, merit_raises_16, merit_raises_17, merit_raises_18])

news_salaried_raises =
  merit_raises_combined[(merit_raises_combined['pay_rate_type'] == 'Salaried')
  & (merit_raises_combined['dept'] == 'News')].groupby(['gender', 'race_grouping']).agg({'base_pay_change': [np.
  count_nonzero, np.median]})
suppress(news_salaried_raises)

gender  race_grouping
Female  person of color  96.00  3000.00
unknown  9.00  3000.00
white    267.00  3000.00
Male     person of color  78.00  2658.52
### news_salaried_raises_scores

```
merit Raises combined [(merit raises combined [pay rate type] == 'Salaried') &
(dept] == 'News')].
```

```
groupby(['gender', 'race grouping']).agg({'performance rating': [np.
count nonzero, np.median]})
suppress(news_salaried_raises_scores)
```

<table>
<thead>
<tr>
<th>Gender</th>
<th>Race Grouping</th>
<th>Count Nonzero</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>Person of color</td>
<td>96.00</td>
<td>3.40</td>
</tr>
<tr>
<td>Unknown</td>
<td></td>
<td>9.00</td>
<td>3.90</td>
</tr>
<tr>
<td>Male</td>
<td>Person of color</td>
<td>78.00</td>
<td>3.40</td>
</tr>
<tr>
<td>Unknown</td>
<td></td>
<td>7.00</td>
<td>3.70</td>
</tr>
<tr>
<td>White</td>
<td></td>
<td>354.00</td>
<td>3.60</td>
</tr>
</tbody>
</table>

### news_hourly_raises

```
merit Raises combined [(merit raises combined [pay rate type] == 'Hourly') &
(dept] == 'News')].
```

```
groupby(['gender', 'race grouping']).agg({'base pay change': [np.
count nonzero, np.median]})
suppress(news_hourly_raises)
```

<table>
<thead>
<tr>
<th>Gender</th>
<th>Race Grouping</th>
<th>Count Nonzero</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>Person of color</td>
<td>18.00</td>
<td>1.27</td>
</tr>
<tr>
<td>White</td>
<td></td>
<td>54.00</td>
<td>1.46</td>
</tr>
<tr>
<td>Male</td>
<td>Person of color</td>
<td>19.00</td>
<td>1.03</td>
</tr>
<tr>
<td>White</td>
<td></td>
<td>28.00</td>
<td>1.16</td>
</tr>
</tbody>
</table>

### news_hourly_raises_scores

```
merit Raises combined [(merit raises combined [pay rate type] == 'Hourly') &
(dept] == 'News')].
```

```
groupby(['gender', 'race grouping']).agg({'performance rating': [np.
count nonzero, np.median]})
suppress(news_hourly_raises_scores)
```

<table>
<thead>
<tr>
<th>Gender</th>
<th>Race Grouping</th>
<th>Count Nonzero</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>Person of color</td>
<td>18.00</td>
<td>3.40</td>
</tr>
<tr>
<td>White</td>
<td></td>
<td>54.00</td>
<td>3.50</td>
</tr>
<tr>
<td>Male</td>
<td>Person of color</td>
<td>19.00</td>
<td>3.40</td>
</tr>
<tr>
<td>White</td>
<td></td>
<td>28.00</td>
<td>3.60</td>
</tr>
</tbody>
</table>
1.5.11 Era

```
[234]: bezos = df[(df['hire_date'] > '2013-10-04') & (df['dept'] == 'News') &
          (df['pay_rate_type'] == 'Salaried')]
graham = df[(df['hire_date'] < '2013-10-05') & (df['dept'] == 'News') &
            (df['pay_rate_type'] == 'Salaried')]

[235]: bezos_gender = bezos.groupby(['gender']).agg({'current_base_pay': [np.
          count_nonzero, np.median]})
suppress_median(bezos_gender)

[235]:

    count_nonzero  median
  gender
     Male        157.00 100780.00
     Female      180.00  87160.00

[236]: graham_gender = graham.groupby(['gender']).agg({'current_base_pay': [np.
          count_nonzero, np.median]})
suppress_median(graham_gender)

[236]:

    count_nonzero  median
  gender
     Male        133.00 127059.40
     Female      104.00 112136.48

[237]: bezos_race = bezos.groupby(['race_ethnicity']).agg({'current_base_pay': [np.
          count_nonzero, np.median]})
suppress_median(bezos_race)

[237]:

    count_nonzero  median
  race_ethnicity
            Black or African American (United States of America)  26.00 94963.74
            White (United States of America) 224.00 94519.11
            Asian (United States of America)  31.00 87000.00
      Prefer Not to Disclose (United States of America)  8.00 82140.00
            Hispanic or Latino (United States of America)  22.00 81249.94
      Two or More Races (United States of America)  14.00 79860.00

[238]: graham_race = graham.groupby(['race_ethnicity']).agg({'current_base_pay': [np.
          count_nonzero, np.median]})
suppress_median(graham_race)

[238]:

    count_nonzero  median
  race_ethnicity
            Hispanic or Latino (United States of America)  6.00 135272.46
            White (United States of America) 182.00 124500.00
            Asian (United States of America)  15.00 111761.01
            Black or African American (United States of America)  22.00 104397.79

[239]: bezos_race_group = bezos.groupby(['race_grouping']).agg({'current_base_pay': [np.
          count_nonzero, np.median]})
```
suppress_median(bezos_race_group)

[239]:
    race_grouping     count_nonzero      median
    unknown           20.00      113890.00
    white             224.00       94519.11
    person of color   93.00       86000.00

[240]:
    graham_race_group = graham.groupby(['race_grouping']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(graham_race_group)

[240]:
    race_grouping     count_nonzero      median
    unknown           9.00       151170.88
    white             182.00      124500.00
    person of color   46.00       110844.65

[241]:
    bezos_gender_race_group = bezos.groupby(['race_grouping', 'gender']).
    agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(bezos_gender_race_group)

[241]:
    race_grouping     gender  count_nonzero      median
    unknown     Male     10.00      121390.00
    white       Male     115.00      102780.00
    person of color Male   32.00      94026.24
    white       Female   109.00       88780.00
    person of color Female 61.00       82000.00

[242]:
    graham_gender_race_group = graham.groupby(['race_grouping', 'gender']).
    agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(graham_gender_race_group)

[242]:
    race_grouping     gender  count_nonzero      median
    unknown     Male     6.00       150975.44
    white       Male     103.00      128629.42
    person of color Male   24.00      117567.07
    white       Female   79.00       112511.94
    person of color Female 22.00       108594.26

[243]:
    bezos_gender_race_group_age5 = bezos.
    groupby(['race_grouping', 'gender', 'age_group_5']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(bezos_gender_race_group_age5)

[243]:
    race_grouping     gender  age_group_5  count_nonzero      median
    white       Female   45-49           7.00       160780.00
<table>
<thead>
<tr>
<th>Gender</th>
<th>Race</th>
<th>Age Group</th>
<th>Current Base Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>Male</td>
<td>55-59</td>
<td>8.00</td>
<td>156806.68</td>
</tr>
<tr>
<td>Female</td>
<td>40-44</td>
<td>6.00</td>
<td>143750.00</td>
</tr>
<tr>
<td>Male</td>
<td>40-44</td>
<td>15.00</td>
<td>136467.50</td>
</tr>
<tr>
<td>Person of Color Male</td>
<td>35-39</td>
<td>8.00</td>
<td>115530.00</td>
</tr>
<tr>
<td>White</td>
<td>Female 50-54</td>
<td>8.00</td>
<td>114975.40</td>
</tr>
<tr>
<td>Male</td>
<td>35-39</td>
<td>24.00</td>
<td>107880.00</td>
</tr>
<tr>
<td>Female</td>
<td>35-39</td>
<td>15.00</td>
<td>105000.00</td>
</tr>
<tr>
<td>Male</td>
<td>45-49</td>
<td>9.00</td>
<td>102795.60</td>
</tr>
<tr>
<td>Person of Color Female 35-39</td>
<td>8.00</td>
<td>99619.25</td>
<td></td>
</tr>
<tr>
<td>White</td>
<td>Male 30-34</td>
<td>29.00</td>
<td>94780.00</td>
</tr>
<tr>
<td>Person of Color Male 25-29</td>
<td>8.00</td>
<td>88540.00</td>
<td></td>
</tr>
<tr>
<td>White</td>
<td>Female 30-34</td>
<td>24.00</td>
<td>87050.00</td>
</tr>
<tr>
<td>Person of Color Male 30-34</td>
<td>5.00</td>
<td>87000.00</td>
<td></td>
</tr>
<tr>
<td>White</td>
<td>Female 30-34</td>
<td>19.00</td>
<td>87000.00</td>
</tr>
<tr>
<td>Person of Color Female 30-34</td>
<td>15.00</td>
<td>77000.00</td>
<td></td>
</tr>
<tr>
<td>White</td>
<td>Male 25-29</td>
<td>21.00</td>
<td>76780.00</td>
</tr>
<tr>
<td>Person of Color Female &lt;25</td>
<td>10.00</td>
<td>64390.00</td>
<td></td>
</tr>
<tr>
<td>White</td>
<td>Female &lt;25</td>
<td>9.00</td>
<td>64280.00</td>
</tr>
</tbody>
</table>

```python
[244]:
graham_gender_race_group_age5 = graham.
    => groupby(['race_grouping', 'gender', 'age_group_5']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(graham_gender_race_group_age5)

[244]:
count_nonzero  median
race_grouping  gender  age_group_5
white          Male   65+          8.00  153937.49
               35-39          11.00  147300.00
               55-59          19.00  146541.57
               Female 55-59  16.00  138564.42
               Male  50-54    21.00  134546.92
               60-64          14.00  123514.68
               Female 40-44  5.00   120780.00
person of color Female 40-44 | 5.00 | 118512.33 |
               Male  50-54    11.00  116349.15
white          Male  40-44    17.00  115236.94
               Female 50-54  15.00  114803.00
               60-64          7.00   112511.94
               Male  45-49    8.00   111473.26
               Female 45-49  12.00  100909.67
               30-34          8.00   100801.50
person of color Female 50-54 | 5.00 | 96994.47 |
white          Female 35-39 | 11.00 | 88000.00 |
               Male  30-34     5.00   83649.71
```
```python
bezos_gender_race_group_age5_tier = bezos.
    groupby(['race_grouping', 'gender', 'age_group_5', 'tier']).
    agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(bezos_gender_race_group_age5_tier)

[245]:

<table>
<thead>
<tr>
<th>race_grouping</th>
<th>gender</th>
<th>age_group_5</th>
<th>tier</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>white</td>
<td>Male</td>
<td>40-44</td>
<td>Tier 1</td>
<td>7.00</td>
<td>193280.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>35-39</td>
<td>Tier 1</td>
<td>10.00</td>
<td>130017.50</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>35-39</td>
<td>Tier 1</td>
<td>8.00</td>
<td>128330.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>30-34</td>
<td>Tier 1</td>
<td>12.00</td>
<td>125233.27</td>
</tr>
<tr>
<td></td>
<td></td>
<td>45-49</td>
<td>Tier 2</td>
<td>5.00</td>
<td>120780.00</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>25-29</td>
<td>Tier 1</td>
<td>5.00</td>
<td>100000.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>30-34</td>
<td>Tier 2</td>
<td>5.00</td>
<td>100000.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>35-39</td>
<td>Tier 2</td>
<td>8.00</td>
<td>98890.00</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>30-34</td>
<td>Tier 2</td>
<td>6.00</td>
<td>93780.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>25-29</td>
<td>Tier 2</td>
<td>6.00</td>
<td>91282.50</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>25-29</td>
<td>Tier 2</td>
<td>9.00</td>
<td>91000.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>35-39</td>
<td>Tier 2</td>
<td>5.00</td>
<td>90280.00</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>30-34</td>
<td>Tier 2</td>
<td>7.00</td>
<td>88132.61</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>30-34</td>
<td>Tier 3</td>
<td>9.00</td>
<td>86000.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>30-34</td>
<td>Tier 3</td>
<td>7.00</td>
<td>86000.00</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>30-34</td>
<td>Tier 3</td>
<td>6.00</td>
<td>83889.94</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>25-29</td>
<td>Tier 3</td>
<td>6.00</td>
<td>80250.00</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>25-29</td>
<td>Tier 3</td>
<td>11.00</td>
<td>77000.00</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>25-29</td>
<td>Tier 3</td>
<td>11.00</td>
<td>74780.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Tier 4</td>
<td>8.00</td>
<td>69890.00</td>
</tr>
</tbody>
</table>

[246]:

```

```python
graham_gender_race_group_age5_tier = graham.
    groupby(['race_grouping', 'gender', 'age_group_5', 'tier']).
    agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(graham_gender_race_group_age5_tier)

[246]:

<table>
<thead>
<tr>
<th>race_grouping</th>
<th>gender</th>
<th>age_group_5</th>
<th>tier</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>white</td>
<td>Male</td>
<td>55-59</td>
<td>Tier 1</td>
<td>5.00</td>
<td>175780.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>35-39</td>
<td>Tier 1</td>
<td>5.00</td>
<td>173280.00</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>50-54</td>
<td>Tier 1</td>
<td>5.00</td>
<td>167780.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>55-59</td>
<td>Tier 1</td>
<td>6.00</td>
<td>162854.23</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>40-44</td>
<td>Tier 1</td>
<td>6.00</td>
<td>151590.08</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>55-59</td>
<td>Tier 2</td>
<td>5.00</td>
<td>149029.98</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>65+</td>
<td>Tier 2</td>
<td>6.00</td>
<td>147473.21</td>
</tr>
<tr>
<td></td>
<td></td>
<td>35-39</td>
<td>Tier 2</td>
<td>5.00</td>
<td>147300.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>55-59</td>
<td>Tier 2</td>
<td>12.00</td>
<td>143129.04</td>
</tr>
<tr>
<td></td>
<td></td>
<td>50-54</td>
<td>Tier 2</td>
<td>13.00</td>
<td>128052.85</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>50-54</td>
<td>Tier 2</td>
<td>7.00</td>
<td>121515.90</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>60-64</td>
<td>Tier 2</td>
<td>6.00</td>
<td>115891.66</td>
</tr>
<tr>
<td></td>
<td>Female</td>
<td>50-54</td>
<td>Tier 2</td>
<td>7.00</td>
<td>108375.68</td>
</tr>
</tbody>
</table>
```

135
| Tier 2 | 5.00 107040.00 |
| Tier 2 | 6.00 98982.30 |
| Tier 2 | 8.00 97119.98 |
| Tier 2 | 6.00 87540.00 |
| Tier 2 | 5.00 87277.77 |
| Tier 3 | 5.00 81108.52 |

1.5.12 Overall disparity calculations

```python
news_groups = news_salaried.groupby(['age_group_5', 'tier']).agg({'current_base_pay': [np.count_nonzero, np.median]})
expected_medians = pd.merge(news_salaried, news_groups, on=['age_group_5', 'tier'])
```

!/Library/Frameworks/Python.framework/Versions/3.6/lib/python3.6/site-packages/pandas/core/reshape/merge.py:522: UserWarning: merging between different levels can give an unintended result (1 levels on the left, 2 on the right)
warnings.warn(msg, UserWarning)

```python
below_expected_medians = expected_medians[expected_medians['current_base_pay'] < expected_medians[('current_base_pay', 'median')]].groupby(['race_grouping', 'gender']).agg({'current_base_pay': [np.count_nonzero]})
suppress(below_expected_medians)
```

```python
count_nonzero
race_grouping gender count_nonzero
person of color Female 48.00
      Male  27.00
unknown   Female  8.00
          Male  8.00
white     Female  93.00
         Male  89.00
```

```python
above_expected_medians = expected_medians[expected_medians['current_base_pay'] > expected_medians[('current_base_pay', 'median')]].groupby(['race_grouping', 'gender']).agg({'current_base_pay': [np.count_nonzero]})
suppress(above_expected_medians)
```

```python
count_nonzero
race_grouping gender count_nonzero
person of color Female 30.00
      Male  21.00
unknown   Female  8.00
white     Female  90.00
         Male 121.00
```
expected_medians['disparity'] = expected_medians['current_base_pay'] -
expected_medians['median']
expected_medians['disparity_pct'] = (expected_medians['current_base_pay'] -
expected_medians['median'])/
expected_medians['median']

disparity = expected_medians.groupby(['race_grouping', 'gender']).
aggs({'disparity': [np.count_nonzero, np.median]})
suppress(disparity)

count_nonzero  median
race_grouping  gender
person of color Female  78.00  -1500.00
          Male       48.00       0.00
unknown   Female  11.00  -3500.00
          Male       16.00  2177.25
white      Female  183.00       0.00
          Male     210.00  2457.75

disparity_pct_above = expected_medians[expected_medians['disparity_pct'] > .05].
groupby(['race_grouping', 'gender']).aggs({'disparity': [np.count_nonzero, np.
median]})
suppress(disparity_pct_above)

count_nonzero  median
race_grouping  gender
person of color Female  21.00  9610.00
          Male       16.00  25880.00
unknown   Female        7.00  30000.00
          Male      61.00  21485.87
white      Female        7.00  28677.74
          Male      100.00  28677.74

disparity_pct_below = expected_medians[expected_medians['disparity_pct'] < -.05].
groupby(['race_grouping', 'gender']).aggs({'disparity': [np.count_nonzero, np.
median]})
suppress(disparity_pct_below)

count_nonzero  median
race_grouping  gender
person of color Female  36.00  -10195.04
          Male        19.00  -15435.00
unknown   Female       5.00  -14220.00
          Male       5.00  -15000.00
white      Female      72.00  -14000.00
          Male       70.00  -18765.53
### Disparity Analysis by Gender

<table>
<thead>
<tr>
<th>Race Grouping</th>
<th>Gender</th>
<th>Count</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>Person of Color</td>
<td>Female</td>
<td>78.00</td>
<td>-0.01</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>48.00</td>
<td>0.03</td>
</tr>
<tr>
<td>Unknown</td>
<td>Female</td>
<td>11.00</td>
<td>-0.05</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>16.00</td>
<td>0.04</td>
</tr>
<tr>
<td>White</td>
<td>Female</td>
<td>183.00</td>
<td>0.05</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>210.00</td>
<td>0.10</td>
</tr>
</tbody>
</table>

### Code Snippet
```python
bezos_news_groups = bezos.groupby(['age_group_5', 'tier']).agg({'current_base_pay': [np.count_nonzero, np.median]})
bezos_expected_medians = pd.merge(bezos, bezos_news_groups, on=['age_group_5', 'tier'])
bezos_disparity_gender = bezos_expected_medians.groupby(['gender']).agg({'disparity_pct': [np.count_nonzero, np.average]})
```

### Disparity Analysis by Race Grouping

<table>
<thead>
<tr>
<th>Race Grouping</th>
<th>Count</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>White</td>
<td>183.00</td>
<td>0.05</td>
</tr>
<tr>
<td></td>
<td>210.00</td>
<td>0.10</td>
</tr>
</tbody>
</table>

### Code Snippet
```python
bezos_disparity_race_group = bezos_expected_medians.groupby(['race_grouping']).agg({'disparity_pct': [np.count_nonzero, np.average]})
```
person of color  83.00  0.02
unknown        19.00  -0.01
white          209.00  0.07

[259]:
bezos_disparity_gender_race_group = bezos_expected_medians.
groupby(['race_grouping', 'gender']).agg({'disparity_pct': [np.count_nonzero,
np.average]})
suppress(bezos_disparity_gender_race_group)

[259]:
count_nonzero average
race_grouping  gender
person of color Female  56.00  0.01
                          Male  27.00  0.05
unknown          Female  9.00  -0.06
                          Male  10.00  0.04
white             Female 104.00  0.06
                          Male 105.00  0.08

[260]:
graham_disparity_gender = graham_expected_medians.groupby(['gender']).
agg({'disparity_pct': [np.count_nonzero, np.average]})
suppress(graham_disparity_gender)

[260]:
count_nonzero average
gender
Female  99.00  0.02
Male   125.00  0.07

[261]:
graham_disparity_race_group = graham_expected_medians.
groupby(['race_grouping']).agg({'disparity_pct': [np.count_nonzero, np.
average]})
suppress(graham_disparity_race_group)

[261]:
count_nonzero average
race_grouping
person of color  43.00  -0.05
unknown         8.00  -0.05
white           173.00  0.07

[262]:
graham_disparity_gender_race_group = graham_expected_medians.
groupby(['race_grouping', 'gender']).agg({'disparity_pct': [np.count_nonzero,
np.average]})
suppress(graham_disparity_gender_race_group)

[262]:
count_nonzero average
race_grouping  gender
person of color Female  21.00  -0.06
                          Male  22.00  -0.04
unknown         Female  5.00  -0.03
                          Male  75.00  0.04
white           Female  98.00  0.10
                          Male
1.5.13 Regression

```python
news_salaried_regression = pd.get_dummies(news_salaried_regression)
```

```python
news_salaried_regression = news_salaried_regression,
```

```python
```

```python
import statsmodels.formula.api as sm
model1 = sm.ols(data=news_salaried_regression, formula = 'current_base_pay ~ gender_Female + gender_Male')
result1 = model1.fit()
result1.summary()
```

```
class 'statsmodels.iolib.summary.Summary'

OLS Regression Results
==============================================================================

Dep. Variable: current_base_pay   R-squared:    0.040
Model:                  OLS            Adj. R-squared: 0.036
Method:        Least Squares       F-statistic:    11.76
Date:                   Wed, 06 Nov 2019 Prob (F-statistic):    9.87e-06
Time:                      10:27:46    Log-Likelihood:    -6931.6
No. Observations:        574        AIC:           1.387e+04
Df Residuals:            571        BIC:           1.388e+04
Df Model:                2
Covariance Type:        nonrobust
```

140
```
model2 = sm.ols(data=news_salaried_regression, formula = 'current_base_pay ~ race_grouping_white + race_grouping_person_of_color')
result2 = model2.fit()
result2.summary()
```

```
OLS Regression Results
==============================================================================
Dep. Variable: current_base_pay R-squared: 0.043
Model: OLS Adj. R-squared: 0.040
Method: Least Squares F-statistic: 12.81
Date: Wed, 06 Nov 2019 Prob (F-statistic): 3.60e-06
Time: 10:27:46 Log-Likelihood: -6930.6
No. Observations: 574 AIC: 1.387e+04
Df Residuals: 571 BIC: 1.388e+04
Df Model: 2
Covariance Type: nonrobust
==============================================================================
 coef    std err    t    P>|t|    [0.025  0.975]
Intercept  7.739e+04  1185.564  65.281  0.000    7.51e+04    7.97e+04
gender_Female  3.007e+04  1880.411  15.992  0.000    2.64e+04    3.38e+04
gender_Male  4.732e+04  1868.654  25.324  0.000    4.37e+04    5.1e+04
```

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 3.93e-28. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```
|                       | coef    | std err | t      | P>|t| |
|-----------------------|---------|---------|--------|-----|
| Intercept             | 8.419e+04 | 5184.782 | 16.238 | 0.000 |
| race_grouping_white   | -2.24e+04 | 9753.945 | -0.771 | 0.441 |
| race_grouping_person_of_color | -2.661e+04 | 8682.201 | -3.065 | 0.002 |
| race_grouping_person_of_color | -4.37e+04 | -9560.605 | 9.55e-55 | 0.000 |

```python
[266]: model3 = sm.ols(data=news_salaried_regression, formula = 'current_base_pay ~gender_Female + gender_Male + race_grouping_white +race_grouping_person_of_color')
result3 = model3.fit()
result3.summary()
```

Omnibus: 128.063
Prob(Omnibus): 0.000
Jarque-Bera (JB): 248.772
Skew: 1.253
Kurtosis: 5.030
Cond. No. 9.91

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
2.82e+04  4.06e+04
gender_Male       4.979e+04  3098.878   16.066   0.000
4.37e+04  5.59e+04
race_grouping_white -6074.5808  8048.101  -0.755   0.451
-2.19e+04  9732.973
race_grouping_person_of_color -2.432e+04  8563.749  -2.840  0.005
-4.11e+04  -7503.406

Omnibus:                      132.663 Durbin-Watson:                1.660
Prob(Omnibus):                  0.000  Jarque-Bera (JB):               270.377
Skew:                            1.269  Prob(JB):                    1.94e-59
Kurtosis:                       5.205  Cond. No.                    1.68e+15

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 4.23e-28. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```
[267]: new_news_salaried_regression = pd.DataFrame({'gender_Female': [1,0,1,0],
          'gender_Male': [0,1,0,1],
          'race_grouping_white': [1,1,0,0],
          'race_grouping_person_of_color': [0,0,1,1],
          'age': [40,40,40,40]})
new_news_salaried_regression['predicted'] = result3.predict(new_news_salaried_regression)
new_news_salaried_regression
```

```
   gender_Female  gender_Male  race_grouping_white    race_grouping_person_of_color  age  predicted
0             1             0                       1                             0       40  112522.15
1             0             1                       1                             0       40  127905.73
2             1             0                       0                             0       40   94272.97
3             0             1                       0                             0       40  109656.55
```

```
[268]: model4 = sm.ols(data=news_salaried_regression, formula = 'current_base_pay ~
          ~gender_Female + gender_Male + age_group_5_25_under + age_group_5_25to29 +
          ~age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 +
          ~age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 +
          ~age_group_5_60to64 + age_group_5_65_over')
result4 = model4.fit()
result4.summary()
```
OLS Regression Results

==============================================================================
Dep. Variable: current_base_pay  R-squared: 0.268
Model: OLS  Adj. R-squared: 0.255
Method: Least Squares  F-statistic: 20.63
Date: Wed, 06 Nov 2019  Prob (F-statistic): 9.77e-33
Time: 10:27:46  Log-Likelihood: -6853.6
No. Observations: 574  AIC: 1.373e+04
Df Residuals: 563  BIC: 1.378e+04
Df Model: 10
Covariance Type: nonrobust

==============================================================================
               coef    std err          t      P>|t|      [0.025      0.975]
-------------------------------------------------------------
Intercept      7.547e+04  1169.257       64.546      0.000       7.32e+04   7.78e+04
gender_Female  3.365e+04  1722.563       19.536      0.000       3.03e+04   3.7e+04
gender_Male    4.182e+04  1697.671       24.632      0.000       3.85e+04   4.52e+04
age_group_5_25_under -4.454e+04  7177.390      -6.205      0.000      -5.86e+04   -3.04e+04
age_group_5_25to29  -2.51e+04  3987.825      -6.294      0.000      -3.29e+04   -1.73e+04
age_group_5_30to34  -8982.7087  3766.135     -2.385      0.017     -1.64e+04   9473.725
age_group_5_35to39  1532.0128  4043.258       0.379      0.705     -6409.700   16409.700
age_group_5_40to44  1.998e+04  4621.927       4.322      0.000       1.09e+04   2.91e+04
age_group_5_45to49  1.214e+04  5439.050       2.231      0.026      1453.537   2.85e+04
age_group_5_50to54  1.483e+04  4405.774       3.367      0.001      6179.782   61879.782
age_group_5_55to59  3.081e+04  5045.129       6.108      0.000       2.09e+04   4.07e+04
age_group_5_60to64  2.446e+04  6619.091       3.695      0.000       1.15e+04   3.75e+04
age_group_5_65_over  5.034e+04  8904.898       5.653      0.000       3.28e+04   6.78e+04

==============================================================================
Omnibus: 164.069  Durbin-Watson: 1.859
Prob(Omnibus): 0.000    Jarque-Bera (JB): 434.791
Skew: 1.424    Prob(JB): 3.86e-95
Kurtosis: 6.173    Cond. No. 3.50e+15

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 7.64e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
model5 = sm.ols(data=news_salaried_regression, formula='current_base_pay ~ race_grouping_white + race_grouping_person_of_color + age_group_5_25_under +
+ age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 +
+ age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 +
+ age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over')
result5 = model5.fit()
result5.summary()
```

```bash
OLS Regression Results
=================================================================================================
Dep. Variable: current_base_pay  R-squared: 0.278
Model: OLS  Adj. R-squared: 0.264
Method: Least Squares  F-statistic: 19.71
Date: Wed, 06 Nov 2019  Prob (F-statistic): 1.04e-33
Time: 10:27:47  Log-Likelihood: -6849.6
No. Observations: 574  AIC: 1.372e+04
Df Residuals: 562  BIC: 1.378e+04
Df Model: 11
Covariance Type: nonrobust

=================================================================================================
                      coef    std err          t      P>|t|      [0.025    0.975]
-------------------------------------------------------------------------------
Intercept           1.214e+05    6405.473     18.954      0.000    1.09e+05    1.34e+05
race_grouping_white -1.047e+04    7206.280     -1.453      0.147    -2.46e+04    3.78e+04
race_grouping_person_of_color -2.275e+04    7648.553     -2.974      0.003    -3.78e+04    -7724.844
age_group_5_25_under  -3.946e+04    7161.284     -5.510      0.000    -5.35e+04    -2.54e+04
age_group_5_25to29    -2.206e+04    3963.986     -5.313      0.000    -3.11e+04    -1.30e+04
```
-2.88e+04  -1.33e+04
age_group_5_30to34  -4725.1241  3744.627  -1.262  0.208
-1.21e+04  2630.051
age_group_5_35to39  7317.7479  4085.001  1.791  0.074
-705.987  1.53e+04
age_group_5_40to44  2.557e+04  4583.847  5.579  0.000
1.66e+04  3.46e+04
age_group_5_45to49  1.616e+04  5474.110  2.953  0.003
5410.289  2.69e+04
age_group_5_50to54  2.101e+04  4434.740  4.738  0.000
1.23e+04  2.97e+04
age_group_5_55to59  3.468e+04  5070.678  6.839  0.000
2.47e+04  4.46e+04
age_group_5_60to64  2.782e+04  6646.739  4.185  0.000
1.48e+04  4.09e+04
age_group_5_65_over  5.409e+04  8887.598  6.086  0.000
3.66e+04  7.15e+04
==============================================================================
Omnibus:  164.311 Durbin-Watson:  1.827
Prob(Omnibus):  0.000 Jarque-Bera (JB):  428.700
Skew:  1.434 Prob(JB):  8.11e-94
Kurtosis:  6.114 Cond. No.  4.33e+15
==============================================================================
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The smallest eigenvalue is 5.26e-29. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.

[270]: model6 = sm.ols(data=news_salaried_regression, formula = 'current_base_pay ~
gender_Female + gender_Male + race_grouping_white +
race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 +
age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 +
age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 +
age_group_5_60to64 + age_group_5_65_over')
result6 = model6.fit()
result6.summary()

[270]: <class 'statsmodels.iolib.summary.Summary'>

OLS Regression Results
==============================================================================
Dep. Variable:  current_base_pay  R-squared:  0.285
Model:  OLS  Adj. R-squared:  0.269
Method:  Least Squares  F-statistic:  18.61
Date:  Wed, 06 Nov 2019  Prob (F-statistic):  4.46e-34

Time: 10:27:47  Log-Likelihood: -6847.0
No. Observations: 574  AIC: 1.372e+04
Df Residuals: 561  BIC: 1.378e+04
Df Model: 12
Covariance Type: nonrobust

==============================================================================
coef    std err     t    P>|t|      [0.025    0.975]
---------    ---------    -----    ------          --------    --------
Intercept     8.317e+04    4390.292    18.944     0.000    7.45e+04    9.18e+04
gender_Female  3.802e+04    2754.420    13.803     0.000    3.26e+04    4.34e+04
gender_Male    4.515e+04    2676.058    16.872     0.000    3.99e+04    5.04e+04
race_grouping_white -1.024e+04    7181.661    -1.426     0.154   -2.44e+04    3.86e+04
race_grouping_person_of_color -2.178e+04    7634.018    -2.853     0.004   -3.68e+04   -6784.597
age_group_5_25_under  -4.129e+04    7165.506    -5.762     0.000   -5.54e+04   -2.72e+04
age_group_5_25to29    -2.371e+04    3972.663    -5.968     0.000   -3.15e+04   -1.59e+04
age_group_5_30to34  -8102.4208    3737.928    -2.168     0.031   -1.54e+04   -760.377
age_group_5_35to39  3133.7668    4052.539     0.773     0.440   -4826.237    1.11e+04
age_group_5_40to44  2.076e+04    4584.120     4.529     0.000     1.29e+04    2.98e+04
age_group_5_45to49  1.224e+04    5430.218     2.254     0.025     1.573.060    2.29e+04
age_group_5_50to54  1.662e+04    4406.543     3.772     0.000     7.964.806    2.53e+04
age_group_5_55to59  3.053e+04    5035.484     6.064     0.000     2.06e+04    4.04e+04
age_group_5_60to64  2.369e+04    6599.643     3.589     0.000     1.07e+04    3.66e+04
age_group_5_65_over  4.929e+04    8839.927     5.576     0.000     3.19e+04    6.67e+04
==============================================================================
Omnibus: 164.304  Durbin-Watson: 1.830
Prob(Omnibus): 0.000  Jarque-Bera (JB): 437.349
Skew: 1.424  Prob(JB): 1.07e-95
Kurtosis: 6.190  Cond. No. 5.46e+15

147
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 4.26e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```
>>> model7 = sm.ols(data=news_salaried_regression, formula = 'current_base_pay ~ gender_Female + gender_Male + race_grouping_white +
race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 +
age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 +
age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 +
age_group_5_60to64 + age_group_5_65_over + tier_Tier_1 + tier_Tier_2 +
tier_Tier_3 + tier_Tier_4')
result7 = model7.fit()
result7.summary()
```

```
OLS Regression Results
==============================================================================
Dep. Variable: current_base_pay  R-squared:                         0.468
Model:                 OLS  Adj. R-squared:                        0.453
Method:                     Least Squares  F-statistic:                30.65
Date:             Wed, 06 Nov 2019  Prob (F-statistic):           5.96e-66
Time:                    10:27:47  Log-Likelihood:           -6762.0
No. Observations:        574  AIC:                            1.356e+04
Df Residuals:            557  BIC:                            1.363e+04
Df Model:                 16  Covariance Type:            nonrobust
==============================================================================

                      coef    std err          t      P>|t|    [0.025  0.975]
------------------------------------------------------------------------------
Intercept            6.71e+04   5610.010      11.961      0.000   5.61e+04  7.81e+04
gender_Female       3.114e+04   3178.867      9.796      0.000   2.49e+04  3.74e+04
gender_Male         3.596e+04   3080.637     11.672      0.000   2.99e+04  4.2e+04
race_grouping_white 1.021e+04   6456.704      1.581      0.114  -2474.772  1590.1942
-2474.772  1590.1942
race_grouping_person_of_color -1.19e+04   6868.290      0.232      0.817  -1.51e+04  1.19e+04
age_group_5_25_under  -3.328e+04   6252.504     -5.323      0.000  -6.51e+04  -1.14e+04
```

148
-4.56e+04  -2.1e+04
age_group_5_25to29  -1.518e+04  3560.433  -4.264  0.000
-2.22e+04  -8187.622
age_group_5_30to34  -7122.3046  3257.952  -2.186  0.029
-1.35e+04  -722.931
age_group_5_35to39  -2713.7685  3565.793  -0.761  0.447
-9717.813  4290.276
age_group_5_40to44  1.51e+04  4003.127  3.772  0.000
7234.726  2.3e+04
age_group_5_45to49  1.045e+04  4759.738  2.195  0.029
1099.480  1.98e+04
age_group_5_50to54  1.739e+04  3856.830  4.510  0.000
9818.978  2.5e+04
age_group_5_55to59  2.519e+04  4413.333  5.709  0.000
1.65e+04  3.39e+04
age_group_5_60to64  1.896e+04  5743.118  3.302  0.001
7683.999  3.02e+04
age_group_5_65_over  3.83e+04  7727.397  4.956  0.000
2.31e+04  5.35e+04
tier_Tier_1  3.272e+04  6633.224  4.933  0.000
1.97e+04  4.58e+04
tier_Tier_2  1744.2999  6479.903  0.269  0.788
-1.1e+04  1.45e+04
tier_Tier_3  -1.888e+04  6638.849  -2.844  0.005
-3.19e+04  -5839.310
tier_Tier_4  -2.075e+04  8097.266  -2.562  0.011
-3.67e+04  -4843.693

==============================================================================
Omnibus:        215.055  Durbin-Watson:        1.864
Prob(Omnibus):  0.000  Jarque-Bera (JB):    959.817
Skew:            1.648  Prob(JB):             3.79e-209
Kurtosis:        8.410  Cond. No.            5.90e+15
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The smallest eigenvalue is 4.11e-29. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.

[272]:
```python
model8 = sm.ols(data=news_salaried_regression, formula='current_base_pay ~ 
gender_Female + gender_Male + race_grouping_white + 
race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 + 
age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + 
age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + 
age_group_5_60to64 + age_group_5_65_over + tier_Tier_1 + tier_Tier_2 + 
tier_Tier_3 + tier_Tier_4 + years_of_service_grouped_0 + 
years_of_service_grouped_1to2 + years_of_service_grouped_3to5 + 
years_of_service_grouped_6to10 + years_of_service_grouped_11to15 + 
years_of_service_grouped_16to20 + years_of_service_grouped_21to25 + 
years_of_service_grouped_25_over')
result8 = model8.fit()
result8.summary()
```

```
[272]: <class 'statsmodels.iolib.summary.Summary'>

```
```
| age_group_5_30to34 | -8875.1051 | 3619.177 | -2.452 | 0.015 |
| age_group_5_35to39 | -4003.6497 | 3846.671 | -1.041 | 0.298 |
| age_group_5_40to44 | 1.462e+04 | 4113.878 | 3.554 | 0.000 |
| age_group_5_45to49 | 1.107e+04 | 4841.180 | 2.287 | 0.023 |
| age_group_5_50to54 | 1.852e+04 | 3991.699 | 4.639 | 0.000 |
| age_group_5_55to59 | 2.653e+04 | 4650.059 | 5.706 | 0.000 |
| age_group_5_60to64 | 1.895e+04 | 6575.681 | 2.882 | 0.004 |
| age_group_5_65_over | 4.064e+04 | 7966.529 | 5.101 | 0.000 |
| tier_Tier_1 | 3.309e+04 | 6652.958 | 4.974 | 0.000 |
| tier_Tier_2 | 2399.3566 | 6509.221 | 0.369 | 0.713 |
| tier_Tier_3 | -1.918e+04 | 6645.430 | -2.886 | 0.004 |
| tier_Tier_4 | -2.167e+04 | 8152.480 | -2.658 | 0.008 |
| years_of_service_grouped_0 | 1.332e+04 | 4281.224 | 3.112 | 0.002 |
| years_of_service_grouped_1to2 | 1.387e+04 | 3327.196 | 4.167 | 0.000 |
| years_of_service_grouped_3to5 | 8772.2226 | 3262.142 | 2.689 | 0.007 |
| years_of_service_grouped_6to10 | 5135.7202 | 4106.907 | 1.251 | 0.212 |
| years_of_service_grouped_11to15 | 5367.3146 | 4446.798 | 1.207 | 0.228 |
| years_of_service_grouped_16to20 | 5973.2655 | 4078.267 | 1.465 | 0.144 |
| years_of_service_grouped_21to25 | -1787.3518 | 6278.264 | -0.285 | 0.776 |
| years_of_service_grouped_25_over | 1.052e+04 | 6148.567 | 1.710 | 0.088 |

==============================================================================
Omnibus: 205.644 Durbin-Watson: 1.878
Prob(Omnibus): 0.000 Jarque-Bera (JB): 854.496
Skew: 1.594 Prob(JB): 2.81e-186
Kurtosis: 8.056 Cond. No. 1.12e+16
==============================================================================
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 1.21e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```
[273]: merit_raises_combined_salaried_regression =
    \arrowmerit_raises_combined[(merit_raises_combined['dept'] == 'News') &
    \arrow(merit_raises_combined['pay_rate_type'] == 'Salaried')]
merit_raises_combined_salaried_regression = pd.
    \arrowget_dummies(merit_raises_combined_salaried_regression,
    \arrowcolumns=['gender','race_grouping','age_group_5'])

[274]: merit_raises_combined_salaried_regression =
    \arrowmerit_raises_combined_salaried_regression
    \arrowrename(columns={'race_grouping_person of color':
        \arrow'race_grouping_person_of_color',
        \arrow'age_group_5_<25':
        \arrow'age_group_5_25to29','age_group_5_30-34':
        \arrow'age_group_5_30to34','age_group_5_35-39':
        \arrow'age_group_5_35to39','age_group_5_40-44':
        \arrow'age_group_5_40to44','age_group_5_45-49':
        \arrow'age_group_5_45to49','age_group_5_50-54':
        \arrow'age_group_5_50to54','age_group_5_55-59':
        \arrow'age_group_5_55to59','age_group_5_60-64':
        \arrow'age_group_5_60to64','age_group_5_65+':
        \arrow'age_group_5_65_over'})
model9 = sm.ols(data=merit_raises_combined_salaried_regression, formula =
    \arrow'base_pay_change ~ gender_Female + gender_Male')
result9 = model9.fit()
result9.summary()
```

```
<class 'statsmodels.iolib.summary.Summary'>
```

```
==============================================================================
Dep. Variable:    base_pay_change  R-squared:      0.004
Model:              OLS             Adj. R-squared:  0.003
Method:             Least Squares  F-statistic:    3.275
Date:               Wed, 06 Nov 2019 Prob (F-statistic): 0.0707
Time:               10:27:48      Log-Likelihood: -7121.9
No. Observations:   811            AIC:            1.425e+04
Df Residuals:       809            BIC:            1.426e+04
Df Model:           1
Covariance Type:    nonrobust
==============================================================================

 coef    std err          t      P>|t|      [0.025
0.025
```
```
0.975]
- Intercept 2116.8178 37.068 57.107 0.000 2044.057
  2189.578
gender_Female 957.7901 60.044 15.951 0.000 839.929
  1075.651
gender_Male 1159.0276 57.138 20.285 0.000 1046.871
  1271.185
Omnibus: 599.428 Durbin-Watson: 1.975
Prob(Omnibus): 0.000 Jarque-Bera (JB): 15042.743
Skew: 3.055 Prob(JB): 0.00
Kurtosis: 23.195 Cond. No. 5.43e+15
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
  specified.
[2] The smallest eigenvalue is 4.14e-29. This might indicate that there are
  strong multicollinearity problems or that the design matrix is singular.
```
2653.106 4200.394
race_grouping_white -179.187 399.177 -0.449 0.654
-962.735 604.359
race_grouping_person_of_color -494.071 411.855 -1.200 0.231
-1302.503 314.361

Omnibus: 595.371 Durbin-Watson: 1.967
Prob(Omnibus): 0.000 Jarque-Bera (JB): 14962.329
Skew: 3.023 Prob(JB): 0.00
Kurtosis: 23.155 Cond. No. 16.1

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

```
[276]: model11 = sm.ols(data=merit_raises_combined_salaried_regression, formula = u'
    base_pay_change ~ gender_Female + gender_Male + race_grouping_white +
    race_grouping_person_of_color')
result11 = model11.fit()
result11.summary()
```

```
OLS Regression Results
==============================================================================
Dep. Variable: base_pay_change R-squared: 0.010
Model: OLS Adj. R-squared: 0.007
Method: Least Squares F-statistic: 2.802
Date: Wed, 06 Nov 2019 Prob (F-statistic): 0.0390
Time: 10:27:48 Log-Likelihood: -7119.3
No. Observations: 811 AIC: 1.425e+04
Df Residuals: 807 BIC: 1.427e+04
Df Model: 3
Covariance Type: nonrobust
==============================================================================

coef std err t P>|t|
[0.025 0.975]
==============================================================================
Intercept 2291.9731 262.539 8.730 0.000
1776.633 2807.313
gender_Female 1056.3093 141.724 7.453 0.000
778.117 1334.501
gender_Male 1235.6638 143.543 8.608 0.000
953.901 1517.426
```
<table>
<thead>
<tr>
<th></th>
<th>race_grouping_white</th>
<th>race_grouping_person_of_color</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-202.9609</td>
<td>-496.0038</td>
</tr>
<tr>
<td></td>
<td>399.061</td>
<td>411.454</td>
</tr>
<tr>
<td></td>
<td>-0.509</td>
<td>-1.205</td>
</tr>
<tr>
<td></td>
<td>0.611</td>
<td>0.228</td>
</tr>
</tbody>
</table>

Omnibus: 595.574  Durbin-Watson: 1.970
Prob(Omnibus): 0.000  Jarque-Bera (JB): 14866.159
Skew: 3.027  Prob(JB): 0.00
Kurtosis: 23.082  Cond. No. 6.20e+15

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 4.58e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
new_reason_for_change_combined_regression = pd.DataFrame({'gender_Female': [1, 0, 1, 0], 'gender_Male': [0, 1, 0, 1], 'race_grouping_white': [1, 1, 0, 0], 'race_grouping_person_of_color': [0, 0, 1, 1]})
new_reason_for_change_combined_regression['predicted'] = result11.predict(new_reason_for_change_combined_regression)
```

```python
model12 = sm.ols(data=merit_raises_combined_salaried_regression, formula = 'base_pay_change ~ gender_Female + gender_Male + age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over')
result12 = model12.fit()
result12.summary()
```

```
<class 'statsmodels.iolib.summary.Summary'>
```

### OLS Regression Results
```
Dep. Variable: base_pay_change  R-squared: 0.047
Model: OLS  Adj. R-squared: 0.035
Method: Least Squares  F-statistic: 3.937
Date: Wed, 06 Nov 2019  Prob (F-statistic): 2.95e-05
Time: 10:27:48  Log-Likelihood: -7104.1
No. Observations: 811  AIC: 1.423e+04
Df Residuals: 800  BIC: 1.428e+04
Df Model: 10
Covariance Type: nonrobust

| coef  | std err | t    | P>|t| | [0.025 |
|-------|---------|------|-----|--------|
| Intercept | 1900.1395 | 51.767 | 36.706 | 0.000 | 1798.525 |
| gender_Female | 837.3584 | 61.567 | 13.601 | 0.000 | 716.506 |
| gender_Male | 1062.7812 | 60.759 | 17.492 | 0.000 | 943.516 |
| age_group_5_25_under | -625.0684 | 577.767 | -1.082 | 0.280 | -1759.186 |
| age_group_5_25to29 | 348.4845 | 185.964 | 1.874 | 0.061 | -16.551 |
| age_group_5_30to34 | 508.1254 | 142.282 | 3.571 | 0.000 | 228.834 |
| age_group_5_35to39 | 681.6571 | 149.030 | 4.574 | 0.000 | 389.122 |
| age_group_5_40to44 | 629.9350 | 163.125 | 3.862 | 0.000 | 309.732 |
| age_group_5_45to49 | 455.9623 | 179.299 | 2.543 | 0.011 | 104.010 |
| age_group_5_50to54 | -199.2284 | 148.416 | -1.342 | 0.180 | -490.559 |
| age_group_5_55to59 | 249.0856 | 165.464 | 1.505 | 0.133 | -75.710 |
| age_group_5_60to64 | 163.6817 | 207.378 | 0.789 | 0.430 | -243.387 |
| age_group_5_65_over | -312.4953 | 240.444 | -1.300 | 0.194 | -784.471 |

Omnibus: 607.312  Durbin-Watson: 1.979
Prob(Omnibus): 0.000  Jarque-Bera (JB): 16080.305
Skew: 3.095  Prob(JB): 0.00
Kurtosis: 23.918  Cond. No. 5.24e+15
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 4.81e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```
[279]: model13 = sm.ols(data=merit_raises_combined_salaried_regression, formula ='
    → 'base_pay_change ~ race_grouping_white + race_grouping_person_of_color +
    → age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 +
    → age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
    → age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
    → age_group_5_65_over')
result13 = model13.fit()
result13.summary()
```

```
<class 'statsmodels.iolib.summary.Summary'>
```

```
OLS Regression Results
==============================================================================
Dep. Variable:    base_pay_change   R-squared:         0.052
Model:              OLS                 Adj. R-squared:   0.039
Method:              Least Squares  F-statistic:      3.976
Date:            Wed, 06 Nov 2019  Prob (F-statistic): 1.17e-05
Time:              10:27:48       Log-Likelihood:   -7101.9
No. Observations:         811   AIC:                  1.423e+04
Df Residuals:           799    BIC:                  1.428e+04
Df Model:              11
Covariance Type:        nonrobust
==============================================================================
                  coef    std err          t      P>|t|    [0.025  0.975]
------------------------------------------------------------------------------
Intercept        2856.8954    360.070       7.934      0.000     2150.101  3563.689
race_grouping_white  -33.7963    395.935     -0.085      0.932     -810.992  743.399
race_grouping_person_of_color  -425.7390    407.658     -1.044      0.297    -1225.947  374.469
age_group_5_25_under   -1225.947    743.399     -1.644      0.100    -2699.570  247.687
age_group_5_25to29     -673.0990   579.089     -1.162      0.245     -1809.814  463.616
age_group_5_30to34        440.9979   187.438      2.353      0.019        73.070  808.926
age_group_5_30to34        628.8243  146.144      4.303      0.000        341.953  915.695
```

157
age_group_5_35to39  816.7998  153.462  5.323  0.000
515.564  1118.035
age_group_5_40to44  803.8584  163.611  4.913  0.000
482.700  1125.017
age_group_5_45to49  540.1748  182.411  2.961  0.003
182.113  898.237
age_group_5_50to54  -33.4153  153.074 -0.218  0.827
-333.891  267.060
age_group_5_55to59  333.7727  169.018  1.975  0.049
2.002  665.544
age_group_5_60to64  248.7739  208.615  1.193  0.233
-160.724  658.272
age_group_5_65_over -249.7919  244.401 -1.022  0.307
-729.535  229.951
==============================================================================
Omnibus:                       601.567  Durbin-Watson:                       1.971
Prob(Omnibus):                  0.000  Jarque-Bera (JB):              15921.391
Skew:                           3.050  Prob(JB):                           0.00
Kurtosis:                       23.832  Cond. No.                       3.09e+15
==============================================================================
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
    specified.
[2] The smallest eigenvalue is 1.53e-28. This might indicate that there are
    strong multicollinearity problems or that the design matrix is singular.

[280]: model14 = sm.ols(data=merit_raises_combined_salaried_regression, formula =
    'base_pay_change ~ gender_Female + gender_Male + race_grouping_white +
    race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 +
    age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 +
    age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 +
    age_group_5_60to64 + age_group_5_65_over')
result14 = model14.fit()
result14.summary()

[280]: <class 'statsmodels.iolib.summary.Summary'>

OLS Regression Results
==============================================================================
Dep. Variable:    base_pay_change   R-squared:                     0.056
Model:                    OLS        Adj. R-squared:               0.041
Method:              Least Squares      F-statistic:            3.916
Date:                        Wed, 06 Nov 2019    Prob (F-statistic):       7.08e-06
Time:                         10:27:48          Log-Likelihood:          -7100.3
No. Observations:          811              AIC:                 1.423e+04
Df Residuals:              798              BIC:                 1.429e+04
==============================================================================
Df Model: 12
Covariance Type: nonrobust

|              | coef    | std err | t      | P>|t| |
|--------------|---------|---------|--------|-----|
| [0.025 0.975]|         |         |        |     |

|              | coef    | std err | t      | P>|t| |
|--------------|---------|---------|--------|-----|
| Intercept    | 1978.5484 | 247.351 | 7.999  | 0.000 |
| 1493.012     | 2464.085 |         |        |     |
| gender_Female| 890.8640 | 133.899 | 6.653  | 0.000 |
| 628.028      | 1153.700 |         |        |     |
| gender_Male  | 1087.6844 | 137.238 | 7.926  | 0.000 |
| 818.295      | 1357.074 |         |        |     |
| race_grouping_white | -64.1327 | 395.777 | -0.162 | 0.871 |
| -841.019     | 712.754  |         |        |     |
| race_grouping_person_of_color | -431.5462 | 407.127 | -1.060 | 0.289 |
| -1230.713    | 367.621  |         |        |     |
| age_group_5_25_under | -688.0832 | 577.471 | -1.192 | 0.234 |
| -1821.625    | 445.459  |         |        |     |
| age_group_5_25to29 | 375.2333 | 186.600 | 2.011  | 0.045 |
| 8.948        | 741.519  |         |        |     |
| age_group_5_30to34 | 548.5259 | 144.590 | 3.794  | 0.000 |
| 264.704      | 832.348  |         |        |     |
| age_group_5_35to39 | 725.4046 | 151.661 | 4.783  | 0.000 |
| 427.703      | 1023.106 |         |        |     |
| age_group_5_40to44 | 687.3088 | 163.983 | 4.191  | 0.000 |
| 365.421      | 1009.197 |         |        |     |
| age_group_5_45to49 | 447.0509 | 180.736 | 2.473  | 0.014 |
| 92.276       | 801.826  |         |        |     |
| age_group_5_50to54 | -140.0335 | 151.593 | -0.924 | 0.356 |
| -437.602     | 157.535  |         |        |     |
| age_group_5_55to59 | 233.3376 | 167.382 | 1.394  | 0.164 |
| -95.223      | 561.898  |         |        |     |
| age_group_5_60to64 | 151.2637 | 207.454 | 0.729  | 0.466 |
| -255.957     | 558.484  |         |        |     |
| age_group_5_65_over | -361.4597 | 242.706 | -1.489 | 0.137 |
| -837.878     | 114.958  |         |        |     |

Omnibus: 602.033 Durbin-Watson: 1.973
Prob(Omnibus): 0.000 Jarque-Bera (JB): 15800.004
Skew: 3.057 Prob(JB): 0.00
Kurtosis: 23.741 Cond. No. 6.22e+15

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The smallest eigenvalue is 4.8e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
model15 = sm.ols(data=merit_raises_combined_salaried_regression, formula='performance_rating ~ gender_Female + gender_Male')
result15 = model15.fit()
result15.summary()
```

[281]: <class 'statsmodels.iolib.summary.Summary'>

```
# OLS Regression Results

==============================================================================
Dep. Variable:     performance_rating   R-squared:                       0.012
Model:               OLS               Adj. R-squared:                  0.011
Date:                Wed, 06 Nov 2019  Prob (F-statistic):               0.00246
Time:                10:27:49         Log-Likelihood:                 -231.28
No. Observations:    763               AIC:                           466.6
Df Residuals:        761               BIC:                           475.8
Df Model:            1
Covariance Type:     nonrobust

==============================================================================
              coef    std err          t      P>|t|      [0.025    0.975]
-----------------------------------------------------------------------------
Intercept      2.3801     0.008  299.623      0.000      2.364     2.396
gender_Female  1.1538     0.013   89.739      0.000      1.129     1.179
gender_Male    1.2262     0.012 100.061      0.000      1.202     1.250

==============================================================================
Omni: 26.124 Durbin-Watson: 1.853
Prob(Omni): 0.000 Jarque-Bera (J): 28.140
Skew: 0.470 Prob(JB): 7.75e-7
Kurtosis: 3.040 Cond. No. 5.04e+15
```

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 4.52e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```
model16 = sm.ols(data=merit_raises_combined_salaried_regression, formula = 'performance_rating ~ race_grouping_white + race_grouping_person_of_color')
result16 = model16.fit()
result16.summary()

model17 = sm.ols(data=merit_raises_combined_salaried_regression, formula = 'performance_rating ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color')
result17 = model17.fit()
result17.summary()
OLS Regression Results

Dep. Variable: performance_rating   R-squared: 0.043
Model: OLS   Adj. R-squared: 0.039
Method: Least Squares   F-statistic: 11.32
Date: Wed, 06 Nov 2019   Prob (F-statistic): 2.88e-07
No. Observations: 763   AIC: 446.4
Df Residuals: 759   BIC: 464.9
Df Model: 3
Covariance Type: nonrobust

==============================================================================
                           coef    std err          t      P>|t|      [0.025    0.975]
==============================================================================
Intercept                2.4859    0.054    46.122      0.000     2.380     2.592
gender_Female            1.2117    0.029    41.457      0.000     1.154     1.269
gender_Male              1.2742    0.030    43.015      0.000     1.216     1.332
race_grouping_white     -0.1331    0.082    -1.624      0.105    -0.294     0.028
race_grouping_person_of_color  -0.2629    0.085    -3.105      0.002    -0.429    -0.097
==============================================================================
Omni: 18.909   Durbin-Watson: 1.865
Prob(Omni): 0.000   Jarque-Bera (JB): 19.811
Skew: 0.394   Prob(JB): 4.99e-05
Kurtosis: 3.041   Cond. No. 5.84e+15

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 4.85e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.
```python
model18 = sm.ols(data=merit_raises_combined_salaried_regression, formula='performance_rating ~ gender_Female + gender_Male + age_group_5_25_under +
                   age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 +
                   age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 +
                   age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over')
result18 = model18.fit()
result18.summary()
```

```
[284]: <class 'statsmodels.iolib.summary.Summary'>

"""
OLS Regression Results
==============================================================================
Dep. Variable:  performance_rating   R-squared:                       0.046
Model:                OLS   Adj. R-squared:                  0.033
Method:              Least Squares  F-statistic:              3.588
Date:       Wed, 06 Nov 2019   Prob (F-statistic):         0.000114
Time:         10:27:49   Log-Likelihood:          -218.10
No. Observations:      763   AIC:                 458.2
Df Residuals:          752   BIC:                 509.2
Df Model:             10
Covariance Type:                  nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025
Intercept       2.2183     0.011    202.702      0.000     2.197
2.240
gender_Female   1.0816     0.013     81.967      0.000     1.056
1.107
gender_Male     1.1368     0.013     87.233      0.000     1.111
1.162
age_group_5_25_under -0.0591     0.121    -0.489      0.625    -0.296
0.178
age_group_5_25to29  0.1312     0.040      3.286      0.001     0.053
0.210
age_group_5_30to34  0.1968     0.030      6.461      0.000     0.137
0.257
age_group_5_35to39  0.2457     0.032      7.720      0.000     0.183
0.308
age_group_5_40to44  0.2914     0.035      8.387      0.000     0.223
0.360
age_group_5_45to49  0.2170     0.038      5.715      0.000     0.142
0.292
age_group_5_50to54  0.2553     0.032      7.916      0.000     0.192
0.319
```
```
age_group_5_55to59 0.3077 0.036 8.563 0.000 0.237
0.378
age_group_5_60to64 0.2676 0.044 6.062 0.000 0.181
0.354
age_group_5_65_over 0.3648 0.052 7.082 0.000 0.264
0.466

==============================================================================
Omnibus: 22.130 Durbin-Watson: 1.879
Prob(Omnibus): 0.000 Jarque-Bera (JB): 23.546
Skew: 0.430 Prob(JB): 7.71e-06
Kurtosis: 3.003 Cond. No. 5.83e+15
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The smallest eigenvalue is 3.65e-29. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.

---

[285]: model19 = sm.ols(data=merit_raises_combined_salaried_regression, formula ='
'performance_rating ~ race_grouping_white + race_grouping_person_of_color +
'age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 +
'age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
'age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
'age_group_5_65_over')
result19 = model19.fit()
result19.summary()

[285]: <class 'statsmodels.iolib.summary.Summary'>

###

OLS Regression Results

==============================================================================
Dep. Variable: performance_rating  R-squared: 0.070
Model: OLS Adj. R-squared: 0.056
Method: Least Squares F-statistic: 5.124
Date: Wed, 06 Nov 2019 Prob (F-statistic): 8.91e-08
Time: 10:27:49 Log-Likelihood: -208.27
No. Observations: 763 AIC: 440.5
Df Residuals: 751 BIC: 496.2
Df Model: 11
Covariance Type: nonrobust
==============================================================================
coef std err t P>|t|
[0.025 0.975]

---
Intercept 3.208 3.500 3.3538 0.075 45.011 0.000 0.075 45.011 0.000
race_grouping_white -0.279 0.043 -0.1183 0.082 -1.443 0.149 -0.419 -0.087 -0.1183 0.082 -1.443 0.149 0.087 -0.1183 0.082 -1.443 0.149 -0.419 -0.087 -0.1183 0.082 -1.443 0.149 0.087 -0.1183 0.082 -1.443 0.149 -0.419 -0.087
race_grouping_person_of_color -0.419 -0.087 -0.2531 0.085 -2.995 0.003 -0.279 0.043 -0.2531 0.085 -2.995 0.003 0.043 -0.2531 0.085 -2.995 0.003 -0.279 0.043 -0.2531 0.085 -2.995 0.003 0.043 -0.2531 0.085 -2.995 0.003 -0.279 0.043
age_group_5_25_under 0.168 0.324 0.145 0.120 0.121 0.904 0.2464 0.040 6.189 0.000 0.040 6.189 0.000 0.2464 0.040 6.189 0.000 0.040 6.189 0.000 0.2464 0.040 6.189 0.000 0.040 6.189 0.000 0.2464 0.040 6.189 0.000 0.040 6.189 0.000
age_group_5_25to29 0.262 0.384 0.3714 0.032 11.469 0.000 0.27 0.384 0.3714 0.032 11.469 0.000 0.27 0.384 0.3714 0.032 11.469 0.000 0.27 0.384 0.3714 0.032 11.469 0.000 0.27 0.384 0.3714 0.032 11.469 0.000 0.27 0.384 0.3714 0.032 11.469 0.000
age_group_5_30to34 0.308 0.435 0.4239 0.034 12.295 0.000 0.306 0.435 0.4239 0.034 12.295 0.000 0.306 0.435 0.4239 0.034 12.295 0.000 0.306 0.435 0.4239 0.034 12.295 0.000 0.306 0.435 0.4239 0.034 12.295 0.000
age_group_5_35to39 0.356 0.492 0.3228 0.031 10.429 0.000 0.3228 0.031 10.429 0.000 0.3228 0.031 10.429 0.000 0.3228 0.031 10.429 0.000 0.3228 0.031 10.429 0.000 0.3228 0.031 10.429 0.000
age_group_5_40to44 0.328 0.454 0.3791 0.033 11.832 0.000 0.3791 0.033 11.832 0.000 0.3791 0.033 11.832 0.000 0.3791 0.033 11.832 0.000 0.3791 0.033 11.832 0.000 0.3791 0.033 11.832 0.000
age_group_5_45to49 0.325 0.454 0.4180 0.036 11.531 0.000 0.4180 0.036 11.531 0.000 0.4180 0.036 11.531 0.000 0.4180 0.036 11.531 0.000 0.4180 0.036 11.531 0.000 0.4180 0.036 11.531 0.000
age_group_5_50to54 0.347 0.489 0.3757 0.044 8.554 0.000 0.3757 0.044 8.554 0.000 0.3757 0.044 8.554 0.000 0.3757 0.044 8.554 0.000 0.3757 0.044 8.554 0.000 0.3757 0.044 8.554 0.000
age_group_5_55to59 0.290 0.462 0.4645 0.052 8.960 0.000 0.4645 0.052 8.960 0.000 0.4645 0.052 8.960 0.000 0.4645 0.052 8.960 0.000 0.4645 0.052 8.960 0.000 0.4645 0.052 8.960 0.000
age_group_5_60to64 0.363 0.566 0.566
==============================================================================
Omnibus: 15.402 Durbin-Watson: 1.897
Prob(Omnibus): 0.000 Jarque-Bera (JB): 15.937
Skew: 0.354 Prob(JB): 0.000346
Kurtosis: 3.028 Cond. No. 3.34e+15
==============================================================================
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 1.22e-28. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

[286]:
model20 = sm.ols(data=merit_raises_combined_salaried_regression, formula =
             'performance_rating ~ gender_Female + gender_Male + race_grouping_white +
             race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 +
             age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 +
             age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 +
             age_group_5_60to64 + age_group_5_65_over')
result20 = model20.fit()
```python
result20.summary()

[286]: <class 'statsmodels.iolib.summary.Summary'>
```

OLS Regression Results
==============================================================================
Dep. Variable:  performance_rating       R-squared:             0.075
Model:                 OLS                Adj. R-squared:         0.060
Method:             Least Squares         F-statistic:         5.031
Date:        Wed, 06 Nov 2019             Prob (F-statistic):  4.32e-08
Time:            10:27:49                 Log-Likelihood:    -206.34
Df Residuals:                750                BIC:                499.0
Df Model:                        12
Covariance Type:             nonrobust
==============================================================================
                  coef    std err          t      P>|t|      [0.025    0.975]
--------------------------------------------------------------------------------
 Intercept         2.3092     0.051    45.135      0.000     2.209     2.410
 gender_Female       1.1315     0.028   40.697      0.000     1.077     1.186
             1.122    1.234
 gender_Male         1.1777     0.029   41.247      0.000     1.122     1.234
 race_grouping_white -0.1256     0.082    -1.533     0.126    -0.286     0.035
 race_grouping_person_of_color -0.2545     0.084    -3.017     0.003    -0.420    -0.089
 age_group_5_25_under -0.0729     0.119    -0.610     0.542    -0.307     0.162
 age_group_5_25to29    0.1473     0.040     3.716      0.000     0.069     0.225
 age_group_5_30to34    0.2187     0.031     7.146      0.000     0.159     0.279
 age_group_5_35to39    0.2665     0.032     8.329      0.000     0.204     0.329
 age_group_5_40to44    0.3129     0.035     9.057      0.000     0.245     0.381
 age_group_5_45to49    0.2210     0.038     5.838      0.000     0.147     0.295
 age_group_5_50to54    0.2823     0.033     8.679      0.000     0.218     0.346
 age_group_5_55to59    0.3102     0.036     8.637      0.000     0.240     0.381
```

166
Strong multicollinearity problems or that the design matrix is singular. The smallest eigenvalue is 1.63e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified. 
[2] The smallest eigenvalue is 1.63e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
[287]: news_hourly_regression = ["department", 'gender', 'race_ethnicity', 'current_base_pay', 'job_profile_current',
news_hourly_regression = pd.get_dummies(news_hourly_regression,
[288]: rename(columns={'race_grouping_person of color':
news_hourly_regression = news_hourly_regression.
```
result21 = model21.fit()
result21.summary()

[288]: <class 'statsmodels.iolib.summary.Summary'>
   ""
   OLS Regression Results
   ===============================================================================
   Dep. Variable: current_base_pay  R-squared: 0.043
   Model: OLS  Adj. R-squared: 0.040
   Method: Least Squares  F-statistic: 12.81
   Date: Wed, 06 Nov 2019  Prob (F-statistic): 3.60e-06
   Time: 10:27:49  Log-Likelihood: -6930.6
   No. Observations: 574  AIC: 1.387e+04
   Df Residuals: 571  BIC: 1.388e+04
   Df Model: 2
   Covariance Type: nonrobust
   ===============================================================================
   coef  std err  t  P>|t|
   0.025  0.975
   --------------------------------------------------------------------------------
   Intercept 1.271e+05  7897.372  16.092 0.000
   1.12e+05  1.43e+05
   race_grouping_white -6301.9244  8174.557 -0.771 0.441
   -2.24e+04  9753.945
   race_grouping_person_of_color -2.661e+04  8682.201 -3.065 0.002
   -4.37e+04 -9560.605
   ===============================================================================
   Omnibus: 128.063  Durbin-Watson: 1.632
   Prob(Omnibus): 0.000  Jarque-Bera (JB): 248.772
   Kurtosis: 5.030  Cond. No. 9.91
   ===============================================================================

   Warnings:
   [1] Standard Errors assume that the covariance matrix of the errors is correctly
   specified.
   ""

[289]: model22 = sm.ols(data=news_hourly_regression, formula = 'current_base_pay ~
race_grouping_white + race_grouping_person_of_color')
result22 = model22.fit()
result22.summary()

[289]: <class 'statsmodels.iolib.summary.Summary'>
   ""
   OLS Regression Results
Dep. Variable: current_base_pay  R-squared: 0.051
Model: OLS  Adj. R-squared: 0.030
Method: Least Squares  F-statistic: 2.484
Date: Wed, 06 Nov 2019  Prob (F-statistic): 0.0889
Time: 10:27:49  Log-Likelihood: -369.15
No. Observations: 96  AIC: 744.3
Df Residuals: 93  BIC: 752.0
Df Model: 2
Covariance Type: nonrobust

==============================================================================
                  coef     std err         t      P>|t|    [0.025     0.975]
------------------------------------------------------------------------------
Intercept        39.2300     8.131       4.825      0.000   23.084     55.376
race_grouping_white -3.6811     8.257      -0.446      0.657  -20.077     12.715
race_grouping_person_of_color -9.0990     8.397      -1.084      0.281 -25.775     7.577
------------------------------------------------------------------------------

Omnibus: 5.387  Durbin-Watson: 1.792
Prob(Omnibus): 0.068  Jarque-Bera (JB): 4.797
Skew: 0.527  Prob(JB): 0.0909
Kurtosis: 3.296  Cond. No. 15.1

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

```python
[290]: model23 = sm.ols(data=news_hourly_regression, formula = 'current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color')
result23 = model23.fit()
result23.summary()
```

### OLS Regression Results

```
Dep. Variable: current_base_pay  R-squared: 0.065
Model: OLS  Adj. R-squared: 0.034
Method: Least Squares  F-statistic: 2.116
Date: Wed, 06 Nov 2019  Prob (F-statistic): 0.104
```
Time: 10:27:49  Log-Likelihood: -368.44
No. Observations: 96   AIC: 744.9
Df Residuals: 92   BIC: 755.1
Df Model: 3
covariance Type: nonrobust

==============================================================================
 coef std err  t    P>|t|    [0.025 0.975]
==============================================================================
 Intercept      25.1888   5.473  4.603  0.000   14.319 36.058
gender_Female   14.0412   2.829  4.964  0.000    8.423 19.659
gender_Male     11.1476   3.171  3.516  0.001    4.851 17.445
race_grouping_white -2.6412   8.289 -0.319  0.751  -19.104 13.821
race_grouping_person_of_color -8.1345  8.422 -0.966  0.337  -24.861 8.592

==============================================================================
 Omnibus:  4.237  Durbin-Watson:  1.806
 Prob(Omnibus): 0.120  Jarque-Bera (JB): 3.664
 Skew: 0.465  Prob(JB): 0.160
 Kurtosis: 3.226 Cond. No. 8.67e+15

==============================================================================
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 2.71e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
cell['predicted'] = result23.predict(new_news_hourly_regression)
```
\[
\begin{array}{ccc}
\text{race_grouping_person_of_color} & \text{age} & \text{predicted} \\
0 & 0 & 40 & 36.59 \\
1 & 0 & 40 & 33.70 \\
2 & 1 & 40 & 31.10 \\
3 & 1 & 40 & 28.20 \\
\end{array}
\]

```
model24 = sm.ols(data=news_hourly_regression, formula = 'current_base_pay ~
gender_Female + gender_Male + age_group_5_25_under + age_group_5_25to29 +
age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 +
age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 +
age_group_5_60to64 + age_group_5_65_over')
result24 = model24.fit()
result24.summary()
```

```
<class 'statsmodels.iolib.summary.Summary'>
```
10.015
age_group_5_45to49  11.8836  3.640  3.265  0.002  4.647
19.120
age_group_5_50to54  3.3536  3.606  0.930  0.355  -3.816
10.523
age_group_5_55to59  1.9492  3.691  0.528  0.599  -5.389
9.288
age_group_5_60to64  7.9013  4.207  1.878  0.064  -0.464
16.267
age_group_5_65_over  11.5193  4.225  2.726  0.008  3.118
19.921

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The smallest eigenvalue is 3e-31. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.

[293]: model25 = sm.ols(data=news_hourly_regression, formula = 'current_base_pay ~
race_grouping_white + race_grouping_person_of_color + age_group_5_25_under +
age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 +
age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 +
age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over')
result25 = model25.fit()
result25.summary()

[293]: <class 'statsmodels.iolib.summary.Summary'>

OLS Regression Results

Dep. Variable: current_base_pay  R-squared:  0.337
Model: OLS  Adj. R-squared:  0.250
Method: Least Squares  F-statistic:  3.876
Date: Wed, 06 Nov 2019  Prob (F-statistic):  0.000154
Time: 10:27:50  Log-Likelihood:  -351.94
No. Observations:  96  AIC:  727.9
Df Residuals:  84  BIC:  758.7
Df Model: 11
Covariance Type: nonrobust

==============================================================================
Omnibus:  0.505 Durbin-Watson:  1.922
Prob(Omnibus):  0.777 Jarque-Bera (JB):  0.653
Skew:  0.092 Prob(JB):  0.721
Kurtosis:  2.640 Cond. No.  2.33e+16
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The smallest eigenvalue is 3e-31. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.

"""
|                | coef     | std err |     t  | P>|t| |
|----------------|----------|---------|--------|-----|
| Intercept      | 42.6892  | 6.752   | 6.322  | 0.000 |
| race_grouping_white | -9.9644  | 7.492   | -1.330 | 0.187 |
| race_grouping_person_of_color | -12.5342 | 7.657   | -1.637 | 0.105 |
| age_group_5_25_under | -6.1759  | 2.703   | -2.285 | 0.025 |
| age_group_5_25to29 | -2.6518  | 2.389   | -1.110 | 0.270 |
| age_group_5_30to34 | -0.7425  | 2.971   | -0.250 | 0.803 |
| age_group_5_35to39 | -0.0048  | 3.496   | -0.001 | 0.999 |
| age_group_5_40to44 | 5.3819   | 3.318   | 1.622  | 0.108 |
| age_group_5_45to49 | 14.5738  | 3.707   | 3.932  | 0.000 |
| age_group_5_50to54 | 5.8780   | 3.695   | 1.591  | 0.115 |
| age_group_5_55to59 | 2.5081   | 3.707   | 0.677  | 0.501 |
| age_group_5_60to64 | 9.2652   | 4.341   | 2.134  | 0.036 |
| age_group_5_65_over | 14.6572  | 4.305   | 3.404  | 0.001 |

Omnibus: 1.450  Durbin-Watson: 1.945
Prob(Omnibus): 0.484  Jarque-Bera (JB): 1.513
Skew: 0.255  Prob(JB): 0.469
Kurtosis: 2.657  Cond. No. 1.10e+16

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 1.33e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.
```python
model26 = sm.ols(data=news_hourly_regression, formula = 'current_base_pay ~
genre_Female + genre_Male + race_grouping_white +
race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 +
age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 +
age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 +
age_group_5_60to64 + age_group_5_65_over')
result26 = model26.fit()
result26.summary()
```

```
<class 'statsmodels.iolib.summary.Summary'>
```

```
OLS Regression Results
==============================================================================
Dep. Variable:  current_base_pay     R-squared:  0.351
Model:                OLS     Adj. R-squared: 0.257
Method:                Least Squares     F-statistic:  3.736
Date:             Wed, 06 Nov 2019     Prob (F-statistic): 0.000160
No. Observations:             96     AIC:  727.8
Df Residuals:              83     BIC:  761.2
Df Model:                12
Covariance Type:          nonrobust
==============================================================================
                      coef     std err          t       P>|t|    [0.025  0.975]
--------------------------------------------------------------------------------
Intercept             28.2477     4.693         6.019      0.000     18.913    37.582
gender_Female        -15.7092     2.436        -6.449      0.000    -20.554   -10.864
gender_Male           12.5385     2.807         4.466      0.000      6.955   18.122
race_grouping_white  -13.6467     2.625         -5.179      0.000     -18.821   -8.472
race_grouping_person_of_color -11.0211     7.705        -1.430      0.156     -26.345    4.303
age_group_5_25_under  -1.4497     3.439        -0.421      0.674     -8.291   5.391
age_group_5_30to34    4.1931     3.261         1.286      0.202
age_group_5_40to44    4.1931     3.261         1.286      0.202
```

```
-2.293   10.679
age_group_5_45to49   12.4474 3.696   3.367   0.001
5.095   19.800
age_group_5_50to54   4.1710 3.648   1.143   0.256
-3.084   11.426
age_group_5_55to59   2.1935 3.727   0.588   0.558
-5.220   9.607
age_group_5_60to64   7.9878 4.282   1.865   0.066
-0.530   16.505
age_group_5_65_over  12.6515 4.276   2.959   0.004
4.147   21.156
==============================================================================
Omnibus: 0.670 Durbin-Watson: 1.944
Prob(Omnibus): 0.715 Jarque-Bera (JB): 0.804
Skew: 0.145 Prob(JB): 0.669
Kurtosis: 2.658 Cond. No. 1.52e+16
==============================================================================
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 9.29e-31. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```
[295]: model27 = sm.ols(data=news_hourly_regression, formula='current_base_pay ~
gender_Female + gender_Male + race_grouping_white +
race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 +
age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 +
age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 +
age_group_5_60to64 + age_group_5_65_over + tier_Tier_1 + tier_Tier_2 +
tier_Tier_3 + tier_Tier_4')
result27 = model27.fit()
result27.summary()
```

[295]: <class 'statsmodels.iolib.summary.Summary'>
```
OLS Regression Results
==============================================================================
Dep. Variable: current_base_pay R-squared: 0.425
Model: OLS Adj. R-squared: 0.309
Method: Least Squares F-statistic: 3.656
Date: Wed, 06 Nov 2019 Prob (F-statistic): 5.89e-05
Time: 10:27:50 Log-Likelihood: -345.05
No. Observations: 96 AIC: 724.1
Df Residuals: 79 BIC: 767.7
Df Model: 16
Covariance Type: nonrobust
```

175
|                | coef   | std err | t      | P>|t|  |
|----------------|--------|---------|--------|--------|
| Intercept      | 34.4064| 5.313   | 6.476  | 0.000  |
| gender_Female  | 19.0722| 2.796   | 6.822  | 0.000  |
| gender_Male    | 15.3342| 2.992   | 5.125  | 0.000  |
| race_grouping_white | -7.5095 | 7.386 | -1.017 | 0.312  |
| race_grouping_person_of_color | -11.2049 | 7.615 | -1.471 | 0.145  |
| age_group_5_25_under | -7.8299 | 2.675 | -2.927 | 0.004  |
| age_group_5_25to29  | -5.5331 | 2.385 | -2.320 | 0.023  |
| age_group_5_30to34  | -1.8309 | 2.974 | -0.616 | 0.540  |
| age_group_5_35to39  | -8.2709 | 3.507 | -2.370 | 0.018  |
| age_group_5_40to44  | -1.2639 | 3.350 | -0.377 | 0.707  |
| age_group_5_45to49  | 13.6650 | 3.609 | 3.786  | 0.000  |
| age_group_5_50to54  | 26.481  | 20.849 | 1.271  | 0.207  |
| age_group_5_55to59  | -1.504  | 12.777 | -0.120 | 0.906  |
| age_group_5_60to64  | 4.2840  | 3.700  | 1.158  | 0.250  |
| age_group_5_65_over | 7.7412  | 4.225  | 1.832  | 0.071  |
| tier_Tier_1     | -9.5152 | 4.919  | -1.934 | 0.057  |
| tier_Tier_2     | -21.810 | -4.708 | -4.600 | 0.031  |
| tier_Tier_3     | 17.957  | -0.879 | 2.049  | 0.040  |
| tier_Tier_4     | -7.6206 | 10.569 | -0.721 | 0.473  |

Omnibus: 0.381  Durbin-Watson: 1.809
Prob(Omnibus): 0.827  Jarque-Bera (JB): 0.242
```
model28 = sm.ols(data=news_hourly_regression, formula = 'current_base_pay ~ gender_Female + gender_Male + race_grouping_white +
race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 +
age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 +
age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 +
age_group_5_60to64 + age_group_5_65_over + tier_Tier_1 + tier_Tier_2 +
tier_Tier_3 + tier_Tier_4 + years_of_service_grouped_0 +
years_of_service_grouped_1to2 + years_of_service_grouped_3to5 +
years_of_service_grouped_6to10 + years_of_service_grouped_11to15 +
years_of_service_grouped_16to20 + years_of_service_grouped_21to25 +
years_of_service_grouped_25_over')
result28 = model28.fit()
result28.summary()
```

```
<class 'statsmodels.iolib.summary.Summary'>
```
```
O LS Regression Results
==============================================================================
Dep. Variable: current_base_pay R-squared: 0.443
Model: OLS Adj. R-squared: 0.266
Method: Least Squares F-statistic: 2.494
Date: Wed, 06 Nov 2019 Prob (F-statistic): 0.00173
Time: 10:27:50 Log-Likelihood: -343.52
No. Observations: 96 AIC: 735.0
Df Residuals: 72 BIC: 796.6
Df Model: 23
Covariance Type: nonrobust
==============================================================================
                  coef   std err      t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept         32.4885     5.312     6.116      0.000      21.900     43.077
gender_Female     18.2562     2.768     6.596      0.000     12.738     23.774
```
<table>
<thead>
<tr>
<th>Variable</th>
<th>Coefficient</th>
<th>Standard Error</th>
<th>t-Value</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>gender_Male</td>
<td>14.2324</td>
<td>3.105</td>
<td>4.584</td>
<td>0.000</td>
</tr>
<tr>
<td>race_grouping_white</td>
<td>-8.7651</td>
<td>7.960</td>
<td>-1.101</td>
<td>0.275</td>
</tr>
<tr>
<td>race_grouping_person_of_color</td>
<td>-12.3227</td>
<td>8.173</td>
<td>-1.508</td>
<td>0.136</td>
</tr>
<tr>
<td>age_group_5_25_under</td>
<td>-10.2986</td>
<td>4.405</td>
<td>-2.338</td>
<td>0.022</td>
</tr>
<tr>
<td>age_group_5_25to29</td>
<td>-7.6966</td>
<td>4.002</td>
<td>-1.923</td>
<td>0.058</td>
</tr>
<tr>
<td>age_group_5_30to34</td>
<td>-3.6324</td>
<td>3.639</td>
<td>-0.998</td>
<td>0.322</td>
</tr>
<tr>
<td>age_group_5_35to39</td>
<td>-2.4335</td>
<td>3.749</td>
<td>-0.649</td>
<td>0.518</td>
</tr>
<tr>
<td>age_group_5_40to44</td>
<td>5.1931</td>
<td>3.533</td>
<td>1.470</td>
<td>0.146</td>
</tr>
<tr>
<td>age_group_5_45to49</td>
<td>13.9918</td>
<td>4.500</td>
<td>3.109</td>
<td>0.003</td>
</tr>
<tr>
<td>age_group_5_50to54</td>
<td>7.0912</td>
<td>4.119</td>
<td>1.721</td>
<td>0.089</td>
</tr>
<tr>
<td>age_group_5_55to59</td>
<td>5.1000</td>
<td>4.027</td>
<td>1.267</td>
<td>0.209</td>
</tr>
<tr>
<td>age_group_5_60to64</td>
<td>10.3023</td>
<td>5.229</td>
<td>1.970</td>
<td>0.053</td>
</tr>
<tr>
<td>age_group_5_65_over</td>
<td>14.8714</td>
<td>4.638</td>
<td>3.207</td>
<td>0.002</td>
</tr>
<tr>
<td>tier_Tier_1</td>
<td>-9.6000</td>
<td>5.224</td>
<td>-1.838</td>
<td>0.070</td>
</tr>
<tr>
<td>tier_Tier_2</td>
<td>-12.9480</td>
<td>4.517</td>
<td>-2.867</td>
<td>0.005</td>
</tr>
<tr>
<td>tier_Tier_3</td>
<td>-9.7998</td>
<td>4.564</td>
<td>-2.147</td>
<td>0.035</td>
</tr>
<tr>
<td>tier_Tier_4</td>
<td>-6.9480</td>
<td>12.118</td>
<td>-0.573</td>
<td>0.568</td>
</tr>
<tr>
<td>years_of_service_grouped_0</td>
<td>5.5195</td>
<td>4.055</td>
<td>1.361</td>
<td>0.178</td>
</tr>
<tr>
<td>years_of_service_grouped_1to2</td>
<td>7.0368</td>
<td>3.627</td>
<td>1.940</td>
<td>0.056</td>
</tr>
<tr>
<td>years_of_service_grouped_3to5</td>
<td>4.9818</td>
<td>3.862</td>
<td>1.290</td>
<td>0.201</td>
</tr>
<tr>
<td>years_of_service_grouped_6to10</td>
<td>5.3426</td>
<td>3.056</td>
<td>1.748</td>
<td>0.085</td>
</tr>
<tr>
<td>years_of_service_grouped_11to15</td>
<td>5.0292</td>
<td>3.876</td>
<td>1.298</td>
<td>0.199</td>
</tr>
<tr>
<td>years_of_service_grouped_16to20</td>
<td>-0.1816</td>
<td>3.514</td>
<td>-0.052</td>
<td>0.959</td>
</tr>
<tr>
<td>years_of_service_grouped_21to25</td>
<td>3.0822</td>
<td>5.187</td>
<td>0.594</td>
<td>0.554</td>
</tr>
</tbody>
</table>
merit_raises_combined_hourly_regression = merit_raises_combined[(merit_raises_combined['dept'] == 'News') & (merit_raises_combined['pay_rate_type'] == 'Hourly')]
merit_raises_combined_hourly_regression = pd.get_dummies(merit_raises_combined_hourly_regression, columns=['gender', 'race_grouping', 'age_group_5'])

model29 = sm.ols(data=merit_raises_combined_hourly_regression, formula='base_pay_change ~ gender_Female + gender_Male')
result29 = model29.fit()
result29.summary()

---

| Dep. Variable: | base_pay_change | R-squared: | 0.010 |
| Model:         | OLS             | Adj. R-squared: | 0.001 |
Method: Least Squares F-statistic: 1.130
Date: Wed, 06 Nov 2019 Prob (F-statistic): 0.290
Time: 10:27:50 Log-Likelihood: -217.43
No. Observations: 119 AIC: 438.9
Df Residuals: 117 BIC: 444.4
Df Model: 1
Covariance Type: nonrobust

==============================================================================

| coef  | std err | t     | P>|t|    | [0.025| 0.975] |
|-------|---------|-------|--------|-------|--------|
| Intercept 1.0256 | 0.095  | 10.816 | 0.000  | 0.838  | 1.213  |
| gender_Female 0.6640 | 0.140  | 4.737  | 0.000  | 0.386  | 0.942  |
| gender_Male 0.3616 | 0.159  | 2.273  | 0.025  | 0.047  | 0.677  |

==============================================================================

Omnibus: 140.664 Durbin-Watson: 1.822
Prob(Omnibus): 0.000 Jarque-Bera (JB): 3520.132
Skew: 4.181 Prob(JB): 0.00
Kurtosis: 28.299 Cond. No. 2.84e+15

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 2.25e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
[299]: model30 = sm.ols(data=merit_raises_combined_hourly_regression, formula = 'base_pay_change ~ race_grouping_white + race_grouping_person_of_color')
result30 = model30.fit()
result30.summary()
```

180
Df Residuals: 117  BIC: 442.0
Df Model: 1
Covariance Type: nonrobust

==============================================================================
              coef    std err          t      P>|t|    [0.025]   [0.975]
==============================================================================
Intercept     0.9759      0.099      9.846      0.000     0.780      1.172
race_grouping_white  0.7693      0.138      5.583      0.000     0.496      1.042
race_grouping_person_of_color  0.2066      0.174      1.190      0.236    -0.137      0.550

==============================================================================
Omniibus: 140.033  Durbin-Watson: 1.726
Prob(Omniibus): 0.000  Jarque-Bera (JB): 3604.750
Skew: 4.131  Prob(JB): 0.00
Kurtosis: 28.666  Cond. No. 4.70e+15
==============================================================================
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The smallest eigenvalue is 8.62e-30. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.

model31 = sm.ols(data=merit_raises_combined_hourly_regression, formula =
'base_pay_change ~ gender_Female + gender_Male + race_grouping_white +
race_grouping_person_of_color')
result31 = model31.fit()
result31.summary()
| coef    | std err | t      | P>|t| |
|---------|---------|--------|------|
| 0.7239  | 0.075   | 9.628  | 0.000|
| 0.4726  | 0.143   | 3.312  | 0.001|
| 0.2512  | 0.153   | 1.645  | 0.103|
| 0.6242  | 0.142   | 4.386  | 0.000|
| 0.0996  | 0.168   | 0.594  | 0.554|

**Intercept**

**gender_Female**

**gender_Male**

**race_grouping_white**

**race_grouping_person_of_color**

---

**Warnings:**

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

[2] The smallest eigenvalue is 7.58e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
[301]: new_reason_for_change_combined_regression = pd.DataFrame({'gender_Female': [1,0,1,0], 'gender_Male': [0,1,0,1], 'race_grouping_white': [1,1,0,0], 'race_grouping_person_of_color': [0,0,1,1])
new_reason_for_change_combined_regression['predicted'] = result31.
new_reason_for_change_combined_regression
```

```
gender_Female  gender_Male  race_grouping_white  \
0          1            0            1
1          0            1            1
2          1            0            0
3          0            1            0

race_grouping_person_of_color  predicted
0          0            1.82
1          0            1.60
2          1            1.30
3          1            1.07
```
model32 = sm.ols(data=merit_raises_combined_hourly_regression, formula = 'base_pay_change ~ gender_Female + gender_Male + age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to49 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over')
result32 = model32.fit()
result32.summary()

---

```
OLS Regression Results

Dep. Variable: base_pay_change   R-squared: 0.076
Model: OLS Adj. R-squared: -0.010
Method: Least Squares F-statistic: 0.8829
Date: Wed, 06 Nov 2019 Prob (F-statistic): 0.552
Time: 10:27:50 Log-Likelihood: -213.33
No. Observations: 119 AIC: 448.7
Df Residuals: 108 BIC: 479.2
Df Model: 10
Covariance Type: nonrobust

==============================================================================
                      coef    std err          t      P>|t|      [0.025
---------           --------  --------  ---------  --------  --------  --------
Intercept         0.9858     0.109     9.020      0.000     0.769
gender_Female     0.6158     0.155     3.964      0.000     0.308
gender_Male       0.3701     0.173     2.135      0.035     0.027
age_group_5_25_under -0.9278     0.814    -1.140      0.257    -2.541
age_group_5_25to29  0.1217     0.330     0.369      0.713    -0.532
age_group_5_30to34  0.1034     0.365     0.284      0.777    -0.619
age_group_5_35to39 -0.1446     0.429    -0.337      0.737    -0.996
age_group_5_40to44  0.2296     0.429     0.535      0.594    -0.622
age_group_5_45to49  0.0921     0.381     0.242      0.809    -0.663
age_group_5_50to54 -0.1722     0.422    -0.409      0.684    -1.008
age_group_5_55to59
age_group_5_60to64
age_group_5_65_over

==============================================================================
```
<table>
<thead>
<tr>
<th>age_group_5_55to59</th>
<th>-0.1867</th>
<th>0.447</th>
<th>-0.418</th>
<th>0.677</th>
<th>-1.073</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.699</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>age_group_5_60to64</td>
<td>1.8448</td>
<td>0.708</td>
<td>2.606</td>
<td>0.010</td>
<td>0.442</td>
</tr>
<tr>
<td>3.248</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>age_group_5_65_over</td>
<td>0.0256</td>
<td>0.639</td>
<td>0.040</td>
<td>0.968</td>
<td>-1.241</td>
</tr>
<tr>
<td>1.292</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 1.28e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
[303]: model33 = sm.ols(data=merit_raises_combined_hourly_regression, formula = 'base_pay_change ~ race_grouping_white + race_grouping_person_of_color +
              age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 +
              age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
              age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
              age_group_5_65_over')
result33 = model33.fit()  
result33.summary()
```

```python
[303]: <class 'statsmodels.iolib.summary.Summary'>
```
<table>
<thead>
<tr>
<th></th>
<th>Coefficient</th>
<th>Standard Error</th>
<th>z-score</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>0.9191</td>
<td>0.115</td>
<td>7.971</td>
<td>0.000</td>
</tr>
<tr>
<td>race_grouping_white</td>
<td>0.7541</td>
<td>0.148</td>
<td>5.084</td>
<td>0.000</td>
</tr>
<tr>
<td>race_grouping_person_of_color</td>
<td>0.1650</td>
<td>0.191</td>
<td>0.866</td>
<td>0.389</td>
</tr>
<tr>
<td>age_group_5_25_under</td>
<td>-1.1631</td>
<td>0.806</td>
<td>-1.444</td>
<td>0.152</td>
</tr>
<tr>
<td>age_group_5_25to29</td>
<td>0.2934</td>
<td>0.327</td>
<td>0.898</td>
<td>0.371</td>
</tr>
<tr>
<td>age_group_5_30to34</td>
<td>-0.0743</td>
<td>0.355</td>
<td>-0.209</td>
<td>0.835</td>
</tr>
<tr>
<td>age_group_5_35to39</td>
<td>-0.0445</td>
<td>0.428</td>
<td>-0.104</td>
<td>0.918</td>
</tr>
<tr>
<td>age_group_5_40to44</td>
<td>0.1334</td>
<td>0.425</td>
<td>0.314</td>
<td>0.754</td>
</tr>
<tr>
<td>age_group_5_45to49</td>
<td>0.1278</td>
<td>0.375</td>
<td>0.341</td>
<td>0.734</td>
</tr>
<tr>
<td>age_group_5_50to54</td>
<td>-0.1457</td>
<td>0.409</td>
<td>-0.356</td>
<td>0.723</td>
</tr>
<tr>
<td>age_group_5_55to59</td>
<td>-0.1780</td>
<td>0.440</td>
<td>-0.405</td>
<td>0.687</td>
</tr>
<tr>
<td>age_group_5_60to64</td>
<td>1.7119</td>
<td>0.704</td>
<td>2.433</td>
<td>0.017</td>
</tr>
<tr>
<td>age_group_5_65_over</td>
<td>0.2583</td>
<td>0.636</td>
<td>0.406</td>
<td>0.685</td>
</tr>
</tbody>
</table>

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 7.27e-31. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```
[304]: model34 = sm.ols(data=merit_raises_combined_hourly_regression, formula='base_pay_change ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over')
result34 = model34.fit()
```
```python
result34.summary()
```

```
[304]: <class 'statsmodels.iolib.summary.Summary'>
""

OLS Regression Results
==============================================================================
Dep. Variable:    base_pay_change   R-squared:           0.099
Model:              OLS              Adj. R-squared:  0.006
Method:              Least Squares    F-statistic:     1.069
Date:                Wed, 06 Nov 2019 Prob (F-statistic):  0.393
Time:                10:27:51        Log-Likelihood: -211.80
No. Observations:    119             AIC:                     447.6
Df Residuals:        107             BIC:                     480.9
Df Model:            11              Covariance Type: nonrobust
==============================================================================
                 coef     std err          t      P>|t|    [0.025  0.975]
--------------------------------------------------------------------------------
Intercept       0.6987     0.088     7.906      0.000    0.523  0.874
gender_Female  0.3987     0.164     2.437      0.016    0.074  0.723
gender_Male    0.2999     0.169     1.779      0.078   -0.034  0.634
race_grouping_white 0.6296     0.158     3.992      0.000    0.317  0.942
race_grouping_person_of_color 0.0691     0.188     0.367      0.714   -0.304  0.442
age_group_5_25_under -1.1511     0.815    -1.412      0.161   -2.768  0.465
age_group_5_25to29  0.2459     0.339     0.725      0.470  -0.427    0.919
age_group_5_30to34 -0.0636     0.372    -0.171      0.865   -0.917  0.791
age_group_5_35to39 -0.0632     0.431    -0.147      0.884  -0.917  0.791
age_group_5_40to44  0.1242     0.429     0.289      0.773  -0.726    0.975
age_group_5_45to49  0.0959     0.379     0.253      0.801  -0.655    0.847
age_group_5_50to54 -0.1909     0.418    -0.456      0.649  -1.020    0.639
age_group_5_55to59 -0.1884     0.444    -0.425      0.672  -1.068    0.691
```
age_group_5_60to64 1.6827 0.706 2.382 0.019
0.282 3.083
age_group_5_65_over 0.2071 0.645 0.321 0.749
-1.072 1.486

Omnibus: 142.272 Durbin-Watson: 1.701
Prob(Omnibus): 0.000 Jarque-Bera (JB): 4047.261
Skew: 4.186 Prob(JB): 0.00
Kurtosis: 30.316 Cond. No. 1.99e+16

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 6.75e-31. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```
[305]: model35 = sm.ols(data=merit_raises_combined_hourly_regression, formula ='
                   ~'performance_rating ~ gender_Female + gender_Male')
result35 = model35.fit()
result35.summary()
```

```python
OLS Regression Results

Dep. Variable: performance_rating R-squared: 0.004
Model: OLS Adj. R-squared: -0.005
Method: Least Squares F-statistic: 0.4057
Date: Wed, 06 Nov 2019 Prob (F-statistic): 0.526
No. Observations: 111 AIC: 84.27
Df Residuals: 109 BIC: 89.69
Df Model: 1
Covariance Type: nonrobust

 coef std err t P>|t| [0.025 0.975]
Intercept 2.3463 0.023 103.057 0.000 2.301 2.391
gender_Female 1.1949 0.033 35.693 0.000 1.075 1.319
gender_Male 1.1514 0.038 30.021 0.000 1.151 1.227
```

187
```python
[306]:
model36 = sm.ols(data=merit_raises_combined_hourly_regression, formula='performance_rating ~ race_grouping_white + race_grouping_person_of_color')
result36 = model36.fit()
result36.summary()
```

```
Omnibus: 7.442 Durbin-Watson: 2.088
Prob(Omnibus): 0.024 Jarque-Bera (JB): 6.902
Skew: 0.544 Prob(JB): 0.0317
Kurtosis: 2.444 Cond. No. 3.47e+15

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 1.41e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```
Kurtosis: 2.451  Cond. No. 3.50e+15

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 1.44e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
[307]: model37 = sm.ols(data=merit_raises_combined_hourly_regression, formula='performance_rating ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color')
result37 = model37.fit()
result37.summary()
```

```
OLS Regression Results
==============================================================================
Dep. Variable: performance_rating  R-squared: 0.044
Model: OLS  Adj. R-squared: 0.026
Method: Least Squares  F-statistic: 2.484
Date: Wed, 06 Nov 2019  Prob (F-statistic): 0.0882
No. Observations: 111  AIC: 81.69
Df Residuals: 108  BIC: 89.82
Df Model: 2
Covariance Type: nonrobust
==============================================================================
          coef       std err          t      P>|t|    [0.025]     [0.975]
Intercept 1.7480     0.018    98.864 0.000     1.713      1.783
gender_Female 0.8811     0.034    25.816 0.000    0.813       0.949
gender_Male 0.8668     0.037    23.645 0.000     0.794      0.940
race_grouping_white 0.9501     0.034   27.947 0.000    0.883      1.018
race_grouping_person_of_color 0.7979     0.039    20.276 0.000     0.720      0.876
==============================================================================
 Omnibus: 5.045  Durbin-Watson: 2.099
Prob(Omnibus): 0.080  Jarque-Bera (JB): 4.403
```

189
Skew: 0.402  Prob(JB): 0.111
Kurtosis: 2.448  Cond. No. 2.18e+16

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 4.93e-31. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
model38 = sm.ols(data=merit_raises_combined_hourly_regression, formula = 'performance_rating ~ gender_Female + gender_Male + age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over')
result38 = model38.fit()
result38.summary()
```

---

### OLS Regression Results

<table>
<thead>
<tr>
<th>Dep. Variable:</th>
<th>performance_rating</th>
<th>R-squared:</th>
<th>0.136</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model:</td>
<td>OLS</td>
<td>Adj. R-squared:</td>
<td>0.050</td>
</tr>
<tr>
<td>Method:</td>
<td>Least Squares</td>
<td>F-statistic:</td>
<td>1.574</td>
</tr>
<tr>
<td>Date:</td>
<td>Wed, 06 Nov 2019</td>
<td>Prob (F-statistic):</td>
<td>0.125</td>
</tr>
<tr>
<td>No. Observations:</td>
<td>111</td>
<td>AIC:</td>
<td>86.46</td>
</tr>
<tr>
<td>Df Residuals:</td>
<td>100</td>
<td>BIC:</td>
<td>116.3</td>
</tr>
<tr>
<td>Df Model:</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Covariance Type:</td>
<td>nonrobust</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

|          | coef    | std err | t      | P>|t| | [0.025|
|----------|---------|---------|--------|------|-------|
| Intercept| 2.1848  | 0.027   | 80.675 | 0.000| 2.131 |
| gender_Female| 1.1135 | 0.037   | 30.395 | 0.000| 1.041 |
| gender_Male | 1.0713 | 0.041   | 26.028 | 0.000| 0.990 |
| age_group_5_25_under | 0.0228 | 0.221 | 0.103 | 0.918 | -0.416 |
| age_group_5_25to29 | 0.1814 | 0.076 | 2.396 | 0.018| 0.031 |

---

190
<table>
<thead>
<tr>
<th>Age Group</th>
<th>Coefficient</th>
<th>Standard Error</th>
<th>t-Statistic</th>
<th>P-value</th>
<th>Normalized Coefficient</th>
</tr>
</thead>
<tbody>
<tr>
<td>age_group_5_25_under</td>
<td>0.2256</td>
<td>0.087</td>
<td>2.580</td>
<td>0.011</td>
<td>0.052</td>
</tr>
<tr>
<td>age_group_5_25to29</td>
<td>0.0520</td>
<td>0.101</td>
<td>0.513</td>
<td>0.609</td>
<td>-0.149</td>
</tr>
<tr>
<td>age_group_5_30to34</td>
<td>0.5228</td>
<td>0.098</td>
<td>5.360</td>
<td>0.000</td>
<td>0.329</td>
</tr>
<tr>
<td>age_group_5_30to39</td>
<td>0.2274</td>
<td>0.087</td>
<td>2.620</td>
<td>0.010</td>
<td>0.055</td>
</tr>
<tr>
<td>age_group_5_35to39</td>
<td>0.3303</td>
<td>0.100</td>
<td>3.303</td>
<td>0.001</td>
<td>0.132</td>
</tr>
<tr>
<td>age_group_5_40to44</td>
<td>0.3030</td>
<td>0.111</td>
<td>2.741</td>
<td>0.007</td>
<td>0.084</td>
</tr>
<tr>
<td>age_group_5_45to49</td>
<td>0.0492</td>
<td>0.182</td>
<td>0.270</td>
<td>0.788</td>
<td>-0.312</td>
</tr>
<tr>
<td>age_group_5_50to54</td>
<td>0.2702</td>
<td>0.144</td>
<td>1.878</td>
<td>0.063</td>
<td>-0.015</td>
</tr>
<tr>
<td>age_group_5_55to59</td>
<td>0.3990</td>
<td>0.225</td>
<td>1.786</td>
<td>0.075</td>
<td>0.101</td>
</tr>
<tr>
<td>age_group_5_60to64</td>
<td>0.0497</td>
<td>0.225</td>
<td>0.221</td>
<td>0.826</td>
<td>0.049</td>
</tr>
<tr>
<td>age_group_5_65_over</td>
<td>0.3303</td>
<td>0.100</td>
<td>3.303</td>
<td>0.001</td>
<td>0.132</td>
</tr>
</tbody>
</table>

Omnibus: 4.456  Durbin-Watson: 2.073
Prob(Omnibus): 0.108  Jarque-Bera (JB): 3.765
Skew: 0.354  Prob(JB): 0.152
Kurtosis: 2.440  Cond. No. 1.51e+16

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 8.12e-31. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
model39 = sm.ols(data=merit_raises_combined_hourly_regression, formula='performance_rating ~ race_grouping_white + race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over')
result39 = model39.fit()
result39.summary()
```
No. Observations: 111  AIC:  82.82  
Df Residuals: 100  BIC:  112.6  
Df Model: 10  
Covariance Type: nonrobust  

================================================================================
=================
 coef    std err       t      P>|t|    [0.025    0.975]
--------------------------------------------------------------------------------
   Intercept 2.1685     0.028  77.356     0.000  2.113    2.224
 race_grouping_white     1.1544     0.034  33.766     0.000     1.087    1.222
 race_grouping_person_of_color  1.0141     0.044  23.298     0.000    0.928    1.100
 age_group_5_25_under    -0.0229     0.218  -0.105     0.917     -0.456    0.410
 age_group_5_25to29     0.2181     0.075   2.921     0.004     0.070    0.366
 age_group_5_30to34    0.1877     0.084  2.230     0.028    0.021    0.355
 age_group_5_35to39   -0.1190     0.281  -0.428     0.673    -0.676    0.438
 age_group_5_40to44    0.5005     0.096  5.205     0.000     0.310    0.691
 age_group_5_45to49   0.2335     0.085  2.741     0.007     0.064    0.402
 age_group_5_50to54    0.3372     0.096  3.522     0.001     0.147    0.527
 age_group_5_55to59    0.3017     0.108  2.789     0.006     0.087    0.516
 age_group_5_60to64   -0.0105     0.180  0.058     0.954   -0.347    0.327
 age_group_5_65_over   0.0390     0.142  0.255     0.799     0.039    0.142
--------------------------------------------------------------------------------

Omnibus:  3.523  Durbin-Watson:  2.050  
Prob(Omnibus):  0.172  Jarque-Bera (JB):  2.913  
Skew:  0.285  Prob(JB):  0.233  
Kurtosis:  2.448  Cond. No.  2.68e+16  

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 2.66e-31. This might indicate that there are
stron multicollinearity problems or that the design matrix is singular.

```python
model40 = sm.ols(data=merit_raises_combined_hourly_regression, formula='performance_rating ~ gender_Female + gender_Male + race_grouping_white +
race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 +
age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 +
age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 +
age_group_5_60to64 + age_group_5_65_over')
result40 = model40.fit()
result40.summary()
```

OLS Regression Results

```
+---------------------------------------------+----------+----------+----------+----------+
| Dep. Variable: performance_rating          | R-squared: | 0.164    |
| Model:                                      | OLS       | 0.071    |
| Method:                                    | Least Squares | 1.764    |
| Date:                                      | Wed, 06 Nov 2019 | 0.0705   |
| Time:                                      | 10:27:51  | -30.408  |
| No. Observations:                         | 111       | 84.82    |
| Df Residuals:                              | 99        | 117.3    |
| Df Model:                                  | 11        |          |
| Covariance Type:                           | nonrobust |          |
+---------------------------------------------+----------+----------+----------+----------+

+-------------------------------+----------+----------+----------+----------+
|                               | coef     | std err  | t        | P>|t|   |
|-------------------------------|----------|----------|----------|--------|
| [0.025 0.975]                 |----------|----------|----------|--------|
|-------------------------------+----------+----------+----------+----------|
| Intercept                     | 1.6522   | 0.022    | 76.828   | 0.000  |
|                               | 1.610    | 1.695    |          |        |
| gender_Female                 | 0.8254   | 0.039    | 21.212   | 0.000  |
|                               | 0.748    | 0.903    |          |        |
| gender_Male                   | 0.8268   | 0.040    | 20.585   | 0.000  |
|                               | 0.747    | 0.907    |          |        |
| race_grouping_white           | 0.8965   | 0.037    | 24.406   | 0.000  |
|                               | 0.824    | 0.969    |          |        |
| race_grouping_person_of_color | 0.7557   | 0.043    | 17.447   | 0.000  |
|                               | 0.670    | 0.842    |          |        |
| age_group_5_25_under           | -0.0748  | 0.220    | -0.340   | 0.734  |
|                               | -0.511   | 0.361    |          |        |
| age_group_5_25to29            | 0.1668   | 0.078    | 2.144    | 0.034  |
|                               | 0.012    | 0.321    |          |        |
| age_group_5_30to34            | 0.1356   | 0.089    | 1.521    | 0.131  |
|                               | 0.041    | 0.312    |          |        |
| age_group_5_35to39            | 0.0292   | 0.102    | 0.287    | 0.775  |
+-------------------------------+----------+----------+----------+----------+
```
1.6 Commercial

1.6.1 Gender

[311]: current_commercial_gender_salaried = commercial_salaried.groupby(['gender']).
       .agg({'current_base_pay': [np.count_nonzero]})
       suppress(current_commercial_gender_salaried)

[311]:

       count_nonzero
       gender
       Female  86.00
       Male  47.00

[312]: current_commercial_gender_hourly = commercial_hourly.groupby(['gender']).
       .agg({'current_base_pay': [np.count_nonzero]})
       suppress(current_commercial_gender_hourly)

[312]:

       count_nonzero
       gender
       Female  74.00
       Male  73.00
current_commercial_gender_salaried_median = commercial_salaried.
groupby(['gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_commercial_gender_salaried_median)

current_commercial_gender_salaried_median

<table>
<thead>
<tr>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>86.00</td>
<td>85977.35</td>
</tr>
<tr>
<td>Male</td>
<td>47.00</td>
<td>86880.00</td>
</tr>
</tbody>
</table>

current_commercial_gender_hourly_median = commercial_hourly.
groupby(['gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_commercial_gender_hourly_median)

current_commercial_gender_hourly_median

<table>
<thead>
<tr>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>74.00</td>
<td>28.89</td>
</tr>
<tr>
<td>Male</td>
<td>73.00</td>
<td>23.45</td>
</tr>
</tbody>
</table>

current_commercial_gender_age_salaried = commercial_salaried.
groupby(['gender'])['age'].median().sort_values(ascending=False)
current_commercial_gender_age_salaried

gender

<table>
<thead>
<tr>
<th>gender</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Male</td>
<td>39.00</td>
</tr>
<tr>
<td>Female</td>
<td>32.00</td>
</tr>
</tbody>
</table>
Name: age, dtype: float64

current_commercial_gender_age_hourly = commercial_hourly.
groupby(['gender'])['age'].median().sort_values(ascending=False)
current_commercial_gender_age_hourly

gender

<table>
<thead>
<tr>
<th>gender</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Male</td>
<td>47.00</td>
</tr>
<tr>
<td>Female</td>
<td>43.50</td>
</tr>
</tbody>
</table>
Name: age, dtype: float64

current_commercial_gender_age_5_salary = commercial_salaried.
groupby(['age_group_5', 'gender']).agg({'current_base_pay': [np.
count_nonzero, np.median]})
suppress(current_commercial_gender_age_5_salary)

current_commercial_gender_age_5_salary

<table>
<thead>
<tr>
<th>age_group_5</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>Female</td>
<td>8.00 63500.00</td>
</tr>
<tr>
<td>25–29</td>
<td>Female</td>
<td>29.00 75000.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>6.00 79140.00</td>
</tr>
<tr>
<td>30–34</td>
<td>Female</td>
<td>9.00 100000.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>7.00 97695.60</td>
</tr>
<tr>
<td>35–39</td>
<td>Female</td>
<td>9.00 149101.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>9.00 77626.78</td>
</tr>
<tr>
<td>40–44</td>
<td>Female</td>
<td>8.00 124287.97</td>
</tr>
</tbody>
</table>
45-49  Female  7.00  90585.00  
        Male   6.00  85089.96  
50-54  Female  7.00  90669.48  
55-59  Female  5.00  96780.00  
        Male   5.00  97134.77  
60-64  Male    6.00  95753.93  

[318]: current_commercial_gender_age_5_hourly = commercial_hourly.
        agroupby(['age_group_5', 'gender']).agg({'current_base_pay': [np.
        count_nonzero, np.median]})
suppress(current_commercial_gender_age_5_hourly)

[318]:

<table>
<thead>
<tr>
<th>age_group_5</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>Male</td>
<td>7.00</td>
<td>23.08</td>
</tr>
<tr>
<td>25-29</td>
<td>Female</td>
<td>14.00</td>
<td>31.76</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>8.00</td>
<td>26.17</td>
</tr>
<tr>
<td>30-34</td>
<td>Female</td>
<td>6.00</td>
<td>30.32</td>
</tr>
<tr>
<td>35-39</td>
<td>Female</td>
<td>5.00</td>
<td>30.77</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>8.00</td>
<td>30.62</td>
</tr>
<tr>
<td>40-44</td>
<td>Female</td>
<td>12.00</td>
<td>29.48</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>5.00</td>
<td>21.50</td>
</tr>
<tr>
<td>45-49</td>
<td>Female</td>
<td>7.00</td>
<td>31.28</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>10.00</td>
<td>22.39</td>
</tr>
<tr>
<td>50-54</td>
<td>Female</td>
<td>6.00</td>
<td>23.27</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>12.00</td>
<td>24.15</td>
</tr>
<tr>
<td>55-59</td>
<td>Female</td>
<td>9.00</td>
<td>26.41</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>7.00</td>
<td>23.45</td>
</tr>
<tr>
<td>60-64</td>
<td>Female</td>
<td>6.00</td>
<td>24.51</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>7.00</td>
<td>24.27</td>
</tr>
<tr>
<td>65+</td>
<td>Female</td>
<td>5.00</td>
<td>27.69</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>6.00</td>
<td>22.73</td>
</tr>
</tbody>
</table>

[319]: current_commercial_gender_age_10_salary = commercial_salaried.
        agroupby(['age_group_10', 'gender']).agg({'current_base_pay': [np.
        count_nonzero, np.median]})
suppress(current_commercial_gender_age_10_salary)

[319]:

<table>
<thead>
<tr>
<th>age_group_10</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>Female</td>
<td>8.00</td>
<td>63500.00</td>
</tr>
<tr>
<td>25-34</td>
<td>Female</td>
<td>38.00</td>
<td>80212.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>13.00</td>
<td>86880.00</td>
</tr>
<tr>
<td>35-44</td>
<td>Female</td>
<td>17.00</td>
<td>143575.94</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>10.00</td>
<td>84029.11</td>
</tr>
<tr>
<td>45-54</td>
<td>Female</td>
<td>14.00</td>
<td>90627.24</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>9.00</td>
<td>85000.00</td>
</tr>
<tr>
<td>55-64</td>
<td>Female</td>
<td>9.00</td>
<td>96780.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>11.00</td>
<td>97134.77</td>
</tr>
</tbody>
</table>
current_commercial_gender_age_10_hourly = commercial_hourly.
  groupby(['age_group_10', 'gender']).agg({'current_base_pay': [np.
  count_nonzero, np.median]})
suppress(current_commercial_gender_age_10_hourly)

count_nonzero  median
age_group_10  gender
<25           Male   7.00  23.08
              Female  20.00 31.03
              Male   11.00 26.04
25-34         Female 17.00 29.74
              Male   13.00 27.18
35-44         Female 13.00 26.14
              Male   22.00 23.49
45-54         Female 15.00 25.36
              Male   14.00 23.86
55-64         Female  5.00 27.69
              Male   6.00 22.73
65+           Female  5.00 27.69
              Male   6.00 22.73

current_commercial_gender_salaried_under_40 =
  commercial_salaried[commercial_salaried['age'] < 40].groupby(['gender']).
  agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_commercial_gender_salaried_under_40)

count_nonzero  median
gender
Female    55.00  80424.00
Male      24.00  83140.00

current_commercial_gender_salaried_over_40 =
  commercial_salaried[commercial_salaried['age'] > 39].groupby(['gender']).
  agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_commercial_gender_salaried_over_40)

count_nonzero  median
gender
Female    31.00  96780.00
Male      23.00  90000.00

current_commercial_gender_hourly_under_40 =
  commercial_hourly[commercial_hourly['age'] < 40].groupby(['gender']).
  agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_commercial_gender_hourly_under_40)

count_nonzero  median
gender
Female    29.00  30.38
Male      26.00  26.53
current_commercial_gender_hourly_over_40 = 
commercial_hourly[commercial_hourly['age'] > 39].groupby(['gender']).
agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_commercial_gender_hourly_over_40)

count_nonzero  median
gender
Female  45.00  27.69
Male  47.00  23.20

1.6.2 Race and ethnicity

current_commercial_race_salaried = commercial_salaried.
        groupby(['race_ethnicity']).agg({'current_base_pay': [np.count_nonzero]})
suppress_count(current_commercial_race_salaried)

count_nonzero  
race_ethnicity
White (United States of America)  99.00
Black or African American (United States of Ame...  14.00
Asian (United States of America)  13.00
Hispanic or Latino (United States of America)  5.00

current_commercial_race_hourly = commercial_hourly.groupby(['race_ethnicity']).
        agg({'current_base_pay': [np.count_nonzero]})
suppress_count(current_commercial_race_hourly)

count_nonzero  
race_ethnicity
Black or African American (United States of Ame...  82.00
White (United States of America)  43.00
Hispanic or Latino (United States of America)  9.00
Asian (United States of America)  7.00

current_commercial_race_group_salaried = commercial_salaried.
        groupby(['race_grouping']).agg({'current_base_pay': [np.count_nonzero]})
suppress_count(current_commercial_race_group_salaried)

count_nonzero  
race_grouping
white  99.00
person of color  32.00

current_commercial_race_group_hourly = commercial_hourly.
        groupby(['race_grouping']).agg({'current_base_pay': [np.count_nonzero]})
suppress_count(current_commercial_race_group_hourly)

count_nonzero  
race_grouping
person of color  101.00
white 43.00

```python
[329]: current_commercial_race_median_salaried = commercial_salaried.
    → groupby(['race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.
    ← median]})
    suppress_median(current_commercial_race_median_salaried)
```

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>White (United States of America)</td>
<td>99.00</td>
<td>88000.00</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>14.00</td>
<td>84640.00</td>
</tr>
<tr>
<td>Asian (United States of America)</td>
<td>13.00</td>
<td>80000.00</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>5.00</td>
<td>80000.00</td>
</tr>
</tbody>
</table>

```python
[330]: current_commercial_race_median_hourly = commercial_hourly.
    → groupby(['race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.
    ← median]})
    suppress_median(current_commercial_race_median_hourly)
```

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>White (United States of America)</td>
<td>43.00</td>
<td>30.38</td>
</tr>
<tr>
<td>Asian (United States of America)</td>
<td>7.00</td>
<td>26.04</td>
</tr>
<tr>
<td>Black or African American (United States of America)</td>
<td>82.00</td>
<td>24.91</td>
</tr>
<tr>
<td>Hispanic or Latino (United States of America)</td>
<td>9.00</td>
<td>23.12</td>
</tr>
</tbody>
</table>

```python
[331]: current_commercial_race_group_median_salaried = commercial_salaried.
    → groupby(['race_grouping']).agg({'current_base_pay': [np.count_nonzero, np.
    ← median]})
    suppress_median(current_commercial_race_group_median_salaried)
```

<table>
<thead>
<tr>
<th>race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>white</td>
<td>99.00</td>
<td>88000.00</td>
</tr>
<tr>
<td>person of color</td>
<td>32.00</td>
<td>83444.64</td>
</tr>
</tbody>
</table>

```python
[332]: current_commercial_race_group_median_hourly = commercial_hourly.
    → groupby(['race_grouping']).agg({'current_base_pay': [np.count_nonzero, np.
    ← median]})
    suppress_median(current_commercial_race_group_median_hourly)
```

<table>
<thead>
<tr>
<th>race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>white</td>
<td>43.00</td>
<td>30.38</td>
</tr>
<tr>
<td>person of color</td>
<td>101.00</td>
<td>25.16</td>
</tr>
</tbody>
</table>

```python
[333]: current_commercial_race_age_salaried = commercial_salaried.
    → groupby(['race_ethnicity'])['age'].median().sort_values(ascending=False)
current_commercial_race_age_salaried
```
```python
[333]: race_ethnicity
    Black or African American (United States of America) 48.00
    Hispanic or Latino (United States of America) 41.00
    Prefer Not to Disclose (United States of America) 35.50
    White (United States of America) 35.00
    Asian (United States of America) 32.00
Name: age, dtype: float64

[334]: current_commercial_race_age_hourly = commercial_hourly.
        groupby(['race_ethnicity'])['age'].median().sort_values(ascending=False)
current_commercial_race_age_hourly

[334]: race_ethnicity
    Black or African American (United States of America) 48.50
    White (United States of America) 39.00
    American Indian or Alaska Native (United States of America) 38.00
    Prefer Not to Disclose (United States of America) 35.00
    Two or More Races (United States of America) 31.00
    Hispanic or Latino (United States of America) 30.00
    Asian (United States of America) 28.00
Name: age, dtype: float64

[335]: current_commercial_race_age_5_salary = commercial_salaried.
        groupby(['age_group_5','race_ethnicity']).agg({'current_base_pay': [np.
        count_nonzero, np.median]})
suppress(current_commercial_race_age_5_salary)

[335]:
<table>
<thead>
<tr>
<th>age_group_5</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>White (United States of America)</td>
<td>9.00</td>
</tr>
<tr>
<td>25-29</td>
<td>White (United States of America)</td>
<td>28.00</td>
</tr>
<tr>
<td>30-34</td>
<td>White (United States of America)</td>
<td>12.00</td>
</tr>
<tr>
<td>35-39</td>
<td>White (United States of America)</td>
<td>13.00</td>
</tr>
<tr>
<td>40-44</td>
<td>White (United States of America)</td>
<td>6.00</td>
</tr>
<tr>
<td>45-49</td>
<td>White (United States of America)</td>
<td>7.00</td>
</tr>
<tr>
<td>50-54</td>
<td>White (United States of America)</td>
<td>9.00</td>
</tr>
<tr>
<td>55-59</td>
<td>White (United States of America)</td>
<td>8.00</td>
</tr>
<tr>
<td>60-64</td>
<td>White (United States of America)</td>
<td>6.00</td>
</tr>
</tbody>
</table>

[336]: current_commercial_race_age_5_hourly = commercial_hourly.
        groupby(['age_group_5','race_ethnicity']).agg({'current_base_pay': [np.
        count_nonzero, np.median]})
suppress(current_commercial_race_age_5_hourly)

[336]:
<table>
<thead>
<tr>
<th>age_group_5</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>Black or African American (United States of Ame...</td>
<td>5.00</td>
</tr>
<tr>
<td>25-29</td>
<td>White (United States of America)</td>
<td>11.00</td>
</tr>
<tr>
<td>35-39</td>
<td>White (United States of America)</td>
<td>6.00</td>
</tr>
<tr>
<td>40-44</td>
<td>Black or African American (United States of Ame...</td>
<td>13.00</td>
</tr>
</tbody>
</table>
```
45-49  Black or African American (United States of America)  14.00
50-54  Black or African American (United States of America)  12.00
       White (United States of America)  5.00
55-59  Black or African American (United States of America)  11.00
       White (United States of America)  5.00
60-64  Black or African American (United States of America)  11.00
       White (United States of America)  5.00
65+    Black or African American (United States of America)  5.00

age_group_5  race_ethnicity
<25  Black or African American (United States of America)  22.36
     White (United States of America)  31.84
25-34 White (United States of America)  30.81
     Black or African American (United States of America)  28.89
45-49 Black or African American (United States of America)  23.11
50-54 Black or African American (United States of America)  23.27
     White (United States of America)  24.44
55-59 Black or African American (United States of America)  27.05
     White (United States of America)  25.36
60-64 Black or African American (United States of America)  24.27
65+    Black or African American (United States of America)  23.39

[337]:
current_commercial_race_age_10_salary = commercial_salaried.
groupby(['age_group_10', 'race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_commercial_race_age_10_salary)

[337]:
age_group_10  race_ethnicity  count_nonzero  median
<25  White (United States of America)  9 63000.00
     Black or African American (United States of America)  5
     Hispanic or Latino (United States of America)  7
     White (United States of America)  2
25-34 White (United States of America)  6 82418.32
     Black or African American (United States of America)  7
     Hispanic or Latino (United States of America)  6
     White (United States of America)  11
35-44 White (United States of America)  40 82000.00
     Black or African American (United States of America)  19
     Hispanic or Latino (United States of America)  14
     White (United States of America)  16
45-54 White (United States of America)  16 88695.95
     Black or African American (United States of America)  17
     Hispanic or Latino (United States of America)  12
     White (United States of America)  17
55-64 White (United States of America)  14 97324.60
     Black or African American (United States of America)  5
     Hispanic or Latino (United States of America)  6
     White (United States of America)  12

[338]:
current_commercial_race_age_10_hourly = commercial_hourly.
groupby(['age_group_10', 'race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_commercial_race_age_10_hourly)

[338]:
age_group_10  race_ethnicity  count_nonzero
<25  Black or African American (United States of America)  5
     Hispanic or Latino (United States of America)  7
     White (United States of America)  12
25-34 Black or African American (United States of America)  7
     Hispanic or Latino (United States of America)  6
     White (United States of America)  12
35-44 Black or African American (United States of America)  17
<table>
<thead>
<tr>
<th>Age Group</th>
<th>Race/Ethnicity</th>
<th>Median Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>45-54</td>
<td>Black or African American</td>
<td>26.00</td>
</tr>
<tr>
<td>55-64</td>
<td>Black or African American</td>
<td>22.00</td>
</tr>
<tr>
<td>65+</td>
<td>Black or African American</td>
<td>5.00</td>
</tr>
</tbody>
</table>

```python
[339]: current_commercial_race_group_age_5_salary = commercial_salaried.
  ➔ groupby(['age_group_5', 'race_grouping']).agg({'current_base_pay': [np.
  ➔ count_nonzero, np.median]})
suppress(current_commercial_race_group_age_5_salary)
```

```python
[339]:
count_nonzero  median
<table>
<thead>
<tr>
<th>Age Group</th>
<th>Race/Ethnicity</th>
<th>Count</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>White</td>
<td>9.00</td>
<td>63000.00</td>
</tr>
<tr>
<td>25-29</td>
<td>Person of Color</td>
<td>7.00</td>
<td>72000.00</td>
</tr>
<tr>
<td>30-34</td>
<td>White</td>
<td>12.00</td>
<td>98847.80</td>
</tr>
<tr>
<td>35-39</td>
<td>Person of Color</td>
<td>5.00</td>
<td>73521.60</td>
</tr>
<tr>
<td>40-44</td>
<td>White</td>
<td>6.00</td>
<td>126864.75</td>
</tr>
<tr>
<td>45-49</td>
<td>Person of Color</td>
<td>6.00</td>
<td>85449.96</td>
</tr>
<tr>
<td>50-54</td>
<td>White</td>
<td>9.00</td>
<td>87391.89</td>
</tr>
<tr>
<td>55-59</td>
<td>White</td>
<td>8.00</td>
<td>96957.39</td>
</tr>
<tr>
<td>60-64</td>
<td>White</td>
<td>6.00</td>
<td>97651.02</td>
</tr>
</tbody>
</table>

```python
[340]: current_commercial_race_group_age_5_hourly = commercial_hourly.
  ➔ groupby(['age_group_5', 'race_grouping']).agg({'current_base_pay': [np.
  ➔ count_nonzero, np.median]})
suppress(current_commercial_race_group_age_5_hourly)
```

```python
[340]:
count_nonzero  median
<table>
<thead>
<tr>
<th>Age Group</th>
<th>Race/Ethnicity</th>
<th>Count</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>Person of Color</td>
<td>7.00</td>
<td>25.64</td>
</tr>
<tr>
<td>Age Group</td>
<td>Race Grouping</td>
<td>Base Pay</td>
<td>Median</td>
</tr>
<tr>
<td>-----------</td>
<td>---------------</td>
<td>----------</td>
<td>--------</td>
</tr>
<tr>
<td>25-29</td>
<td>person of color</td>
<td>10.00</td>
<td>26.29</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>11.00</td>
<td>31.84</td>
</tr>
<tr>
<td>30-34</td>
<td>person of color</td>
<td>8.00</td>
<td>28.82</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>6.00</td>
<td>30.81</td>
</tr>
<tr>
<td>35-39</td>
<td>person of color</td>
<td>6.00</td>
<td>30.81</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>6.00</td>
<td>30.81</td>
</tr>
<tr>
<td>40-44</td>
<td>person of color</td>
<td>14.00</td>
<td>28.52</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>11.00</td>
<td>31.84</td>
</tr>
<tr>
<td>45-49</td>
<td>person of color</td>
<td>14.00</td>
<td>23.11</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>6.00</td>
<td>30.81</td>
</tr>
<tr>
<td>50-54</td>
<td>person of color</td>
<td>13.00</td>
<td>23.19</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>5.00</td>
<td>24.44</td>
</tr>
<tr>
<td>55-59</td>
<td>person of color</td>
<td>11.00</td>
<td>27.05</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>5.00</td>
<td>25.36</td>
</tr>
<tr>
<td>60-64</td>
<td>person of color</td>
<td>11.00</td>
<td>24.27</td>
</tr>
<tr>
<td>65+</td>
<td>person of color</td>
<td>7.00</td>
<td>23.40</td>
</tr>
</tbody>
</table>

```
[341]: current_commercial_race_group_age_10_salary = commercial_salaried.
    -> groupby(['age_group_10', 'race_grouping']).agg({'current_base_pay': [np.
    -> count_nonzero, np.median]})
    suppress(current_commercial_race_group_age_10_salary)

[341]:

<table>
<thead>
<tr>
<th>age_group_10</th>
<th>race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>white</td>
<td>9.00</td>
<td>63000.00</td>
</tr>
<tr>
<td>25-34</td>
<td>person of color</td>
<td>10.00</td>
<td>74918.32</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>40.00</td>
<td>82000.00</td>
</tr>
<tr>
<td>35-44</td>
<td>person of color</td>
<td>7.00</td>
<td>90431.45</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>19.00</td>
<td>148729.50</td>
</tr>
<tr>
<td>45-54</td>
<td>person of color</td>
<td>7.00</td>
<td>85000.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>16.00</td>
<td>88695.95</td>
</tr>
<tr>
<td>55-64</td>
<td>person of color</td>
<td>6.00</td>
<td>82708.86</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>14.00</td>
<td>97324.60</td>
</tr>
</tbody>
</table>

[342]: current_commercial_race_group_age_10_hourly = commercial_hourly.
    -> groupby(['age_group_10', 'race_grouping']).agg({'current_base_pay': [np.
    -> count_nonzero, np.median]})
    suppress(current_commercial_race_group_age_10_hourly)

[342]:

<table>
<thead>
<tr>
<th>age_group_10</th>
<th>race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>person of color</td>
<td>7.00</td>
<td>25.64</td>
</tr>
<tr>
<td>25-34</td>
<td>person of color</td>
<td>18.00</td>
<td>26.52</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>12.00</td>
<td>31.76</td>
</tr>
<tr>
<td>35-44</td>
<td>person of color</td>
<td>20.00</td>
<td>29.06</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>8.00</td>
<td>30.57</td>
</tr>
<tr>
<td>45-54</td>
<td>person of color</td>
<td>27.00</td>
<td>23.19</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>8.00</td>
<td>30.81</td>
</tr>
<tr>
<td>55-64</td>
<td>person of color</td>
<td>22.00</td>
<td>24.54</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>7.00</td>
<td>26.41</td>
</tr>
<tr>
<td>65+</td>
<td>person of color</td>
<td>7.00</td>
<td>23.40</td>
</tr>
</tbody>
</table>
```
current_commercial_race_under_40_salaried = commercial_salaried[commercial_salaried['age'] < 40].
groupby(['race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_commercial_race_under_40_salaried)

race_ethnicity
White (United States of America) 62.00 82000.00
Asian (United States of America) 10.00 77418.32

current_commercial_race_over_40_salaried = commercial_salaried[commercial_salaried['age'] > 39].
groupby(['race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_commercial_race_over_40_salaried)

race_ethnicity
White (United States of America) 37.00 97134.77
Black or African American (United States of America) 10.00 84848.86

current_commercial_race_under_40_hourly = commercial_hourly[commercial_hourly['age'] < 40].groupby(['race_ethnicity']).
agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_commercial_race_under_40_hourly)

current_commercial_race_over_40_hourly = commercial_hourly[commercial_hourly['age'] > 39].groupby(['race_ethnicity']).
agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_commercial_race_over_40_hourly)

race_ethnicity
White (United States of America) 22.00 31.46
Black or African American (United States of America) 16.00 26.50
Hispanic or Latino (United States of America) 8.00 25.62

1.6.3 Gender × race/ethnicity

current_commercial_race_gender_salaried = commercial_salaried.
groupby(['race_ethnicity', 'gender']).agg({'current_base_pay': [np.count_nonzero]})
suppress(current_commercial_race_gender_salaried)
```python
race_ethnicity gender
Asian (United States of America) Female 8.00
Male 5.00
Black or African American (United States of America) Female 7.00
Male 7.00
White (United States of America) Female 67.00
Male 32.00
```

```python
race_gender_hourly = commercial_hourly.groupby(['race_ethnicity', 'gender']).agg({'current_base_pay': [np.count_nonzero]})
suppress(race_gender_hourly)
```

```python
race_gender_median_salaried = commercial_salaried.groupby(['race_grouping', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(race_gender_median_salaried)
```

```python
race_gender_median_hourly = commercial_hourly.groupby(['race_grouping', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(race_gender_median_hourly)
```

```python
race_gender_under_40_salaried = commercial_salaried[commercial_salaried['age'] < 40].groupby(['race_ethnicity', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
```
### suppress(current_commercial_race_gender_under_40_salaried)

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Asian (United States of America)</td>
<td>Female</td>
<td>6.00</td>
<td>85000.00</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>Female</td>
<td>46.00</td>
<td>80212.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>16.00</td>
<td>90940.00</td>
</tr>
</tbody>
</table>

### current_commercial_race_gender_under_40_hourly

```python
commercial_hourly[commercial_hourly['age'] < 40].groupby(['race_ethnicity', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_commercial_race_gender_under_40_hourly)
```

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Black or African American</td>
<td>Female</td>
<td>8.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>8.00</td>
<td></td>
</tr>
<tr>
<td>Hispanic or Latino</td>
<td>Female</td>
<td>6.00</td>
<td></td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>Female</td>
<td>12.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>10.00</td>
<td></td>
</tr>
</tbody>
</table>

### median

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>gender</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Black or African American</td>
<td>Female</td>
<td>26.50</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>26.31</td>
</tr>
<tr>
<td>Hispanic or Latino</td>
<td>Female</td>
<td>28.51</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>Female</td>
<td>33.28</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>30.57</td>
</tr>
</tbody>
</table>

### current_commercial_race_gender_over_40_salaried

```python
commercial_salaried[commercial_salaried['age'] > 39].groupby(['race_ethnicity', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_commercial_race_gender_over_40_salaried)
```

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Black or African American</td>
<td>Female</td>
<td>6.00</td>
<td></td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>Female</td>
<td>21.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>16.00</td>
<td></td>
</tr>
</tbody>
</table>

### median

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>gender</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Black or African American</td>
<td>Female</td>
<td>94950.50</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>Female</td>
<td>97546.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>95564.10</td>
</tr>
</tbody>
</table>
current_commercial_race_gender_over_40_hourly =
commercial_hourly[commercial_hourly["age"] > 39].
groupby(["race_ethnicity", "gender"]).aggs({'current_base_pay': [np.
count_nonzero, np.median]})
suppress(current_commercial_race_gender_over_40_hourly)

<table>
<thead>
<tr>
<th>race_ethnicity</th>
<th>gender</th>
<th>count_nonzero</th>
</tr>
</thead>
<tbody>
<tr>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>33.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>33.00</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>Female</td>
<td>10.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>11.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>median</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>26.14</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>23.07</td>
</tr>
<tr>
<td>White (United States of America)</td>
<td>Female</td>
<td>31.02</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>23.85</td>
</tr>
</tbody>
</table>

### Years of service

<table>
<thead>
<tr>
<th>years_of_service_grouped</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>31.00</td>
<td>82000.00</td>
</tr>
<tr>
<td>1-2</td>
<td>36.00</td>
<td>80212.00</td>
</tr>
<tr>
<td>3-5</td>
<td>26.00</td>
<td>95769.71</td>
</tr>
<tr>
<td>6-10</td>
<td>15.00</td>
<td>99316.00</td>
</tr>
<tr>
<td>11-15</td>
<td>6.00</td>
<td>76331.03</td>
</tr>
<tr>
<td>16-20</td>
<td>6.00</td>
<td>81765.65</td>
</tr>
<tr>
<td>21-25</td>
<td>8.00</td>
<td>94006.52</td>
</tr>
<tr>
<td>25+</td>
<td>5.00</td>
<td>93490.62</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>years_of_service_grouped</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>26.00</td>
<td>25.64</td>
</tr>
<tr>
<td>1-2</td>
<td>33.00</td>
<td>26.99</td>
</tr>
<tr>
<td>3-5</td>
<td>14.00</td>
<td>23.16</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>6-10</td>
<td>19.00</td>
<td>23.98</td>
</tr>
<tr>
<td>11-15</td>
<td>14.00</td>
<td>30.15</td>
</tr>
<tr>
<td>16-20</td>
<td>17.00</td>
<td>24.32</td>
</tr>
<tr>
<td>21-25</td>
<td>9.00</td>
<td>29.74</td>
</tr>
<tr>
<td>25+</td>
<td>15.00</td>
<td>26.34</td>
</tr>
</tbody>
</table>

[357]:
```python
current_commercial_yos_gender_salary = commercial_salaried.
groupby(['years_of_service_grouped', 'gender']).agg({'current_base_pay': [np.
count_nonzero, np.median]})
suppress(current_commercial_yos_gender_salary)
```

[357]:
```
<table>
<thead>
<tr>
<th>years_of_service_grouped</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Female</td>
<td>22.00</td>
<td>74640.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>9.00</td>
<td>90000.00</td>
</tr>
<tr>
<td>1-2</td>
<td>Female</td>
<td>26.00</td>
<td>80212.00</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>10.00</td>
<td>81640.00</td>
</tr>
<tr>
<td>3-5</td>
<td>Female</td>
<td>16.00</td>
<td>94107.74</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>10.00</td>
<td>102496.71</td>
</tr>
<tr>
<td>6-10</td>
<td>Female</td>
<td>12.00</td>
<td>99499.70</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>6.00</td>
<td>91466.08</td>
</tr>
</tbody>
</table>

[358]:
```python
current_commercial_yos_gender_hourly = commercial_hourly.
groupby(['years_of_service_grouped', 'gender']).agg({'current_base_pay': [np.
count_nonzero, np.median]})
suppress(current_commercial_yos_gender_hourly)
```

[358]:
```
<table>
<thead>
<tr>
<th>years_of_service_grouped</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Female</td>
<td>10.00</td>
<td>29.48</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>16.00</td>
<td>22.05</td>
</tr>
<tr>
<td>1-2</td>
<td>Female</td>
<td>18.00</td>
<td>30.29</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>15.00</td>
<td>24.35</td>
</tr>
<tr>
<td>3-5</td>
<td>Female</td>
<td>5.00</td>
<td>30.77</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>9.00</td>
<td>22.14</td>
</tr>
<tr>
<td>6-10</td>
<td>Female</td>
<td>5.00</td>
<td>26.27</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>14.00</td>
<td>23.62</td>
</tr>
<tr>
<td>11-15</td>
<td>Male</td>
<td>10.00</td>
<td>29.04</td>
</tr>
<tr>
<td>16-20</td>
<td>Female</td>
<td>10.00</td>
<td>24.16</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>7.00</td>
<td>27.26</td>
</tr>
<tr>
<td>21-25</td>
<td>Female</td>
<td>8.00</td>
<td>27.94</td>
</tr>
<tr>
<td>25+</td>
<td>Female</td>
<td>14.00</td>
<td>26.58</td>
</tr>
</tbody>
</table>

[359]:
```python
current_commercial_yos_race_salary = commercial_salaried.
groupby(['years_of_service_grouped', 'race_ethnicity']).
agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_commercial_yos_race_salary)
```

[359]:
```python
<table>
<thead>
<tr>
<th>years_of_service_grouped</th>
<th>race_ethnicity</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td>10.00</td>
<td>29.48</td>
</tr>
<tr>
<td>1-2</td>
<td></td>
<td>18.00</td>
<td>30.29</td>
</tr>
<tr>
<td>3-5</td>
<td></td>
<td>5.00</td>
<td>30.77</td>
</tr>
<tr>
<td>6-10</td>
<td></td>
<td>5.00</td>
<td>26.27</td>
</tr>
<tr>
<td>11-15</td>
<td></td>
<td>10.00</td>
<td>29.04</td>
</tr>
<tr>
<td>16-20</td>
<td></td>
<td>10.00</td>
<td>24.16</td>
</tr>
<tr>
<td>21-25</td>
<td></td>
<td>8.00</td>
<td>27.94</td>
</tr>
</tbody>
</table>
| 25+                      |                | 14.00         | 26.58      |```
<table>
<thead>
<tr>
<th>years_of_service_grouped</th>
<th>race_ethnicity</th>
<th>median years_of_service_grouped race_ethnicity</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>White (United States of America)</td>
<td>82000.00</td>
</tr>
<tr>
<td>1-2</td>
<td>White (United States of America)</td>
<td>80212.00</td>
</tr>
<tr>
<td>3-5</td>
<td>White (United States of America)</td>
<td>108780.00</td>
</tr>
<tr>
<td>6-10</td>
<td>White (United States of America)</td>
<td>102500.00</td>
</tr>
<tr>
<td>16-20</td>
<td>White (United States of America)</td>
<td>87391.89</td>
</tr>
<tr>
<td>21-25</td>
<td>White (United States of America)</td>
<td>97651.02</td>
</tr>
</tbody>
</table>

```python
[360]: `current_commercial_yos_race_hourly = commercial_hourly.
        groupby(['years_of_service_grouped', 'race_ethnicity']).
        agg({'current_base_pay': [np.count_nonzero, np.median]})
        suppress(current_commercial_yos_race_hourly)
```

```python
[360]: count_nonzero \ years_of_service_grouped race_ethnicity
<table>
<thead>
<tr>
<th>years_of_service_grouped</th>
<th>race_ethnicity</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Black or African American (United States of Ame...</td>
</tr>
<tr>
<td>11.00</td>
<td>White (United States of America)</td>
</tr>
<tr>
<td>6.00</td>
<td>Black or African American (United States of Ame...</td>
</tr>
<tr>
<td>14.00</td>
<td>White (United States of America)</td>
</tr>
<tr>
<td>13.00</td>
<td>Black or African American (United States of Ame...</td>
</tr>
<tr>
<td>6.00</td>
<td>White (United States of America)</td>
</tr>
<tr>
<td>5.00</td>
<td>Black or African American (United States of Ame...</td>
</tr>
<tr>
<td>6-10</td>
<td>White (United States of America)</td>
</tr>
<tr>
<td>12.00</td>
<td>Black or African American (United States of Ame...</td>
</tr>
<tr>
<td>6.00</td>
<td>White (United States of America)</td>
</tr>
<tr>
<td>11-15</td>
<td>Black or African American (United States of Ame...</td>
</tr>
<tr>
<td>7.00</td>
<td>White (United States of America)</td>
</tr>
<tr>
<td>6.00</td>
<td>Black or African American (United States of Ame...</td>
</tr>
<tr>
<td>16-20</td>
<td>Black or African American (United States of Ame...</td>
</tr>
<tr>
<td>12.00</td>
<td>Black or African American (United States of Ame...</td>
</tr>
<tr>
<td>21-25</td>
<td>Black or African American (United States of Ame...</td>
</tr>
<tr>
<td>9.00</td>
<td>Black or African American (United States of Ame...</td>
</tr>
</tbody>
</table>
median

years_of_service_grouped race_ethnicity
0  Black or African American (United States of America)
  White (United States of America)
25.64
29.52  Black or African American (United States of America)
  White (United States of America)
34.72  White (United States of America)
3-5  Black or African American (United States of America)
  White (United States of America)
21.83  White (United States of America)
23.20  Black or African American (United States of America)
  White (United States of America)
23.62  White (United States of America)
29.91  White (United States of America)
11-15  Black or African American (United States of America)
  White (United States of America)
30.38  White (United States of America)
26.01  Black or African American (United States of America)
  White (United States of America)
24.13  Black or African American (United States of America)
  White (United States of America)
29.74  Black or African American (United States of America)
  White (United States of America)
24.71  White (United States of America)
24.71

[361]:
current_commercial_yos_race_gender_salary = commercial_salaried.
  groupby(["years_of_service_grouped","race_grouping","gender"]).
  agg({"current_base_pay": [np.count_nonzero, np.median]})
suppress(current_commercial_yos_race_gender_salary)

[361]:

years_of_service_grouped race_grouping gender  count_nonzero median
0  person of color Female  6.00 78500.00
   white Female  15.00 74280.00
   Male  8.00 92500.00
1-2  person of color Female  5.00 96980.00
   white Female  21.00 77383.00
   Male  9.00 83280.00
3-5  person of color Male  5.00 74836.65
   white Female  14.00 94107.74
   Male  5.00 125530.00
6-10  white Female  10.00 101091.70
```python
[362]: current_commercial_yos_race_gender_hourly = commercial_hourly.
    -> groupby(['years_of_service_grouped', 'race_grouping', 'gender']).
    -> agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_commercial_yos_race_gender_hourly)

[362]:

<table>
<thead>
<tr>
<th>years_of_service_grouped</th>
<th>race_grouping</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>person of color</td>
<td>Female</td>
<td>7.00</td>
<td>29.74</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>10.00</td>
<td>21.35</td>
</tr>
<tr>
<td>1-2</td>
<td>person of color</td>
<td>Female</td>
<td>9.00</td>
<td>26.73</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>11.00</td>
<td>22.36</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>9.00</td>
<td>35.01</td>
</tr>
<tr>
<td>3-5</td>
<td>person of color</td>
<td>Male</td>
<td>6.00</td>
<td>21.83</td>
</tr>
<tr>
<td>6-10</td>
<td>person of color</td>
<td>Male</td>
<td>10.00</td>
<td>23.42</td>
</tr>
<tr>
<td>11-15</td>
<td>person of color</td>
<td>Male</td>
<td>5.00</td>
<td>29.92</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Male</td>
<td>5.00</td>
<td>26.76</td>
</tr>
<tr>
<td>16-20</td>
<td>person of color</td>
<td>Female</td>
<td>9.00</td>
<td>23.99</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>5.00</td>
<td>24.27</td>
</tr>
<tr>
<td>21-25</td>
<td>person of color</td>
<td>Female</td>
<td>8.00</td>
<td>27.94</td>
</tr>
<tr>
<td>25+</td>
<td>person of color</td>
<td>Female</td>
<td>10.00</td>
<td>25.52</td>
</tr>
</tbody>
</table>

1.6.5 Age

[363]: current_median_commercial_age_5_salaried = commercial_salaried.
    -> groupby(['age_group_5']).agg({'current_base_pay': [np.count_nonzero, np.
    median]})
suppress(current_median_commercial_age_5_salaried)

[363]:

<table>
<thead>
<tr>
<th>age_group_5</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>10.00</td>
<td>64000.00</td>
</tr>
<tr>
<td>25-29</td>
<td>35.00</td>
<td>75000.00</td>
</tr>
<tr>
<td>30-34</td>
<td>16.00</td>
<td>98847.80</td>
</tr>
<tr>
<td>35-39</td>
<td>18.00</td>
<td>101091.70</td>
</tr>
<tr>
<td>40-44</td>
<td>9.00</td>
<td>143575.94</td>
</tr>
<tr>
<td>45-49</td>
<td>13.00</td>
<td>86104.69</td>
</tr>
<tr>
<td>50-54</td>
<td>10.00</td>
<td>87002.45</td>
</tr>
<tr>
<td>55-59</td>
<td>10.00</td>
<td>96957.39</td>
</tr>
<tr>
<td>60-64</td>
<td>10.00</td>
<td>95753.93</td>
</tr>
</tbody>
</table>

[364]: current_median_commercial_age_5_hourly = commercial_hourly.
    -> groupby(['age_group_5']).agg({'current_base_pay': [np.count_nonzero, np.
    median]})
suppress(current_median_commercial_age_5_hourly)

[364]:

<table>
<thead>
<tr>
<th>age_group_5</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>11.00</td>
<td>25.64</td>
</tr>
</tbody>
</table>
```

211
<table>
<thead>
<tr>
<th>Age Group</th>
<th>Median Base Pay</th>
<th>Median Salary</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>22.00</td>
<td>29.77</td>
</tr>
<tr>
<td>25-34</td>
<td>9.00</td>
<td>29.51</td>
</tr>
<tr>
<td>35-39</td>
<td>13.00</td>
<td>30.77</td>
</tr>
<tr>
<td>40-44</td>
<td>17.00</td>
<td>28.89</td>
</tr>
<tr>
<td>45-49</td>
<td>17.00</td>
<td>23.99</td>
</tr>
<tr>
<td>50-54</td>
<td>18.00</td>
<td>23.60</td>
</tr>
<tr>
<td>55-59</td>
<td>16.00</td>
<td>26.23</td>
</tr>
<tr>
<td>60-64</td>
<td>13.00</td>
<td>24.32</td>
</tr>
<tr>
<td>65+</td>
<td>11.00</td>
<td>23.40</td>
</tr>
</tbody>
</table>

```python
[365]: current_median_commercial_age_10_salaried = commercial_salaried.
    .groupby(['age_group_10']).agg({'current_base_pay': [np.count_nonzero, np.
    median]})
    suppress(current_median_commercial_age_10_salaried)
```

```python
[365]:
count_nonzero median
    age_group_10
    <25     10.00  64000.00
    25-34   51.00  82000.00
    35-44   27.00 105000.00
    45-54   23.00  86613.00
    55-64   20.00  96957.39
```

```python
[366]: current_median_commercial_age_10_hourly = commercial_hourly.
    .groupby(['age_group_10']).agg({'current_base_pay': [np.count_nonzero, np.
    median]})
    suppress(current_median_commercial_age_10_hourly)
```

```python
[366]:
count_nonzero median
    age_group_10
    <25     11.00  25.64
    25-34   31.00  29.51
    35-44   30.00  29.23
    45-54   35.00  23.85
    55-64   29.00  24.71
    65+     11.00  23.40
```

```python
[367]: current_commercial_age_5_yos_salary = commercial_salaried.
    .groupby(['age_group_5','years_of_service_grouped']).agg({'current_base_pay': [np.
    count_nonzero, np.median]})
    suppress(current_commercial_age_5_yos_salary)
```

```python
[367]:
count_nonzero median
    age_group_5 years_of_service_grouped
    <25     0       6.00  62500.00
    25-29   0       14.00 75000.00
    1-2     17.00  76000.00
    30-34   0       6.00 100000.00
    1-2     7.00  96980.00
    35-39   3-5     7.00 149101.00
```
6-10 6.00 101091.70
40-44 3-5 5.00 167000.00
60-64 21-25 5.00 97514.43

[368]: current_commercial_age_5_yos_hourly = commercial_hourly.
   groupby(['age_group_5','years_of_service_grouped']).agg({'current_base_pay':
   [np.count_nonzero, np.median]})
suppress(current_commercial_age_5_yos_hourly)

[368]:

<table>
<thead>
<tr>
<th>age_group_5</th>
<th>years_of_service_grouped</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td></td>
<td>0</td>
<td>5.00</td>
</tr>
<tr>
<td>25-29</td>
<td></td>
<td>0</td>
<td>6.00</td>
</tr>
<tr>
<td>30-34</td>
<td></td>
<td>1-2</td>
<td>6.00</td>
</tr>
<tr>
<td>35-39</td>
<td>11-15</td>
<td>0</td>
<td>5.00</td>
</tr>
<tr>
<td>40-44</td>
<td>3-5</td>
<td>1-2</td>
<td>5.00</td>
</tr>
<tr>
<td>45-54</td>
<td>12-15</td>
<td>5.00</td>
<td>6.00</td>
</tr>
<tr>
<td>55-64</td>
<td>21-25</td>
<td>6.00</td>
<td>5.00</td>
</tr>
<tr>
<td>65+</td>
<td>25+</td>
<td>5.00</td>
<td>5.00</td>
</tr>
</tbody>
</table>

[369]: current_commercial_age_10_yos_salary = commercial_salaried.
   groupby(['age_group_10','years_of_service_grouped']).agg({'current_base_pay':
   [np.count_nonzero, np.median]})
suppress(current_commercial_age_10_yos_salary)

[369]:

<table>
<thead>
<tr>
<th>age_group_10</th>
<th>years_of_service_grouped</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td></td>
<td>0</td>
<td>6.00</td>
</tr>
<tr>
<td>25-34</td>
<td></td>
<td>0</td>
<td>20.00</td>
</tr>
<tr>
<td>35-44</td>
<td>3-5</td>
<td>1-2</td>
<td>24.00</td>
</tr>
<tr>
<td>45-54</td>
<td>6-10</td>
<td>3-5</td>
<td>12.00</td>
</tr>
<tr>
<td>55-64</td>
<td>21-25</td>
<td>6.00</td>
<td>5.00</td>
</tr>
<tr>
<td>65+</td>
<td>25+</td>
<td>5.00</td>
<td>5.00</td>
</tr>
</tbody>
</table>

[370]: current_commercial_age_10_yos_hourly = commercial_hourly.
   groupby(['age_group_10','years_of_service_grouped']).agg({'current_base_pay':
   [np.count_nonzero, np.median]})
suppress(current_commercial_age_10_yos_hourly)

[370]:

<table>
<thead>
<tr>
<th>age_group_10</th>
<th>years_of_service_grouped</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td></td>
<td>0</td>
<td>5.00</td>
</tr>
<tr>
<td>1-2</td>
<td></td>
<td>6.00</td>
<td>23.08</td>
</tr>
<tr>
<td>25-34</td>
<td></td>
<td>0</td>
<td>11.00</td>
</tr>
<tr>
<td>1-2</td>
<td></td>
<td>15.00</td>
<td>26.23</td>
</tr>
<tr>
<td>35-44</td>
<td></td>
<td>0</td>
<td>5.00</td>
</tr>
</tbody>
</table>
current_median_commercial_age_5_gender_salaried = commercial_salaried.
    groupby(['age_group_5', 'gender']).agg({'current_base_pay': [np.
    count_nonzero, np.median]})
suppress(current_median_commercial_age_5_gender_salaried)

current_median_commercial_age_5_gender_hourly = commercial_hourly.
    groupby(['age_group_5', 'gender']).agg({'current_base_pay': [np.
    count_nonzero, np.median]})
suppress(current_median_commercial_age_5_gender_hourly)
<table>
<thead>
<tr>
<th>Age Group</th>
<th>Gender</th>
<th>Current Median Commercial Age 10 Gender Salaried</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Male</td>
<td>10.00 22.39</td>
</tr>
<tr>
<td>50-54</td>
<td>Female</td>
<td>6.00 23.27</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>12.00 24.15</td>
</tr>
<tr>
<td>55-59</td>
<td>Female</td>
<td>9.00 26.41</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>7.00 23.45</td>
</tr>
<tr>
<td>60-64</td>
<td>Female</td>
<td>6.00 24.51</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>7.00 24.27</td>
</tr>
<tr>
<td>65+</td>
<td>Female</td>
<td>5.00 27.69</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>6.00 22.73</td>
</tr>
</tbody>
</table>

```python
[373]: current_median_commercial_age_10_gender_salaried = commercial_salaried.
    groupby(['age_group_10', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
    suppress(current_median_commercial_age_10_gender_salaried)
```

```python
[373]:
    age_group_10 gender count_nonzero median
    <25 Female 8.00 63500.00
    25-34 Female 38.00 80212.00
    Male 13.00 86880.00
    35-44 Female 17.00 143575.94
    Male 10.00 84029.11
    45-54 Female 14.00 90627.24
    Male 9.00 85000.00
    55-64 Female 9.00 96780.00
    Male 11.00 97134.77

[374]: current_median_commercial_age_10_gender_hourly = commercial_hourly.
    groupby(['age_group_10', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
    suppress(current_median_commercial_age_10_gender_hourly)
```

```python
[374]:
    age_group_10 gender count_nonzero median
    <25 Male 7.00 23.08
    25-34 Female 20.00 31.03
    Male 11.00 26.04
    35-44 Female 17.00 29.74
    Male 13.00 27.18
    45-54 Female 13.00 26.14
    Male 22.00 23.49
    55-64 Female 15.00 25.36
    Male 14.00 23.86
    65+ Female 5.00 27.69
    Male 6.00 22.73
```

```python
[375]: current_median_commercial_age_5_race_salaried = commercial_salaried.
    groupby(['age_group_5', 'race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.median]})
```
```python
suppress(current_median_commercial_age_5_race_salaried)

[375]:

<table>
<thead>
<tr>
<th>age_group_5</th>
<th>race_ethnicity</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>White (United States of America)</td>
<td>9.00</td>
<td>63000.00</td>
</tr>
<tr>
<td>25-29</td>
<td>White (United States of America)</td>
<td>28.00</td>
<td>78691.50</td>
</tr>
<tr>
<td>30-34</td>
<td>White (United States of America)</td>
<td>12.00</td>
<td>98847.80</td>
</tr>
<tr>
<td>35-39</td>
<td>White (United States of America)</td>
<td>13.00</td>
<td>149101.00</td>
</tr>
<tr>
<td>40-44</td>
<td>White (United States of America)</td>
<td>6.00</td>
<td>126864.75</td>
</tr>
<tr>
<td>45-49</td>
<td>White (United States of America)</td>
<td>7.00</td>
<td>90000.00</td>
</tr>
<tr>
<td>50-54</td>
<td>White (United States of America)</td>
<td>9.00</td>
<td>87391.89</td>
</tr>
<tr>
<td>55-59</td>
<td>White (United States of America)</td>
<td>8.00</td>
<td>96957.39</td>
</tr>
<tr>
<td>60-64</td>
<td>White (United States of America)</td>
<td>6.00</td>
<td>97651.02</td>
</tr>
</tbody>
</table>

[376]:

current_median_commercial_age_5_race_hourly = commercial_hourly.

...groupby(['age_group_5', 'race_ethnicity']).agg({'current_base_pay': [np.
...count_nonzero, np.median]})

suppress(current_median_commercial_age_5_race_hourly)

[376]:

<table>
<thead>
<tr>
<th>age_group_5</th>
<th>race_ethnicity</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>Black or African American (United States of Ame...</td>
<td>5.00</td>
<td>22.36</td>
</tr>
<tr>
<td>25-29</td>
<td>White (United States of America)</td>
<td>11.00</td>
<td>31.84</td>
</tr>
<tr>
<td>35-39</td>
<td>White (United States of America)</td>
<td>6.00</td>
<td>30.81</td>
</tr>
<tr>
<td>40-44</td>
<td>Black or African American (United States of Ame...</td>
<td>13.00</td>
<td>28.89</td>
</tr>
<tr>
<td>45-49</td>
<td>Black or African American (United States of Ame...</td>
<td>14.00</td>
<td>23.11</td>
</tr>
<tr>
<td>50-54</td>
<td>Black or African American (United States of Ame...</td>
<td>12.00</td>
<td>23.27</td>
</tr>
<tr>
<td>55-59</td>
<td>Black or African American (United States of Ame...</td>
<td>5.00</td>
<td>24.44</td>
</tr>
<tr>
<td>60-64</td>
<td>Black or African American (United States of Ame...</td>
<td>11.00</td>
<td>27.05</td>
</tr>
<tr>
<td>65+</td>
<td>White (United States of America)</td>
<td>5.00</td>
<td>25.36</td>
</tr>
</tbody>
</table>

[377]:
```
```python
current_median_commercial_age_5_race_group_salaried = commercial_salaried.
groupby(['age_group_5', 'race_grouping']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_commercial_age_5_race_group_salaried)
```

```text
age_group_5 race_grouping
<25 white 9.00 63000.00
25-29 person of color 7.00 72000.00
   white 28.00 78691.50
30-34 white 12.00 98847.80
35-39 person of color 5.00 73521.60
   white 28.00 98847.80
40-44 white 6.00 126864.75
45-49 person of color 6.00 85449.96
   white 7.00 90000.00
50-54 white 9.00 87391.89
55-59 white 8.00 96957.39
60-64 white 6.00 97651.02
```

```python
current_median_commercial_age_5_race_group_hourly = commercial_hourly.
groupby(['age_group_5', 'race_grouping']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_commercial_age_5_race_group_hourly)
```

```text
age_group_5 race_grouping
<25 person of color 7.00 25.64
25-29 person of color 10.00 26.29
   white 11.00 31.84
30-34 person of color 8.00 28.82
35-39 person of color 6.00 30.81
   white 6.00 30.81
40-44 person of color 14.00 28.52
45-49 person of color 14.00 23.11
50-54 person of color 13.00 23.19
   white 5.00 24.44
55-59 person of color 11.00 27.05
   white 5.00 25.36
60-64 person of color 11.00 24.27
65+ person of color 7.00 23.40
```

```python
current_median_commercial_age_10_race_salaried = commercial_salaried.
groupby(['age_group_10', 'race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress(current_median_commercial_age_10_race_salaried)
```

```text
age_group_10 race_ethnicity
<25 person of color 7.00 25.64
25-29 person of color 10.00 26.29
   white 11.00 31.84
30-34 person of color 8.00 28.82
35-39 person of color 6.00 30.81
   white 6.00 30.81
40-44 person of color 14.00 28.52
45-49 person of color 14.00 23.11
50-54 person of color 13.00 23.19
   white 5.00 24.44
55-59 person of color 11.00 27.05
   white 5.00 25.36
60-64 person of color 11.00 24.27
65+ person of color 7.00 23.40
```
<table>
<thead>
<tr>
<th>Age Group</th>
<th>Race/Ethnicity</th>
<th>Median Current Base Pay</th>
<th>Median Total Earnings</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>White (United States of America)</td>
<td>9.00</td>
<td>63,000.00</td>
</tr>
<tr>
<td>25-34</td>
<td>Asian (United States of America)</td>
<td>6.00</td>
<td>82,418.32</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>40.00</td>
<td>82,000.00</td>
</tr>
<tr>
<td>35-44</td>
<td>White (United States of America)</td>
<td>19.00</td>
<td>148,729.50</td>
</tr>
<tr>
<td>45-54</td>
<td>White (United States of America)</td>
<td>16.00</td>
<td>88,695.95</td>
</tr>
<tr>
<td>55-64</td>
<td>White (United States of America)</td>
<td>14.00</td>
<td>97,324.60</td>
</tr>
</tbody>
</table>

```python
[380]:
current_median_commercial_age_10_race_hourly = commercial_hourly.
  -> groupby(['age_group_10', 'race_ethnicity']).agg({'current_base_pay': [np.
                  -> count_nonzero, np.median]})
  suppress(current_median_commercial_age_10_race_hourly)

[380]:
count_nonzero
  \ age_group_10 race_ethnicity
  <25 Black or African American (United States of Ame...  5.00
  25-34 Black or African American (United States of Ame...  7.00
  Hispanic or Latino (United States of America)          6.00
  White (United States of America)                       12.00
  35-44 Black or African American (United States of Ame...  17.00
  White (United States of America)                       8.00
  45-54 Black or African American (United States of Ame...  26.00
  White (United States of America)                       8.00
  55-64 Black or African American (United States of Ame...  22.00
  White (United States of America)                       7.00
  65+ Black or African American (United States of Ame...  5.00

median

age_group_10 race_ethnicity
<25 Black or African American (United States of Ame...  22.36
25-34 Black or African American (United States of Ame...  26.73
Hispanic or Latino (United States of America)        24.99
White (United States of America)                    31.76
35-44 Black or African American (United States of Ame...  29.23
White (United States of America)                    30.57
45-54 Black or African American (United States of Ame...  23.27
White (United States of America)                    30.81
55-64 Black or African American (United States of Ame...  24.54
White (United States of America)                    26.41
65+ Black or African American (United States of Ame...  23.39

[381]:
current_median_commercial_age_10_race_group_salaried = commercial_salaried.
  -> groupby(['age_group_10', 'race_grouping']).agg({'current_base_pay': [np.
                  -> count_nonzero, np.median]})
  suppress(current_median_commercial_age_10_race_group_salaried)

[381]:
count_nonzero median
  \ age_group_10 race_grouping
  <25 white 9.00 63000.00
```
25-34  person of color  10.00  74918.32
   white   40.00  82000.00
35-44  person of color  7.00  90431.45
   white   19.00  148729.50
45-54  person of color  7.00  85000.00
   white   16.00  88695.95
55-64  person of color  6.00  82708.86
   white   14.00  97324.60

[382]: current_median_commercial_age_10_race_group_hourly = commercial_hourly.
    -> groupby(['age_group_10', 'race_grouping']).agg({'current_base_pay': [np.
    -> count_nonzero, np.median]})
    suppress(current_median_commercial_age_10_race_group_hourly)

[382]:

<table>
<thead>
<tr>
<th>age_group_10</th>
<th>race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>person of color</td>
<td>7.00</td>
<td>25.64</td>
</tr>
<tr>
<td>25-34</td>
<td>person of color</td>
<td>18.00</td>
<td>26.52</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>12.00</td>
<td>31.76</td>
</tr>
<tr>
<td>35-44</td>
<td>person of color</td>
<td>20.00</td>
<td>29.06</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>8.00</td>
<td>30.57</td>
</tr>
<tr>
<td>45-54</td>
<td>person of color</td>
<td>27.00</td>
<td>23.19</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>8.00</td>
<td>30.81</td>
</tr>
<tr>
<td>55-64</td>
<td>person of color</td>
<td>22.00</td>
<td>24.54</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>7.00</td>
<td>26.41</td>
</tr>
<tr>
<td>65+</td>
<td>person of color</td>
<td>7.00</td>
<td>23.40</td>
</tr>
</tbody>
</table>

[383]: current_median_commercial_age_5_race_gender_salaried = commercial_salaried.
    -> groupby(['age_group_5', 'race_ethnicity', 'gender']).agg({'current_base_pay': [np.
    -> count_nonzero, np.median]})
    suppress(current_median_commercial_age_5_race_gender_salaried)

[383]:

<table>
<thead>
<tr>
<th>age_group_5</th>
<th>race_ethnicity</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>White (United States of America) Female</td>
<td>7.00</td>
<td>62000.00</td>
<td></td>
</tr>
<tr>
<td>25-29</td>
<td>White (United States of America) Female</td>
<td>25.00</td>
<td>76000.00</td>
<td></td>
</tr>
<tr>
<td>30-34</td>
<td>White (United States of America) Female</td>
<td>5.00</td>
<td>131097.12</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>7.00</td>
<td>97695.60</td>
<td></td>
</tr>
<tr>
<td>35-39</td>
<td>White (United States of America) Female</td>
<td>9.00</td>
<td>149101.00</td>
<td></td>
</tr>
<tr>
<td>40-44</td>
<td>White (United States of America) Female</td>
<td>6.00</td>
<td>126864.75</td>
<td></td>
</tr>
<tr>
<td>50-54</td>
<td>White (United States of America) Female</td>
<td>6.00</td>
<td>98281.24</td>
<td></td>
</tr>
<tr>
<td>55-59</td>
<td>White (United States of America) Male</td>
<td>5.00</td>
<td>97134.77</td>
<td></td>
</tr>
</tbody>
</table>

[384]: current_median_commercial_age_5_race_gender_hourly = commercial_hourly.
    -> groupby(['age_group_5', 'race_ethnicity', 'gender']).agg({'current_base_pay': [np.
    -> count_nonzero, np.median]})
    suppress(current_median_commercial_age_5_race_gender_hourly)

[384]:

<table>
<thead>
<tr>
<th>age_group_5</th>
<th>race_ethnicity</th>
<th>gender</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>count_nonzero</td>
</tr>
</tbody>
</table>

219
<table>
<thead>
<tr>
<th>age_group_5</th>
<th>race_ethnicity</th>
<th>gender</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>Black or African American (United States of America)</td>
<td>Male</td>
<td>22.36</td>
</tr>
<tr>
<td>25-29</td>
<td>White (United States of America)</td>
<td>Female</td>
<td>35.01</td>
</tr>
<tr>
<td>40-44</td>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>29.74</td>
</tr>
<tr>
<td>45-49</td>
<td>Black or African American (United States of America)</td>
<td>Male</td>
<td>22.39</td>
</tr>
<tr>
<td>50-54</td>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>23.27</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>Male</td>
<td>23.01</td>
</tr>
<tr>
<td>55-59</td>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>28.61</td>
</tr>
<tr>
<td>60-64</td>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>24.32</td>
</tr>
</tbody>
</table>

```python
[385]:
current_median_commercial_age_5_race_group_gender_salaried =
    ~commercial_salaried.groupby(['age_group_5', 'race_grouping', 'gender']).
    ~agg({'current_base_pay': [np.count_nonzero, np.median]})

[385]:
count_nonzero  median
<table>
<thead>
<tr>
<th>age_group_5</th>
<th>race_grouping</th>
<th>gender</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>white</td>
<td>Female</td>
<td>7.00</td>
</tr>
<tr>
<td>25-29</td>
<td>white</td>
<td>Female</td>
<td>25.00</td>
</tr>
<tr>
<td>30-34</td>
<td>white</td>
<td>Female</td>
<td>5.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>7.00</td>
</tr>
<tr>
<td>35-39</td>
<td>person of color</td>
<td>Male</td>
<td>5.00</td>
</tr>
<tr>
<td>40-44</td>
<td>white</td>
<td>Female</td>
<td>9.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>6.00</td>
</tr>
</tbody>
</table>
```
<table>
<thead>
<tr>
<th>Age Group</th>
<th>Race</th>
<th>Gender</th>
<th>Hourly Pay</th>
<th>Median Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>50-54</td>
<td>white</td>
<td>Female</td>
<td>6.00</td>
<td>98281.24</td>
</tr>
<tr>
<td>55-59</td>
<td>white</td>
<td>Male</td>
<td>5.00</td>
<td>97134.77</td>
</tr>
</tbody>
</table>

```python
[386]: current_median_commercial_age_5_race_group_gender_hourly = commercial_hourly.
        groupby(['age_group_5', 'race_grouping', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
        suppress(current_median_commercial_age_5_race_group_gender_hourly)
```

<table>
<thead>
<tr>
<th>Age Group</th>
<th>Race</th>
<th>Gender</th>
<th>Hourly Pay</th>
<th>Median Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>person of color Male</td>
<td>5.00</td>
<td>22.36</td>
<td></td>
</tr>
<tr>
<td>25-29</td>
<td>person of color Female</td>
<td>7.00</td>
<td>26.27</td>
<td></td>
</tr>
<tr>
<td></td>
<td>White Female</td>
<td>7.00</td>
<td>35.01</td>
<td></td>
</tr>
<tr>
<td>30-34</td>
<td>person of color Female</td>
<td>5.00</td>
<td>30.38</td>
<td></td>
</tr>
<tr>
<td>40-44</td>
<td>person of color Female</td>
<td>10.00</td>
<td>29.48</td>
<td></td>
</tr>
<tr>
<td>45-49</td>
<td>person of color Male</td>
<td>10.00</td>
<td>22.39</td>
<td></td>
</tr>
<tr>
<td>50-54</td>
<td>person of color Female</td>
<td>6.00</td>
<td>23.27</td>
<td></td>
</tr>
<tr>
<td></td>
<td>White Male</td>
<td>7.00</td>
<td>21.10</td>
<td></td>
</tr>
<tr>
<td>55-59</td>
<td>person of color Female</td>
<td>7.00</td>
<td>28.61</td>
<td></td>
</tr>
<tr>
<td>60-64</td>
<td>person of color Female</td>
<td>5.00</td>
<td>24.32</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>6.00</td>
<td>23.80</td>
<td></td>
</tr>
</tbody>
</table>

```python
[387]: current_median_commercial_age_10_race_gender_salaried = commercial_salaried.
        groupby(['age_group_10', 'race_ethnicity', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
        suppress(current_median_commercial_age_10_race_gender_salaried)
```

<table>
<thead>
<tr>
<th>Age Group</th>
<th>Race</th>
<th>Gender</th>
<th>Hourly Pay</th>
<th>Median Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>White (United States of America) Female</td>
<td>7.00</td>
<td>62000.00</td>
<td></td>
</tr>
<tr>
<td>25-34</td>
<td>Asian (United States of America) Female</td>
<td>5.00</td>
<td>90000.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>White (United States of America) Female</td>
<td>30.00</td>
<td>78691.50</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>10.00</td>
<td>96347.80</td>
<td></td>
</tr>
<tr>
<td>35-44</td>
<td>White (United States of America) Female</td>
<td>15.00</td>
<td>148729.50</td>
<td></td>
</tr>
<tr>
<td>45-54</td>
<td>White (United States of America) Female</td>
<td>10.00</td>
<td>98281.24</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>6.00</td>
<td>86195.95</td>
<td></td>
</tr>
<tr>
<td>55-64</td>
<td>White (United States of America) Female</td>
<td>5.00</td>
<td>96780.00</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>9.00</td>
<td>97514.43</td>
<td></td>
</tr>
</tbody>
</table>

```python
[388]: current_median_commercial_age_10_race_gender_hourly = commercial_hourly.
        groupby(['age_group_10', 'race_ethnicity', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
        suppress(current_median_commercial_age_10_race_gender_hourly)
```

<table>
<thead>
<tr>
<th>Age Group</th>
<th>Race</th>
<th>Gender</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>Black or African American (United States of America) Male</td>
<td>5.00</td>
</tr>
</tbody>
</table>
| 25-34     | Black or African American (United States of America) Female | }
<table>
<thead>
<tr>
<th>Age Group</th>
<th>Race/Ethnicity</th>
<th>Gender</th>
<th>Median Base Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>Black or African American</td>
<td>Male</td>
<td>22.36</td>
</tr>
<tr>
<td>25-34</td>
<td>Hispanic or Latino</td>
<td>Female</td>
<td>26.50</td>
</tr>
<tr>
<td></td>
<td>White</td>
<td>Female</td>
<td>28.13</td>
</tr>
<tr>
<td></td>
<td>White</td>
<td>Female</td>
<td>33.42</td>
</tr>
<tr>
<td>35-44</td>
<td>Black or African American</td>
<td>Female</td>
<td>28.74</td>
</tr>
<tr>
<td></td>
<td>White</td>
<td>Female</td>
<td>29.74</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td></td>
<td>24.84</td>
</tr>
<tr>
<td>45-54</td>
<td>Black or African American</td>
<td>Female</td>
<td>23.67</td>
</tr>
<tr>
<td></td>
<td>White</td>
<td>Female</td>
<td>22.39</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td></td>
<td>24.44</td>
</tr>
<tr>
<td>55-64</td>
<td>Black or African American</td>
<td>Female</td>
<td>24.99</td>
</tr>
<tr>
<td></td>
<td>White</td>
<td>Female</td>
<td>23.86</td>
</tr>
</tbody>
</table>

```python
[389]: current_median_commercial_age_10_race_group_gender_salaried = ~commercial_salaried.groupby(['age_group_10', 'race_grouping', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]}).suppress(current_median_commercial_age_10_race_group_gender_salaried)

[389]:
<table>
<thead>
<tr>
<th>Age Group</th>
<th>Race/Ethnicity</th>
<th>Gender</th>
<th>Count Nonzero</th>
<th>Median Base Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>white</td>
<td>Female</td>
<td>7.00</td>
<td>62000.00</td>
</tr>
<tr>
<td>25-34</td>
<td>person of color</td>
<td>Female</td>
<td>7.00</td>
<td>85000.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>30.00</td>
<td>78691.50</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>10.00</td>
<td>96347.80</td>
</tr>
<tr>
<td>35-44</td>
<td>person of color</td>
<td>Male</td>
<td>6.00</td>
<td>81976.52</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>15.00</td>
<td>148729.50</td>
</tr>
<tr>
<td>45-54</td>
<td>white</td>
<td>Female</td>
<td>10.00</td>
<td>98281.24</td>
</tr>
</tbody>
</table>
```
Male 6.00 86195.95
55-64 white  Female 5.00 96780.00
Male 9.00 97514.43

```python
[390]: current_median_commercial_age_10_race_group_gender_hourly = commercial_hourly.
    .groupby(['age_group_10', 'race_grouping', 'gender']).agg({'current_base_pay':
        [np.count_nonzero, np.median]})
    suppress(current_median_commercial_age_10_race_group_gender_hourly)
```

<table>
<thead>
<tr>
<th>age_group_10</th>
<th>race_grouping</th>
<th>gender</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;25</td>
<td>person of color</td>
<td>Male</td>
<td>5.00</td>
<td>22.36</td>
</tr>
<tr>
<td>25-34</td>
<td>person of color</td>
<td>Female</td>
<td>12.00</td>
<td>27.43</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Female</td>
<td>8.00</td>
<td>33.42</td>
</tr>
<tr>
<td>35-44</td>
<td>person of color</td>
<td>Female</td>
<td>13.00</td>
<td>29.74</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Male</td>
<td>7.00</td>
<td>23.12</td>
</tr>
<tr>
<td>45-54</td>
<td>person of color</td>
<td>Female</td>
<td>10.00</td>
<td>23.67</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Male</td>
<td>17.00</td>
<td>22.34</td>
</tr>
<tr>
<td>55-64</td>
<td>person of color</td>
<td>Female</td>
<td>12.00</td>
<td>24.99</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Male</td>
<td>10.00</td>
<td>23.86</td>
</tr>
</tbody>
</table>

1.6.6 Departments

```python
[391]: current_commercial_median_department_salaried = commercial_salaried.
    .groupby(['department']).agg({'current_base_pay': [np.count_nonzero, np.median]})
    suppress_median(current_commercial_median_department_salaried)
```

```excel
<table>
<thead>
<tr>
<th>department</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finance</td>
<td>8.00</td>
<td>90575.50</td>
</tr>
<tr>
<td>WP News Media Services</td>
<td>9.00</td>
<td>86104.69</td>
</tr>
<tr>
<td>Client Solutions</td>
<td>102.00</td>
<td>85633.86</td>
</tr>
<tr>
<td>Marketing</td>
<td>7.00</td>
<td>81196.11</td>
</tr>
<tr>
<td>Production</td>
<td>5.00</td>
<td>71665.06</td>
</tr>
</tbody>
</table>
```

```python
[392]: current_commercial_median_department_hourly = commercial_hourly.
    .groupby(['department']).agg({'current_base_pay': [np.count_nonzero, np.median]})
    suppress_median(current_commercial_median_department_hourly)
```

```excel
<table>
<thead>
<tr>
<th>department</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Public Relations</td>
<td>5.00</td>
<td>35.01</td>
</tr>
<tr>
<td>Client Solutions</td>
<td>62.00</td>
<td>29.41</td>
</tr>
<tr>
<td>Finance</td>
<td>23.00</td>
<td>29.23</td>
</tr>
<tr>
<td>Circulation</td>
<td>49.00</td>
<td>22.44</td>
</tr>
</tbody>
</table>
```

223
Current Commercial Median Department Gender Salaried:

```python
current_commercial_median_department_gender_salaried = commercial_salaried.
→groupby(['department', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_commercial_median_department_gender_salaried)
```

<table>
<thead>
<tr>
<th>Department</th>
<th>Gender</th>
<th>Count Nonzero</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Finance</td>
<td>Female</td>
<td>5.00</td>
<td>96780.00</td>
</tr>
<tr>
<td>Client Solutions</td>
<td>Male</td>
<td>31.00</td>
<td>90000.00</td>
</tr>
<tr>
<td>WP News Media Services</td>
<td>Male</td>
<td>5.00</td>
<td>85899.92</td>
</tr>
<tr>
<td>Client Solutions</td>
<td>Female</td>
<td>71.00</td>
<td>85000.00</td>
</tr>
</tbody>
</table>

Current Commercial Median Department Gender Hourly:

```python
current_commercial_median_department_gender_hourly = commercial_hourly.
→groupby(['department', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_commercial_median_department_gender_hourly)
```

<table>
<thead>
<tr>
<th>Department</th>
<th>Gender</th>
<th>Count Nonzero</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Public Relations</td>
<td>Female</td>
<td>5.00</td>
<td>35.01</td>
</tr>
<tr>
<td>Client Solutions</td>
<td>Male</td>
<td>24.00</td>
<td>30.13</td>
</tr>
<tr>
<td>Finance</td>
<td>Female</td>
<td>17.00</td>
<td>29.23</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>6.00</td>
<td>28.85</td>
</tr>
<tr>
<td>Client Solutions</td>
<td>Female</td>
<td>38.00</td>
<td>28.83</td>
</tr>
<tr>
<td>Circulation</td>
<td>Female</td>
<td>9.00</td>
<td>23.19</td>
</tr>
<tr>
<td></td>
<td>Male</td>
<td>40.00</td>
<td>22.40</td>
</tr>
</tbody>
</table>

Current Commercial Median Department Race Salaried:

```python
current_commercial_median_department_race_salaried = commercial_salaried.
→groupby(['department', 'race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_commercial_median_department_race_salaried)
```

<table>
<thead>
<tr>
<th>Department</th>
<th>Race Ethnicity</th>
<th>Count Nonzero</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client Solutions</td>
<td>White</td>
<td>79.00</td>
<td>90000.00</td>
</tr>
<tr>
<td>WP News Media Services</td>
<td>White</td>
<td>8.00</td>
<td>90000.00</td>
</tr>
<tr>
<td>Client Solutions</td>
<td>Black or African American</td>
<td>10.00</td>
<td>85000.00</td>
</tr>
<tr>
<td>Marketing</td>
<td>White</td>
<td>5.00</td>
<td>85000.00</td>
</tr>
<tr>
<td>Client Solutions</td>
<td>Asian</td>
<td>9.00</td>
<td>90000.00</td>
</tr>
</tbody>
</table>

Median Department Race Ethnicity:

<table>
<thead>
<tr>
<th>Department</th>
<th>Race Ethnicity</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client Solutions</td>
<td>White</td>
<td>90000.00</td>
</tr>
</tbody>
</table>
current_commercial_median_department_race_hourly = commercial_hourly.
  groupby(['department', 'race_ethnicity']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_commercial_median_department_race_hourly)

count_nonzero \ 
department       race_ethnicity
Client Solutions White (United States of America) 24.00
Finance          White (United States of America) 5.00
                  Black or African American (United States of Ame... 16.00
Client Solutions Hispanic or Latino (United States of America) 6.00
                  Black or African American (United States of Ame... 25.00
                  Asian (United States of America) 5.00
Circulation      White (United States of America) 8.00
                  Black or African American (United States of Ame... 35.00

median

department       race_ethnicity
Client Solutions White (United States of America) 31.00
Finance          White (United States of America) 29.49
                  Black or African American (United States of Ame... 29.06
Client Solutions Hispanic or Latino (United States of America) 28.51
                  Black or African American (United States of Ame... 26.99
                  Asian (United States of America) 26.30
Circulation      White (United States of America) 22.80
                  Black or African American (United States of Ame... 22.36

suppress_median(current_commercial_median_department_race_gender_salaried)
count_nonzero \
  department race_ethnicity gender
Client Solutions White (United States of America) Male
  22.00 Black or African American (United States of America) Female
  6.00 White (United States of America) Female
  57.00 Asian (United States of America) Female
  5.00

median 
  department race_ethnicity gender
Client Solutions White (United States of America) Male
  98893.80 Black or African American (United States of America) Female
  92158.00 White (United States of America) Female
  86613.00 Asian (United States of America) Female
  80000.00

 current_commercial_median_department_race_gender_hourly = commercial_hourly.
  ~groupby(['department', 'race_ethnicity', 'gender']).agg({'current_base_pay':
  ~[np.count_nonzero, np.median]})
suppress_median(current_commercial_median_department_race_gender_hourly)

count_nonzero \
  department race_ethnicity gender
Client Solutions White (United States of America) Female
  13.00
  11.00 Finance Black or African American (United States of America) Female
  12.00
Client Solutions Hispanic or Latino (United States of America) Female
  6.00 Black or African American (United States of America) Male
  9.00
  16.00 Circulation Black or African American (United States of America) Female
  9.00
  8.00 White (United States of America) Male
  8.00 Black or African American (United States of America) Male
  26.00
<table>
<thead>
<tr>
<th>Department</th>
<th>Race/Ethnicity</th>
<th>Gender</th>
<th>Median Base Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client Solutions</td>
<td>White (United States of America)</td>
<td>Female</td>
<td>31.68</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>30.77</td>
</tr>
<tr>
<td>Finance</td>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>29.06</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>28.16</td>
</tr>
<tr>
<td>Client Solutions</td>
<td>Hispanic or Latino (United States of America)</td>
<td>Female</td>
<td>28.51</td>
</tr>
<tr>
<td></td>
<td>Black or African American (United States of America)</td>
<td>Male</td>
<td>28.16</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Female</td>
<td>25.95</td>
</tr>
<tr>
<td>Circulation</td>
<td>Black or African American (United States of America)</td>
<td>Female</td>
<td>23.19</td>
</tr>
<tr>
<td></td>
<td>White (United States of America)</td>
<td>Male</td>
<td>22.80</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Female</td>
<td>22.35</td>
</tr>
</tbody>
</table>

[399]:
```python
current_commercial_median_department_race_group_gender_salaried =
commercial_salaried.groupby(['department', 'race_grouping', 'gender']).
agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_commercial_median_department_race_group_gender_salaried)
```

[399]:
```python
<table>
<thead>
<tr>
<th>Department</th>
<th>Race/Ethnicity</th>
<th>Gender</th>
<th>Count Nonzero</th>
<th>Median Base Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client Solutions</td>
<td>white</td>
<td>Male</td>
<td>22.00</td>
<td>98893.80</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Female</td>
<td>57.00</td>
<td>86613.00</td>
</tr>
<tr>
<td></td>
<td>person of color</td>
<td>Female</td>
<td>13.00</td>
<td>80000.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>9.00</td>
<td>76139.41</td>
</tr>
</tbody>
</table>
```

[400]:
```python
current_commercial_median_department_race_group_gender_hourly =
commercial_hourly.groupby(['department', 'race_grouping', 'gender']).
agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_commercial_median_department_race_group_gender_hourly)
```

[400]:
```python
<table>
<thead>
<tr>
<th>Department</th>
<th>Race/Ethnicity</th>
<th>Gender</th>
<th>Count Nonzero</th>
<th>Median Base Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client Solutions</td>
<td>white</td>
<td>Female</td>
<td>13.00</td>
<td>31.68</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Male</td>
<td>11.00</td>
<td>30.77</td>
</tr>
<tr>
<td>Finance</td>
<td>person of color</td>
<td>Female</td>
<td>13.00</td>
<td>28.89</td>
</tr>
<tr>
<td>Client Solutions</td>
<td>person of color</td>
<td>Male</td>
<td>13.00</td>
<td>27.05</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Female</td>
<td>25.00</td>
<td>26.34</td>
</tr>
<tr>
<td>Circulation</td>
<td>person of color</td>
<td>Female</td>
<td>9.00</td>
<td>23.19</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>Male</td>
<td>8.00</td>
<td>22.80</td>
</tr>
<tr>
<td></td>
<td>person of color</td>
<td>Male</td>
<td>30.00</td>
<td>22.35</td>
</tr>
</tbody>
</table>
```

227
current_commercial_median_department_race_gender_age5_salaried =
commercial_salaried.
groupby(['department', 'race_ethnicity', 'gender', 'age_group_5']).
agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_commercial_median_department_race_gender_age5_salaried)

count_nonzero \
department   race_ethnicity       gender age_group_5
Client Solutions White (United States of America) Female 35-39 9.00 40-44
       6.00         50-54
       5.00         Male 30-34
       5.00  Female 25-29
       23.00 <25
       6.00

current_commercial_median_department_race_gender_age5_hourly =
commercial_hourly.
groupby(['department', 'race_ethnicity', 'gender', 'age_group_5']).
agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_commercial_median_department_race_gender_age5_hourly)

count_nonzero \
department   race_ethnicity       gender age_group_5
Client Solutions White (United States of America) Female 35-39 149101.00 40-44 126864.75
       50-54 105893.00
       Male 30-34 100000.00
       Female 25-29 75000.00
       <25 61000.00

median

current_commercial_median_department_race_gender_age5_salaried =
current_commercial_median_department_race_gender_age5_salaried

current_commercial_median_department_race_gender_age5_hourly =
current_commercial_median_department_race_gender_age5_hourly

current_commercial_median_department_race_gender_age5_salaried =
current_commercial_median_department_race_gender_age5_salaried

current_commercial_median_department_race_gender_age5_hourly =
current_commercial_median_department_race_gender_age5_hourly

Client Solutions White (United States of America) Female 25-29 5.00

median
current_commercial_median_department_race_gender_age5_salaried =
current_commercial_median_department_race_gender_age5_salaried

current_commercial_median_department_race_gender_age5_salaried =
current_commercial_median_department_race_gender_age5_salaried

current_commercial_median_department_race_gender_age5_salaried =
current_commercial_median_department_race_gender_age5_salaried

current_commercial_median_department_race_gender_age5_salaried =
current_commercial_median_department_race_gender_age5_salaried

current_commercial_median_department_race_gender_age5_salaried =
current_commercial_median_department_race_gender_age5_salaried

current_commercial_median_department_race_gender_age5_salaried =
current_commercial_median_department_race_gender_age5_salaried

current_commercial_median_department_race_gender_age5_salaried =
current_commercial_median_department_race_gender_age5_salaried

current_commercial_median_department_race_gender_age5_salaried =
current_commercial_median_department_race_gender_age5_salaried

Client Solutions White (United States of America) Female 25-29 5.00

median
### 1.6.7 Job profiles

```python
[405]: current_commercial_median_job_salaried = commercial_salaried.
        ~groupby(['job_profile_current']).agg({'current_base_pay': [np.count_nonzero, np.median]})
        suppress_median(current_commercial_median_job_salaried)
```

<table>
<thead>
<tr>
<th>job_profile_current</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>450220 - Sales Representative</td>
<td>25.00</td>
<td>153987.30</td>
</tr>
<tr>
<td>350227 - Custom Content Writer</td>
<td>7.00</td>
<td>100000.00</td>
</tr>
<tr>
<td>551104 - Senior Financial Accountant</td>
<td>5.00</td>
<td>90566.00</td>
</tr>
</tbody>
</table>
450120 - Account Manager 26.00 88644.94
390110 - Multiplatform Editor 9.00 86104.69
280228 - Designer 7.00 85000.00
340227 - Artist 5.00 75035.28
481205 - Digital Analyst 5.00 75000.00
660127 - Make-Up Person 5.00 71665.06
231303 - Client Service Manager 15.00 67095.60

[406]: current_commercial_median_job_hourly = commercial_hourly.
groupby(['job_profile_current']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_commercial_median_job_hourly)

[406]:

<table>
<thead>
<tr>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>341027 - Desktop Publisher</td>
<td>6.00</td>
</tr>
<tr>
<td>574504 - Senior Accounting Specialist</td>
<td>11.00</td>
</tr>
<tr>
<td>565005 - Accounting Specialist</td>
<td>12.00</td>
</tr>
<tr>
<td>470121 - Account Executive</td>
<td>16.00</td>
</tr>
<tr>
<td>600318 - Circulation Driver (Class A)</td>
<td>35.00</td>
</tr>
</tbody>
</table>

[407]: current_commercial_median_job_gender_salaried = commercial_salaried.
groupby(['job_profile_current', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_commercial_median_job_gender_salaried)

[407]:

<table>
<thead>
<tr>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>450220 - Sales Representative Male</td>
<td>6.00</td>
</tr>
<tr>
<td>450120 - Account Manager Female</td>
<td>19.00</td>
</tr>
<tr>
<td>450120 - Account Manager Female</td>
<td>17.00</td>
</tr>
<tr>
<td>390110 - Multiplatform Editor Male</td>
<td>5.00</td>
</tr>
<tr>
<td>450120 - Account Manager Male</td>
<td>9.00</td>
</tr>
<tr>
<td>231303 - Client Service Manager Female</td>
<td>13.00</td>
</tr>
</tbody>
</table>

[408]: current_commercial_median_job_gender_hourly = commercial_hourly.
groupby(['job_profile_current', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_commercial_median_job_gender_hourly)

[408]:

<table>
<thead>
<tr>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>574504 - Senior Accounting Specialist Female</td>
<td>10.00</td>
</tr>
<tr>
<td>565005 - Accounting Specialist Male</td>
<td>5.00</td>
</tr>
<tr>
<td>470121 - Account Executive Female</td>
<td>7.00</td>
</tr>
<tr>
<td>470121 - Account Executive Male</td>
<td>15.00</td>
</tr>
<tr>
<td>600318 - Circulation Driver (Class A) Male</td>
<td>34.00</td>
</tr>
</tbody>
</table>

[409]:

230
```python
current_commercial_median_job_race_salaried = commercial_salaried.
    groupby(['job_profile_current', 'race_ethnicity']).agg({'current_base_pay':
    [np.count_nonzero, np.median]})
suppress_median(current_commercial_median_job_race_salaried)
```

```plaintext
[409]:
    count_nonzero \ 
    job_profile_current                    race_ethnicity
450220 - Sales Representative White (United States of America) 23.00
350227 - Custom Content Writer White (United States of America) 6.00
450120 - Account Manager White (United States of America) 15.00
390110 - Multiplatform Editor White (United States of America) 8.00
450120 - Account Manager Black or African American (United States of Ame... 7.00
231303 - Client Service Manager White (United States of America) 14.00

    median
    job_profile_current                    race_ethnicity
450220 - Sales Representative White (United States of America) 150780.00
350227 - Custom Content Writer White (United States of America) 100000.00
450120 - Account Manager White (United States of America) 90669.48
390110 - Multiplatform Editor White (United States of America) 88301.65
450120 - Account Manager Black or African American (United States of Ame... 85417.73
231303 - Client Service Manager White (United States of America) 65548.47
```

```python
[410]:
    current_commercial_median_job_race_hourly = commercial_hourly.
    groupby(['job_profile_current', 'race_ethnicity']).agg({'current_base_pay':
    [np.count_nonzero, np.median]})
suppress_median(current_commercial_median_job_race_hourly)
```

```plaintext
[410]:
    count_nonzero \ 
    job_profile_current                    race_ethnicity
574504 - Senior Accounting Specialist Black or African American (United States of Ame... 8.00
565005 - Accounting Specialist Black or African American (United States of Ame... 7.00
470121 - Account Executive White (United States of America) 5.00
```

231
<table>
<thead>
<tr>
<th>Job Profile</th>
<th>Race Ethnicity</th>
<th>Gender</th>
<th>Median Base Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>600318 - Circulation Driver</td>
<td>Black or African American</td>
<td>Male</td>
<td>9.00</td>
</tr>
<tr>
<td>600318 - Circulation Driver</td>
<td>Black or African American</td>
<td>Female</td>
<td>23.00</td>
</tr>
<tr>
<td>574504 - Senior Accounting Specialist</td>
<td>Black or African American</td>
<td>Male</td>
<td>30.06</td>
</tr>
<tr>
<td>565005 - Accounting Specialist</td>
<td>Black or African American</td>
<td>Female</td>
<td>26.04</td>
</tr>
<tr>
<td>470121 - Account Executive</td>
<td>White</td>
<td>Male</td>
<td>25.36</td>
</tr>
<tr>
<td>470121 - Account Executive</td>
<td>White</td>
<td>Female</td>
<td>24.70</td>
</tr>
<tr>
<td>565005 - Accounting Specialist</td>
<td>Black or African American</td>
<td>Male</td>
<td>22.98</td>
</tr>
<tr>
<td>565005 - Accounting Specialist</td>
<td>Black or African American</td>
<td>Female</td>
<td>22.36</td>
</tr>
<tr>
<td>440220 - Sales Representative</td>
<td>White</td>
<td>Male</td>
<td>5.00</td>
</tr>
<tr>
<td>440220 - Sales Representative</td>
<td>White</td>
<td>Female</td>
<td>18.00</td>
</tr>
<tr>
<td>450120 - Account Manager</td>
<td>White</td>
<td>Male</td>
<td>11.00</td>
</tr>
<tr>
<td>231303 - Client Service Manager</td>
<td>White</td>
<td>Male</td>
<td>12.00</td>
</tr>
<tr>
<td>231303 - Client Service Manager</td>
<td>White</td>
<td>Female</td>
<td>24.70</td>
</tr>
</tbody>
</table>

```python
[411]: current_commercial_median_job_race_gender_salaried = commercial_salaried.
    -> groupby(['job_profile_current', 'race_ethnicity', 'gender']).
    -> agg({'current_base_pay': [np.count_nonzero, np.median]})
    suppress_median(current_commercial_median_job_race_gender_salaried)
```

```python
[411]: count_nonzero \
    job_profile_current race_ethnicity gender
  450220 - Sales Representative White (United States of America) Male 5.00
         White (United States of America) Female 18.00
  450120 - Account Manager     White (United States of America) Female 11.00
  231303 - Client Service Manager White (United States of America) Female 12.00
```

<table>
<thead>
<tr>
<th>Job Profile</th>
<th>Race Ethnicity</th>
<th>Gender</th>
<th>Median Base Pay</th>
</tr>
</thead>
<tbody>
<tr>
<td>450220 - Sales Representative</td>
<td>White</td>
<td>Male</td>
<td>155300.00</td>
</tr>
<tr>
<td>450220 - Sales Representative</td>
<td>White</td>
<td>Female</td>
<td>149940.50</td>
</tr>
<tr>
<td>450120 - Account Manager</td>
<td>White</td>
<td>Male</td>
<td>90110.00</td>
</tr>
<tr>
<td>231303 - Client Service Manager</td>
<td>White</td>
<td>Male</td>
<td>66000.67</td>
</tr>
<tr>
<td>231303 - Client Service Manager</td>
<td>White</td>
<td>Female</td>
<td>66000.67</td>
</tr>
</tbody>
</table>
current_commercial_median_job_race_gender_hourly = commercial_hourly.
groupby(['job_profile_current', 'race_ethnicity', 'gender']).
agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_commercial_median_job_race_gender_hourly)

current_commercial_median_job_race_group_gender_salaried = commercial_salaried.
groupby(['desk', 'race_grouping', 'gender']).agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_commercial_median_job_race_group_gender_salaried)

current_commercial_median_job_race_group_gender_hourly = commercial_hourly.
groupby(['job_profile_current', 'race_grouping', 'gender']).
agg({'current_base_pay': [np.count_nonzero, np.median]})
suppress_median(current_commercial_median_job_race_group_gender_hourly)

233
Count Nonzero

<table>
<thead>
<tr>
<th>Job Profile Current</th>
<th>Race Grouping</th>
<th>Gender</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>574504 - Senior Accounting Specialist</td>
<td>Person of color Female</td>
<td>7.00</td>
<td></td>
</tr>
<tr>
<td>565005 - Accounting Specialist</td>
<td>Person of color Female</td>
<td>6.00</td>
<td></td>
</tr>
<tr>
<td>470121 - Account Executive</td>
<td>Person of color Female</td>
<td>11.00</td>
<td></td>
</tr>
<tr>
<td>600318 - Circulation Driver (Class A)</td>
<td>White Male</td>
<td>7.00</td>
<td></td>
</tr>
</tbody>
</table>

Median

<table>
<thead>
<tr>
<th>Job Profile Current</th>
<th>Race Grouping</th>
<th>Gender</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>574504 - Senior Accounting Specialist</td>
<td>Person of color Female</td>
<td>29.74</td>
<td></td>
</tr>
<tr>
<td>565005 - Accounting Specialist</td>
<td>Person of color Female</td>
<td>25.84</td>
<td></td>
</tr>
<tr>
<td>470121 - Account Executive</td>
<td>Person of color Female</td>
<td>24.70</td>
<td></td>
</tr>
<tr>
<td>600318 - Circulation Driver (Class A)</td>
<td>White Male</td>
<td>22.98</td>
<td></td>
</tr>
</tbody>
</table>

Suppress Median

Median

<table>
<thead>
<tr>
<th>Job Profile Current</th>
<th>Race Ethnicity</th>
<th>Gender</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>450220 - Sales Representative</td>
<td>White (United States of America) Female</td>
<td>35-39</td>
<td>149940.50</td>
</tr>
<tr>
<td>231303 - Client Service Manager</td>
<td>White (United States of America) Female</td>
<td>25-29</td>
<td>149940.50</td>
</tr>
</tbody>
</table>

Median

<table>
<thead>
<tr>
<th>Job Profile Current</th>
<th>Race Ethnicity</th>
<th>Gender</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>450220 - Sales Representative</td>
<td>White (United States of America) Female</td>
<td>35-39</td>
<td>66212.61</td>
</tr>
<tr>
<td>231303 - Client Service Manager</td>
<td>White (United States of America) Female</td>
<td>25-29</td>
<td>66212.61</td>
</tr>
</tbody>
</table>

Suppress Median

Median

<table>
<thead>
<tr>
<th>Job Profile Current</th>
<th>Race Ethnicity</th>
<th>Gender</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>600318 - Circulation Driver (Class A)</td>
<td>Black or African American (United States of America) Male</td>
<td>60-64</td>
<td>6.00</td>
</tr>
<tr>
<td></td>
<td></td>
<td>45-49</td>
<td>7.00</td>
</tr>
<tr>
<td>Median</td>
<td>Race/Ethnicity</td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>---------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>600318 - Circulation Driver (Class A) Black or African American (United States of America)</td>
<td>Male 60-64 23.80</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>45-49 21.51</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```python
[417]: current_commercial_median_job_race_group_gender_age5_salaried = commercial_salaried.
    groupby(['job_profile_current', 'race_grouping', 'gender', 'age_group_5']).
    agg({'current_base_pay': [np.count_nonzero, np.median]})
    suppress_median(current_commercial_median_job_race_group_gender_age5_salaried)
```

<table>
<thead>
<tr>
<th>Median</th>
<th>Race/Ethnicity</th>
</tr>
</thead>
<tbody>
<tr>
<td>450220 - Sales Representative white</td>
<td>Female 35-39 8.00</td>
</tr>
<tr>
<td>231303 - Client Service Manager white</td>
<td>Female 25-29 8.00</td>
</tr>
</tbody>
</table>

```python
[418]: current_commercial_median_job_race_group_gender_age5_hourly = commercial_hourly.
    groupby(['job_profile_current', 'race_grouping', 'gender', 'age_group_5']).
    agg({'current_base_pay': [np.count_nonzero, np.median]})
    suppress_median(current_commercial_median_job_race_group_gender_age5_hourly)
```

<table>
<thead>
<tr>
<th>Median</th>
<th>Race/Ethnicity</th>
</tr>
</thead>
<tbody>
<tr>
<td>600318 - Circulation Driver (Class A) person of color</td>
<td>Male 60-64 6.00</td>
</tr>
<tr>
<td></td>
<td>45-49 7.00</td>
</tr>
</tbody>
</table>

```python
[419]: commercial_ratings = ratings_combined[ratings_combined['dept'] == "Commercial"]
[420]: commercial_ratings_gender = commercial_ratings.groupby(['gender']).
    agg({'performance_rating': [np.count_nonzero, np.median]})
```

### 1.6.8 Performance evaluations

<table>
<thead>
<tr>
<th>Median</th>
<th>Race/Ethnicity</th>
</tr>
</thead>
<tbody>
<tr>
<td>600318 - Circulation Driver (Class A) person of color</td>
<td>Male 60-64 23.80</td>
</tr>
<tr>
<td></td>
<td>45-49 21.51</td>
</tr>
</tbody>
</table>
```python
[420]:
    performance_rating
    count_nonzero median
    gender
    Female  1308.00  3.30
    Male  984.00  3.20

[421]:
    commercial_ratings_race = commercial_ratings.groupby(['race_ethnicity']).
    agg({'performance_rating': [np.count_nonzero, np.median]})
    suppress_median(commercial_ratings_race)

[421]:

    count_nonzero median
count
race_ethnicity
Asian (United States of America)  168.00  3.30
Two or More Races (United States of America)  36.00  3.30
White (United States of America)  1096.00  3.30
Black or African American (United States of Am...  860.00  3.20
Hispanic or Latino (United States of America)  96.00  3.15
Prefer Not to Disclose (United States of America)  28.00  3.00

[422]:
    commercial_ratings_race_gender = commercial_ratings.
    groupby(['race_ethnicity', 'gender']).agg({'performance_rating': [np.
    count_nonzero, np.median]})
    suppress(commercial_ratings_race_gender)

[422]:

    count_nonzero median
race_ethnicity gender
Asian (United States of America) Female  116.00  3.30
       Male  52.00  
Black or African American (United States of Am... Female  408.00  3.20
       Male  452.00  
Hispanic or Latino (United States of America) Female  56.00  3.15
       Male  40.00  
Prefer Not to Disclose (United States of America) Female  16.00  nan
       Male  12.00  
Two or More Races (United States of America) Female  20.00  nan
       Male  16.00  
White (United States of America) Female  684.00  3.30
       Male  412.00  
```
### Two or More Races (United States of America)

<table>
<thead>
<tr>
<th></th>
<th>Female</th>
<th>Male</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3.30</td>
<td>3.35</td>
</tr>
</tbody>
</table>

### White (United States of America)

<table>
<thead>
<tr>
<th></th>
<th>Female</th>
<th>Male</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>3.30</td>
<td>3.30</td>
</tr>
</tbody>
</table>

#### 1.6.9 Pay changes

```python
[423]:
commercial_change = ▼
    ~reason_for_change_combined[reason_for_change_combined['dept'] == ▼
    ~'Commercial']

[424]:
commercial_change_gender = commercial_change.
    ~groupby(["business_process_reason", 'gender']).agg({'business_process_reason':
        ~ [np.count_nonzero]})
    suppress_count(commercial_change_gender)
```

<table>
<thead>
<tr>
<th>business_process_reason</th>
<th>gender</th>
<th>count_nonzero</th>
</tr>
</thead>
<tbody>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Cont...</td>
<td>Female</td>
<td>475</td>
</tr>
<tr>
<td>Merit &gt; Performance &gt; Annual Performance Appraisal</td>
<td>Female</td>
<td>295</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Chan...</td>
<td>Female</td>
<td>198</td>
</tr>
<tr>
<td>Promotion &gt; Promotion &gt; Promotion</td>
<td>Female</td>
<td>144</td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Move to another Manager</td>
<td>Female</td>
<td>123</td>
</tr>
<tr>
<td>Data Change &gt; Data Change &gt; Change Job Details</td>
<td>Female</td>
<td>85</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Chan...</td>
<td>Male</td>
<td>85</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire &gt; Fill Vacancy</td>
<td>Female</td>
<td>70</td>
</tr>
<tr>
<td>Data Change &gt; Data Change &gt; Change Job Details</td>
<td>Male</td>
<td>61</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire &gt; Fill Vacancy</td>
<td>Male</td>
<td>58</td>
</tr>
<tr>
<td>Promotion &gt; Promotion &gt; Promotion</td>
<td>Male</td>
<td>52</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire &gt; New Position</td>
<td>Female</td>
<td>31</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Job ...</td>
<td>Male</td>
<td>20</td>
</tr>
<tr>
<td>Transfer &gt; Transfer &gt; Transfer between companies</td>
<td>Female</td>
<td>18</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Incr...</td>
<td>Male</td>
<td>15</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Job ...</td>
<td>Female</td>
<td>11</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Perf...</td>
<td>Male</td>
<td>7</td>
</tr>
<tr>
<td>Hire Employee &gt; New Hire &gt; Conversion</td>
<td>Female</td>
<td>6</td>
</tr>
<tr>
<td>Hire Employee &gt; Rehire &gt; Fill Vacancy</td>
<td>Female</td>
<td>6</td>
</tr>
<tr>
<td>Request Compensation Change &gt; Adjustment &gt; Job ...</td>
<td>Male</td>
<td>5</td>
</tr>
</tbody>
</table>
```

[425]:

237
commercial_change_race =
commercial_change[commercial_change['business_process_reason'] == 'Merit > Performance > Annual Performance Appraisal'].
groupby(['business_process_reason','race_ethnicity']).
agg({'business_process_reason': [np.count_nonzero]})
suppress_count(commercial_change_race)

[425]:
count_nonzero
business_process_reason      race_ethnicity
Merit > Performance > Annual Performance Appraisal Black or African American (United States of America) 239
White (United States of America) 220
Asian (United States of America) 36
Hispanic or Latino (United States of America) 19

[426]:
count_nonzero
business_process_reason      race_ethnicity
gender
Merit > Performance > Annual Performance Appraisal White (United States of America) Female 132
Black or African American (United States of America) Female 126
Male 113
White (United States of America) Male 88
Asian (United States of America) Female 19
Male 17
Hispanic or Latino (United States of America) Male 10
Female 9

1.6.10 Performance evaluations x merit raises

[427]: import re
reason_for_change_combined['merit_raises'] =
reason_for_change_combined['business_process_reason'].str.contains('Merit',
re.IGNORECASE)
twenty14 = np.datetime64('2016-04-01')
twenty15 = np.datetime64('2017-04-01')
twenty16 = np.datetime64('2018-04-01')
twenty17 = np.datetime64('2019-04-01')
twenty18 = np.datetime64('2020-04-01')

def raise_time(row):
    if row['effective_date'] < twenty14:
        return 'before 2015'
    if row['effective_date'] < twenty15:
        return '2015'
    if row['effective_date'] < twenty16:
        return '2016'
    if row['effective_date'] < twenty17:
        return '2017'
    if row['effective_date'] < twenty18:
        return '2018'
    return 'unknown'

reason_for_change_combined['raise_after'] = reason_for_change_combined.apply(lambda row: raise_time(row), axis=1)

merit_raises_commercial_gender_salaried = reason_for_change_combined[(reason_for_change_combined['merit_raises'] == True) & (reason_for_change_combined['dept'] == 'Commercial') & (reason_for_change_combined['pay_rate_type'] == 'Salaried')].groupby(['gender']).agg({'base_pay_change': [np.count_nonzero, np.median]})

merit_raises_commercial_gender_salaried

<table>
<thead>
<tr>
<th>Gender</th>
<th>Count Nonzero</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>97.00</td>
<td>1317.48</td>
</tr>
<tr>
<td>Male</td>
<td>74.00</td>
<td>1205.07</td>
</tr>
</tbody>
</table>

merit_raises_commercial_gender_hourly = reason_for_change_combined[(reason_for_change_combined['merit_raises'] == True) & (reason_for_change_combined['dept'] == 'Commercial') & (reason_for_change_combined['pay_rate_type'] == 'Hourly')].groupby(['gender']).agg({'base_pay_change': [np.count_nonzero, np.median]})

merit_raises_commercial_gender_hourly

<table>
<thead>
<tr>
<th>Gender</th>
<th>Count Nonzero</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>170.00</td>
<td>0.42</td>
</tr>
<tr>
<td>Male</td>
<td>138.00</td>
<td>0.33</td>
</tr>
</tbody>
</table>
merit_raises_commercial_race_salaried =
   reason_for_change_combined[(reason_for_change_combined['merit_raises'] == True) &
   (reason_for_change_combined['dept'] == 'Commercial') &
   (reason_for_change_combined['pay_rate_type'] == 'Salaried')].
   groupby(['race_ethnicity']).agg({'base_pay_change': [np.count_nonzero, np.
   median]})
suppress_median(merit_raises_commercial_race_salaried)

count_nonzero median
race_ethnicity
Asian (United States of America) 23.00 1375.00
Hispanic or Latino (United States of America) 6.00 1321.85
White (United States of America) 110.00 1286.88
Black or African American (United States of America) 30.00 1117.12

merit_raises_commercial_race_hourly =
   reason_for_change_combined[(reason_for_change_combined['merit_raises'] == True) &
   (reason_for_change_combined['dept'] == 'Commercial') &
   (reason_for_change_combined['pay_rate_type'] == 'Hourly')].
   groupby(['race_ethnicity']).agg({'base_pay_change': [np.count_nonzero, np.
   median]})
suppress_median(merit_raises_commercial_race_hourly)

count_nonzero median
race_ethnicity
Asian (United States of America) 11.00 0.45
White (United States of America) 85.00 0.42
Hispanic or Latino (United States of America) 11.00 0.37
Black or African American (United States of America) 197.00 0.35

merit_raises_commercial_race_group_salaried =
   reason_for_change_combined[(reason_for_change_combined['merit_raises'] == True) &
   (reason_for_change_combined['dept'] == 'Commercial') &
   (reason_for_change_combined['pay_rate_type'] == 'Salaried')].
   groupby(['race_grouping']).agg({'base_pay_change': [np.count_nonzero, np.
   median]})
suppress_median(merit_raises_commercial_race_group_salaried)

count_nonzero median
race_grouping
white 110.00 1286.88
person of color 60.00 1225.00

merit_raises_commercial_race_group_hourly =
   reason_for_change_combined[(reason_for_change_combined['merit_raises'] == True) &
   (reason_for_change_combined['dept'] == 'Commercial') &
   (reason_for_change_combined['pay_rate_type'] == 'Hourly')].
   groupby(['race_grouping']).agg({'base_pay_change': [np.count_nonzero, np.
   median]})
suppress_median(merit_raises_commercial_race_group_hourly)
### count_nonzero median

<table>
<thead>
<tr>
<th>race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>white</td>
<td>85.00</td>
<td>0.42</td>
</tr>
<tr>
<td>person of color</td>
<td>223.00</td>
<td>0.35</td>
</tr>
</tbody>
</table>

### merit_raises_commercial_gender_race_group_salaried

```python
definition = reason_for_change_combined['merit_raises']
~reason_for_change_combined['reason_for_change_combined']['merit_raises'] == True & (reason_for_change_combined['dept'] == 'Commercial') &
~(reason_for_change_combined['pay_rate_type'] == 'Salaried')].
groupby(['gender', 'race_grouping']).agg({'base_pay_change': [np.count_nonzero, np.median]})
suppress_median(merit_raises_commercial_gender_race_group_salaried)
```

### count_nonzero median

<table>
<thead>
<tr>
<th>gender</th>
<th>race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>white</td>
<td>69.00</td>
<td>1317.48</td>
</tr>
<tr>
<td></td>
<td>person of color</td>
<td>27.00</td>
<td>1305.00</td>
</tr>
<tr>
<td>Male</td>
<td>white</td>
<td>41.00</td>
<td>1282.47</td>
</tr>
<tr>
<td></td>
<td>person of color</td>
<td>33.00</td>
<td>1134.24</td>
</tr>
</tbody>
</table>

### merit_raises_commercial_gender_race_group_hourly

```python
definition = reason_for_change_combined['merit_raises']
~reason_for_change_combined['reason_for_change_combined']['merit_raises'] == True & (reason_for_change_combined['dept'] == 'Commercial') &
~(reason_for_change_combined['pay_rate_type'] == 'Hourly')].
groupby(['gender', 'race_grouping']).agg({'base_pay_change': [np.count_nonzero, np.median]})
suppress_median(merit_raises_commercial_gender_race_group_hourly)
```

### count_nonzero median

<table>
<thead>
<tr>
<th>gender</th>
<th>race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>white</td>
<td>44.00</td>
<td>0.52</td>
</tr>
<tr>
<td></td>
<td>person of color</td>
<td>126.00</td>
<td>0.38</td>
</tr>
<tr>
<td>Male</td>
<td>white</td>
<td>41.00</td>
<td>0.35</td>
</tr>
<tr>
<td></td>
<td>person of color</td>
<td>97.00</td>
<td>0.32</td>
</tr>
</tbody>
</table>

### fifteen_raises

```python
definition = (reason_for_change_combined['merit_raises'] == True & (reason_for_change_combined['dept'] == 'Commercial') &
(reason_for_change_combined['pay_rate_type'] == 'Salaried') &
(reason_for_change_combined['raise_after'] == '2015']).
groupby(['gender', 'race_grouping']).agg({'base_pay_change': [np.count_nonzero, np.median]},{'2015_annual_performance_rating': [np.count_nonzero, np.median]})
suppress(fifteen_raises)
```

### count_nonzero median

<table>
<thead>
<tr>
<th>gender</th>
<th>race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>white</td>
<td>7.00</td>
<td>937.13</td>
</tr>
<tr>
<td>Male</td>
<td>white</td>
<td>5.00</td>
<td>850.75</td>
</tr>
</tbody>
</table>
```python
fifteen_raises =
(reason_for_change_combined['merit_raises'] == True) &
(reason_for_change_combined['dept'] == 'Commercial') &
(reason_for_change_combined['pay_rate_type'] == 'Salaried') &
(reason_for_change_combined['raise_after'] == '2015').

groupby(['gender', 'race_grouping']).agg({'2015_annual_performance_rating':
[np.count_nonzero, np.median]})
suppress(fifteen_raises)

<table>
<thead>
<tr>
<th>gender</th>
<th>race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>white</td>
<td>7.00</td>
<td>3.50</td>
</tr>
<tr>
<td>Male</td>
<td>white</td>
<td>5.00</td>
<td>3.50</td>
</tr>
</tbody>
</table>

sixteen_raises =
(reason_for_change_combined['merit_raises'] == True) &
(reason_for_change_combined['dept'] == 'Commercial') &
(reason_for_change_combined['pay_rate_type'] == 'Salaried') &
(reason_for_change_combined['raise_after'] == '2016').

groupby(['gender', 'race_grouping']).agg({'2016_annual_performance_rating':
[np.count_nonzero, np.median],
'base_pay_change':
[np.count_nonzero, np.median]})
suppress(sixteen_raises)

| gender              | race_grouping       | count_nonzero | median |
|                     |                     |               |        |
| Female, person of color | white               | 5.00          | 1729.40|
|                      | white               | 9.00          | 1683.00|
| Male, person of color | white               | 6.00          | 1506.78|
|                      | white               | 7.00          | 1291.29|

sixteen_raises =
(reason_for_change_combined['merit_raises'] == True) &
(reason_for_change_combined['dept'] == 'Commercial') &
(reason_for_change_combined['pay_rate_type'] == 'Salaried') &
(reason_for_change_combined['raise_after'] == '2016').

groupby(['gender', 'race_grouping']).agg({'2016_annual_performance_rating':
[np.count_nonzero, np.median]})
suppress(sixteen_raises)

| gender              | race_grouping       | count_nonzero | median |
|                     |                     |               |        |
| Female, person of color | white               | 5.00          | 3.50   |
|                      | white               | 9.00          | 3.40   |
| Male, person of color | white               | 6.00          | 3.25   |
|                      | white               | 7.00          | 3.20   |
```

242
seventeen_raises =
(reason_for_change_combined['merit_raises'] == True) & (reason_for_change_combined['dept'] == 'Commercial') & (reason_for_change_combined['pay_rate_type'] == 'Salaried') & (reason_for_change_combined['raise_after'] == '2017').
groupby(['gender', 'race_grouping']).agg({
    'base_pay_change': [np.count_nonzero, np.median],
    '2017_annual_performance_rating': [np.count_nonzero, np.median]
}).
suppress(seventeen_raises)

#### 2017

<table>
<thead>
<tr>
<th></th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Female</strong></td>
<td>13.00</td>
<td>1398.48</td>
</tr>
<tr>
<td>white</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>Male</strong></td>
<td>8.00</td>
<td>1000.00</td>
</tr>
<tr>
<td>person of color</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>white</strong></td>
<td>5.00</td>
<td>1414.60</td>
</tr>
</tbody>
</table>

#### 2018

<table>
<thead>
<tr>
<th></th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Female</strong></td>
<td>7.00</td>
<td>1415.60</td>
</tr>
<tr>
<td>person of color</td>
<td></td>
<td></td>
</tr>
<tr>
<td>white</td>
<td>21.00</td>
<td>1668.88</td>
</tr>
<tr>
<td><strong>Male</strong></td>
<td>8.00</td>
<td>1417.48</td>
</tr>
<tr>
<td>person of color</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>white</strong></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

eighteen_raises =
(reason_for_change_combined['merit_raises'] == True) & (reason_for_change_combined['dept'] == 'Commercial') & (reason_for_change_combined['pay_rate_type'] == 'Salaried') & (reason_for_change_combined['raise_after'] == '2018').
groupby(['gender', 'race_grouping']).agg({
    'base_pay_change': [np.count_nonzero, np.median],
    '2018_annual_performance_rating': [np.count_nonzero, np.median]
}).
suppress(eighteen_raises)

#### 2018

<table>
<thead>
<tr>
<th></th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Female</strong></td>
<td>7.00</td>
<td>1415.60</td>
</tr>
<tr>
<td>person of color</td>
<td></td>
<td></td>
</tr>
<tr>
<td>white</td>
<td>21.00</td>
<td>1668.88</td>
</tr>
<tr>
<td><strong>Male</strong></td>
<td>7.00</td>
<td>1050.00</td>
</tr>
<tr>
<td>person of color</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>white</strong></td>
<td>8.00</td>
<td>1417.48</td>
</tr>
</tbody>
</table>

243
eighteen_raises =
  ~reason_for_change_combined[(reason_for_change_combined['merit_raises'] == True) & (reason_for_change_combined['dept'] == 'Commercial') &(~reason_for_change_combined['pay_rate_type'] == 'Salaried') & (~reason_for_change_combined['raise_after'] == '2018').groupby(['gender', 'race_grouping']).agg({2018_annual_performance_rating: np.count_nonzero, np.median})]
suppress(eighteen_raises)

<table>
<thead>
<tr>
<th>gender race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female person of color</td>
<td>7.00</td>
<td>3.40</td>
</tr>
<tr>
<td>white</td>
<td>21.00</td>
<td>3.50</td>
</tr>
<tr>
<td>Male person of color</td>
<td>7.00</td>
<td>3.30</td>
</tr>
<tr>
<td>white</td>
<td>8.00</td>
<td>3.50</td>
</tr>
</tbody>
</table>

merit_raises_15 =
  ~reason_for_change_combined[(reason_for_change_combined['raise_after'] == '2015') & (reason_for_change_combined['merit_raises'] == True)]
merit_raises_16 =
  ~reason_for_change_combined[(reason_for_change_combined['raise_after'] == '2016') & (reason_for_change_combined['merit_raises'] == True)]
merit_raises_17 =
  ~reason_for_change_combined[(reason_for_change_combined['raise_after'] == '2017') & (reason_for_change_combined['merit_raises'] == True)]
merit_raises_18 =
  ~reason_for_change_combined[(reason_for_change_combined['raise_after'] == '2018') & (reason_for_change_combined['merit_raises'] == True)]

merit_raises_15[['base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping']
  rename(columns={2015_annual_performance_rating: 'performance_rating'})
merit_raises_16[['base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping']
  rename(columns={2016_annual_performance_rating: 'performance_rating'})
merit_raises_17[['base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping']
  rename(columns={2017_annual_performance_rating: 'performance_rating'})
merit_raises_18[['base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping']
  rename(columns={2018_annual_performance_rating: 'performance_rating'})

merit_raises_15 = pd.DataFrame(merit_raises_15)
merit_raises_16 = pd.DataFrame(merit_raises_16)
merit_raises_17 = pd.DataFrame(merit_raises_17)
merit_raises_18 = pd.DataFrame(merit_raises_18)

244
merit_raises_combined = pd.concat([merit_raises_15, merit_raises_16, merit_raises_17, merit_raises_18])

commercial_salaried_raises = merit_raises_combined[merit_raises_combined['pay_rate_type'] == 'Salaried'].groupby(['gender', 'race_grouping']).agg({'base_pay_change': [np.count_nonzero, np.median]})
suppress(commercial_salaried_raises)

<table>
<thead>
<tr>
<th>gender</th>
<th>race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>person of color</td>
<td>116.00</td>
<td>2812.50</td>
</tr>
<tr>
<td></td>
<td>unknown</td>
<td>10.00</td>
<td>2860.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>317.00</td>
<td>2500.00</td>
</tr>
<tr>
<td>Male</td>
<td>person of color</td>
<td>102.00</td>
<td>2310.00</td>
</tr>
<tr>
<td></td>
<td>unknown</td>
<td>7.00</td>
<td>2500.00</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>379.00</td>
<td>3000.00</td>
</tr>
</tbody>
</table>

commercial_salaried_raises_scores = merit_raises_combined[merit_raises_combined['pay_rate_type'] == 'Salaried'].groupby(['gender', 'race_grouping']).agg({'base_pay_change': [np.count_nonzero, np.median]})
suppress(commercial_salaried_raises_scores)

<table>
<thead>
<tr>
<th>gender</th>
<th>race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>person of color</td>
<td>116.00</td>
<td>3.40</td>
</tr>
<tr>
<td></td>
<td>unknown</td>
<td>10.00</td>
<td>3.80</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>317.00</td>
<td>3.50</td>
</tr>
<tr>
<td>Male</td>
<td>person of color</td>
<td>102.00</td>
<td>3.40</td>
</tr>
<tr>
<td></td>
<td>unknown</td>
<td>7.00</td>
<td>3.70</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>379.00</td>
<td>3.60</td>
</tr>
</tbody>
</table>

commercial_hourly_raises = merit_raises_combined[merit_raises_combined['pay_rate_type'] == 'Hourly'].groupby(['gender', 'race_grouping']).agg({'base_pay_change': [np.count_nonzero, np.median]})
suppress(commercial_hourly_raises)

<table>
<thead>
<tr>
<th>gender</th>
<th>race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>person of color</td>
<td>120.00</td>
<td>0.43</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>88.00</td>
<td>0.78</td>
</tr>
<tr>
<td>Male</td>
<td>person of color</td>
<td>108.00</td>
<td>0.35</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>65.00</td>
<td>0.45</td>
</tr>
</tbody>
</table>

commercial_hourly_raises_scores = merit_raises_combined[merit_raises_combined['pay_rate_type'] == 'Hourly'].groupby(['gender', 'race_grouping']).agg({'base_pay_change': [np.count_nonzero, np.median]})

<table>
<thead>
<tr>
<th>gender</th>
<th>race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>person of color</td>
<td>120.00</td>
<td>0.43</td>
</tr>
<tr>
<td></td>
<td>white</td>
<td>88.00</td>
<td>0.78</td>
</tr>
</tbody>
</table>
1.6.11 Regression

```python
suppress(commercial_hourly_raises_scores)

<table>
<thead>
<tr>
<th>gender race_grouping</th>
<th>count_nonzero</th>
<th>median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female person of color</td>
<td>120.00</td>
<td>3.30</td>
</tr>
<tr>
<td>white</td>
<td>88.00</td>
<td>3.50</td>
</tr>
<tr>
<td>Male person of color</td>
<td>108.00</td>
<td>3.20</td>
</tr>
<tr>
<td>white</td>
<td>65.00</td>
<td>3.30</td>
</tr>
</tbody>
</table>
```

```python
commercial_salaried_regression =
commercial_salaried_regression = pd.get_dummies(commercial_salaried_regression, columns=['gender', 'race_ethnicity', 'age_group_5', 'years_of_service_grouped', 'dept', 'desk',
S
commercial_salaried_regression = commercial_salaried_regression
model41 = sm.ols(data=commercial_salaried_regression, formula='
current_base_pay ~ gender_Female + gender_Male')
result41 = model41.fit()
result41.summary()
```
Dep. Variable: current_base_pay  R-squared: 0.001
Model: OLS  Adj. R-squared: -0.007
Method: Least Squares  F-statistic: 0.07662
Date: Wed, 06 Nov 2019  Prob (F-statistic): 0.782
Time: 10:28:08  Log-Likelihood: -1577.9
Df Residuals: 131  BIC: 3166.
Df Model: 1
Covariance Type: nonrobust

==============================================================================
       coef     std err          t      P>|t|      [0.025      0.975]
-----------------------------------------------------------------------------
Intercept          6.382e+04    2093.898      30.480     0.000   5.97e+04    6.8e+04
6.8e+04
gender_Female     3.278e+04    3005.419      10.907     0.000   2.68e+04    3.87e+04
3.87e+04
gender_Male       3.104e+04    3590.196      8.646     0.000   2.39e+04    3.81e+04
3.81e+04

==============================================================================
Omnibus: 30.714  Durbin-Watson: 1.641
Prob(Omnibus): 0.000  Jarque-Bera (JB): 42.867
Kurtosis: 4.064  Cond. No. 3.62e+15

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 1.58e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

[452]: model42 = sm.ols(data=commercial_salaried_regression, formula = 'current_base_pay ~ race_grouping_white + race_grouping_person_of_color')
result42 = model42.fit()
result42.summary()

[452]: <class 'statsmodels.iolib.summary.Summary'>

OLS Regression Results
==============================================================================
Dep. Variable: current_base_pay  R-squared: 0.025
Model: OLS  Adj. R-squared: 0.010
Method: Least Squares  F-statistic: 1.645
Date: Wed, 06 Nov 2019  Prob (F-statistic): 0.197

247
Time: 10:28:09  Log-Likelihood: -1576.3
Df Residuals: 130  BIC: 3167.
Df Model: 2
Covariance Type: nonrobust

--------------------------------------------------------------------------------
coef   std err     t    P>|t|  [0.025  0.975]
--------------------------------------------------------------------------------
Intercept   7.84e+04   2.43e+04  3.229  0.002 3.04e+04 1.26e+05
race_grouping_white   2.068e+04   2.45e+04  0.843  0.401 -2.78e+04 6.92e+04
race_grouping_person_of_color  9089.4666  2.5e+04  0.363  0.717 -4.04e+04 5.86e+04

==============================================================================
Omnibus: 28.825  Durbin-Watson: 1.642
Prob(Omnibus): 0.000  Jarque-Bera (JB): 39.096
Skew: 1.238  Prob(JB): 3.24e-09
Kurtosis: 3.964  Cond. No. 18.2
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.

```python
model43 = sm.ols(data=commercial_salaried_regression, formula = 'current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color')
result43 = model43.fit()
result43.summary()
```

```
<class 'statsmodels.iolib.summary.Summary'>
```
| coef | std err | t     | P>|t| |
|------|---------|-------|------|
| 5.199e+04 | 1.64e+04 | 3.173 | 0.002 |
| 2.641e+04 | 8394.824 | 3.146 | 0.002 |
| 2.558e+04 | 9156.599 | 2.794 | 0.006 |
| 2.095e+04 | 2.47e+04 | 0.848 | 0.398 |
| 9479.6077 | 2.53e+04 | 0.375 | 0.709 |

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 1.36e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
new_commercial_salaried_regression = pd.DataFrame({'gender_Female': [1, 0, 1, 0], 'gender_Male': [0, 1, 0, 1], 'race_grouping_white': [1, 1, 0, 0], 'race_grouping_person_of_color': [0, 0, 1, 1], 'age': [40, 40, 40, 40]})
new_commercial_salaried_regression['predicted'] = result43.predict(new_commercial_salaried_regression)
new_commercial_salaried_regression
```

```python
<table>
<thead>
<tr>
<th>gender_Female</th>
<th>gender_Male</th>
<th>race_grouping_white</th>
<th>race_grouping_person_of_color</th>
<th>age</th>
<th>predicted</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>40</td>
<td>99356.99</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>40</td>
<td>98524.69</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>40</td>
<td>87883.11</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>40</td>
<td>87050.81</td>
</tr>
</tbody>
</table>
```
```python
model44 = sm.ols(data=commercial_salaried_regression, formula='current_base_pay ~ gender_Female + gender_Male + age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over')
result44 = model44.fit()
result44.summary()
```

```plaintext
OLS Regression Results
==============================================================================
Dep. Variable:  current_base_pay   R-squared:  0.286
Model:              OLS     Adj. R-squared:  0.227
Method:             Least Squares   F-statistic:  4.882
Date:             Wed, 06 Nov 2019   Prob (F-statistic):  6.47e-06
Df Residuals:          122   BIC:  3165.
Df Model:              10
Covariance Type:      nonrobust
==============================================================================
                  coef    std err          t      P>|t|      [0.025    0.975]
------------------------------------------------------------------------------
  Intercept         6.157e+04    2123.471     28.997      0.000     5.74e+04    6.58e+04
  gender_Female    3.556e+04    3023.460     11.762      0.000     2.96e+04    4.15e+04
  gender_Male      2.601e+04    3274.551      7.944      0.000     1.95e+04    3.25e+04
age_group_5_25_under -3.072e+04    9309.126     -3.300      0.001    -4.91e+04    -1.23e+04
age_group_5_25to29 -1.766e+04    5809.577     -3.040      0.003    -2.92e+04   -6162.194
age_group_5_30to34   2.149e+04    7531.035      2.853      0.005    6579.270    8540.680
age_group_5_35to39   2.277e+04    7189.104      3.168      0.002    8540.680    1.7e+04
age_group_5_40to44   2.951e+04    9833.731      3.001      0.003     1e+04     4.9e+04
age_group_5_45to49  9655.6318    8217.596      1.175      0.242    -6611.919    2.59e+04
age_group_5_50to54 -1292.9123    9239.767     -0.140      0.889     -1.96e+04    1.7e+04
```

The model evaluates the relationship between current base pay and various demographic factors including gender and age groups, indicating a significant variation in base pay across different categories.
```python
[456]: model45 = sm.ols(data=commercial_salaried_regression, formula='current_base_pay ~ race_grouping_white + race_grouping_person_of_color +
                        age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 +
                        age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
                        age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
                        age_group_5_65_over')
result45 = model45.fit()
result45.summary()
```

```
Omnibus: 14.188  Durbin-Watson: 1.771
Prob(Omnibus): 0.001  Jarque-Bera (JB): 15.735
Skew: 0.720  Prob(JB): 0.000383
Kurtosis: 3.874  Cond. No. 9.78e+15
```

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 2.37e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```
```
Intercept 5.016e+04 1.97e+04 2.553 0.012
race_grouping_white 4.933e+04 2.18e+04 2.264 0.025
race_grouping_person_of_color 3.255e+04 2.23e+04 1.462 0.146
age_group_5_25_under -3.33e+04 9266.922 -3.594 0.000
age_group_5_25to29 -1.83e+04 5876.870 -3.114 0.002
age_group_5_30to34 2.118e+04 7351.305 2.882 0.005
age_group_5_35to39 2.03e+04 7310.811 2.777 0.006
age_group_5_40to44 3.53e+04 9345.043 3.778 0.000
age_group_5_45to49 1.064e+04 8367.434 1.271 0.206
age_group_5_50to54 -4834.2752 9266.922 -0.522 0.603
age_group_5_55to59 -1681.1728 9219.577 -0.182 0.856
age_group_5_60to64 771.0941 9248.687 0.083 0.934
age_group_5_65_over 2.007e+04 1.92e+04 1.044 0.298

Omnibus: 10.496 Durbin-Watson: 1.847
Prob(Omnibus): 0.005 Jarque-Bera (JB): 10.654
Skew: 0.624 Prob(JB): 0.00486
Kurtosis: 3.606 Cond. No. 8.32e+15

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 3.44e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
model46 = sm.ols(data=commercial_salaried_regression, formula = 'current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over')
result46 = model46.fit()
```
result46.summary()

```
[457]: <class 'statsmodels.iolib.summary.Summary'>
""

OLS Regression Results
==============================================================================
Dep. Variable: current_base_pay  R-squared:                       0.350
Model: OLS                  Adj. R-squared:                   0.285
Method: Least Squares       F-statistic:                     5.377
Date: Wed, 06 Nov 2019     Prob (F-statistic):           3.10e-07
Time: 10:28:09             Log-Likelihood:               -1549.3
Df Residuals:             120                           BIC:             3162.
Df Model:                12
Covariance Type:         nonrobust
==============================================================================
                        coef     std err          t      P>|t|      [0.025    0.975]
-------------------------------------------------------------
Intercept                           3.231e+04   1.35e+04       2.396      0.018    5616.362  5616.362
gender_Female                       2.084e+04   7063.061       2.950      0.004   6853.164  3.48e+04
gender_Male                         1.148e+04   7585.687       1.513      0.133  -3541.955 2.65e+04
race_grouping_white                 5.196e+04   2.17e+04       2.394      0.018  8994.931 9.49e+04
race_grouping_person_of_color      3.599e+04   2.22e+04       1.620      0.108  -7990.410 8e+04
age_group_5_25_under                 1.709e+04   8653.679       1.944      0.054 -2.50e+04 1.03e+04
age_group_5_25to29                  1.967e+04   7265.155       2.707      0.008  -7729.468 2.51e+04
age_group_5_30to34                  1.914e+04   7117.210       2.689      0.008 -2.785.8549 9032.997
age_group_5_35to39                  1.971e+04   7117.210       2.689      0.008 -2.785.8549 9032.997
age_group_5_40to44                  1.921e+04   9477.598       2.039      0.042 -2.785.8549 9032.997
age_group_5_45to49                 -7224.019   2.51e+04      -2.888      0.004  -2.785.8549 9032.997
age_group_5_50to54                 -7729.468   9113.273      -0.848      0.398 -2.785.8549 9032.997
age_group_5_55to59                 -2.07e+04   1.51e+04      -1.375      0.170 -2.785.8549 9032.997
age_group_5_60to64                 -7224.019   2.51e+04      -2.888      0.004  -2.785.8549 9032.997
age_group_5_65to69                 -7224.019   2.51e+04      -2.888      0.004  -2.785.8549 9032.997
age_group_5_70to74                 -7224.019   2.51e+04      -2.888      0.004  -2.785.8549 9032.997
age_group_5_75to79                 -7224.019   2.51e+04      -2.888      0.004  -2.785.8549 9032.997
age_group_5_80to84                 -7224.019   2.51e+04      -2.888      0.004  -2.785.8549 9032.997
age_group_5_85to89                 -7224.019   2.51e+04      -2.888      0.004  -2.785.8549 9032.997
age_group_5_90to94                 -7224.019   2.51e+04      -2.888      0.004  -2.785.8549 9032.997
age_group_5_95to99                 -7224.019   2.51e+04      -2.888      0.004  -2.785.8549 9032.997
age_group_5_100to104               -7224.019   2.51e+04      -2.888      0.004  -2.785.8549 9032.997
age_group_6_25_under                1.921e+04   9477.598       2.039      0.042 -2.785.8549 9032.997
age_group_6_25to29                 1.971e+04   7117.210       2.689      0.008 -2.785.8549 9032.997
age_group_6_30to34                 1.971e+04   7117.210       2.689      0.008 -2.785.8549 9032.997
age_group_6_35to39                 1.971e+04   7117.210       2.689      0.008 -2.785.8549 9032.997
age_group_6_40to44                 1.971e+04   7117.210       2.689      0.008 -2.785.8549 9032.997
age_group_6_45to49                 1.971e+04   7117.210       2.689      0.008 -2.785.8549 9032.997
age_group_6_50to54                 1.971e+04   7117.210       2.689      0.008 -2.785.8549 9032.997
age_group_6_55to59                 1.971e+04   7117.210       2.689      0.008 -2.785.8549 9032.997
age_group_6_60to64                 1.971e+04   7117.210       2.689      0.008 -2.785.8549 9032.997
age_group_6_65to69                 1.971e+04   7117.210       2.689      0.008 -2.785.8549 9032.997
age_group_6_70to74                 1.971e+04   7117.210       2.689      0.008 -2.785.8549 9032.997
age_group_6_75to79                 1.971e+04   7117.210       2.689      0.008 -2.785.8549 9032.997
age_group_6_80to84                 1.971e+04   7117.210       2.689      0.008 -2.785.8549 9032.997
age_group_6_85to89                 1.971e+04   7117.210       2.689      0.008 -2.785.8549 9032.997
age_group_6_90to94                 1.971e+04   7117.210       2.689      0.008 -2.785.8549 9032.997
age_group_6_95to99                 1.971e+04   7117.210       2.689      0.008 -2.785.8549 9032.997
age_group_6_100to104               1.971e+04   7117.210       2.689      0.008 -2.785.8549 9032.997
```
```python
merit_raises_combined_salaried_regression = merit_raises_combined[(merit_raises_combined['dept'] == 'Commercial') & (merit_raises_combined['pay_rate_type'] == 'Salaried')]
merit_raises_combined_salaried_regression = pd.get_dummies(merit_raises_combined_salaried_regression, columns=['gender', 'race_grouping', 'age_group_5'])
```

```
merit_raises_combined_salaried_regression = merit_raises_combined_salaried_regression.rename(columns={'race_grouping_person of color': 'race_grouping_person_of_color', 'age_group_5_60to64': 'age_group_5_60to64', 'age_group_5_65_over': 'age_group_5_65_over'}
model47 = sm.ols(data=merit_raises_combined_salaried_regression, formula='base_pay_change ~ gender_Female + gender_Male')
result47 = model47.fit()
result47.summary()
```

```
OLS Regression Results
==============================================================================
Dep. Variable:    base_pay_change    R-squared:     0.022
==============================================================================
```

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 1.89e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```
Model:       OLS  Adj. R-squared:   0.014
Method:      Least Squares  F-statistic:  2.664
Date:        Wed, 06 Nov 2019  Prob (F-statistic):  0.105
Time:        10:28:09  Log-Likelihood: -999.84
Df Model:      1  
Covariance Type:  nonrobust
==============================================================================
                 coef   std err      t      P>|t|     [0.025    0.975]
==============================================================================
        Intercept    1002.1929    62.763    15.968    0.000      877.905       1126.480
        gender_Female   654.7617    93.620     6.994    0.000       469.369       840.154
        gender_Male    347.4312   104.552     3.323    0.001       140.389       554.473
==============================================================================
Omnibus:       63.911  Durbin-Watson:  1.948
Prob(Omnibus): 0.000  Jarque-Bera (JB): 203.112
Skew:          2.035  Prob(JB): 7.85e-45
Kurtosis:      7.905  Cond. No.  3.59e+15
==============================================================================
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 1.42e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
[460]: model48 = sm.ols(data=merit_raises_combined_salaried_regression, formula='base_pay_change ~ race_grouping_white + race_grouping_person_of_color')
result48 = model48.fit()
result48.summary()
```

```
OLS Regression Results
==============================================================================
Dep. Variable:    base_pay_change  R-squared:   0.005
Model:             OLS  Adj. R-squared:   -0.012
Method:            Least Squares  F-statistic:  0.3188
Date:              Wed, 06 Nov 2019  Prob (F-statistic):  0.728
Time:              10:28:09  Log-Likelihood: -1000.9
```
Df Model: 2  
Covariance Type: nonrobust

==============================================================================
                      coef    std err          t      P>|t|      [0.025   0.975]
------------------------------------------------------------------------------
         Intercept      1400.000  1026.778      1.363      0.175    -633.479  3433.479
         race_grouping_white      189.377  1033.600      0.183      0.855  -1857.613  2236.368
   race_grouping_person_of_color      35.728  1038.380      0.034      0.973  -2020.729  2092.186
==============================================================================
Omnibus: 66.033 Durbin-Watson: 1.921
Prob(Omnibus): 0.000 Jarque-Bera (JB): 218.590
Skew: 2.092 Prob(JB): 3.42e-48
Kurtosis: 8.120 Cond. No. 23.6

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

[461]: model49 = sm.ols(data=merit_raises_combined_salaried_regression, formula = 'base_pay_change ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color')
result49 = model49.fit()
result49.summary()
|                  | coef   | std err  | t   | P>|t| |
|------------------|--------|----------|-----|-----|
| Intercept        | 835.9007 | 683.945  | 1.222 | 0.224 |
| -518.738         | 2190.540 |
| gender_Female    | 564.0993 | 346.551  | 1.628 | 0.106 |
| -122.288         | 1250.486 |
| gender_Male      | 271.8013 | 364.288  | 0.746 | 0.457 |
| -449.716         | 993.319  |
| race_grouping_white | 286.8101 | 1030.126 | 0.278 | 0.781 |
| -1753.485        | 2327.105 |
| race_grouping_person_of_color | 195.1637 | 1038.272 | 0.188 | 0.851 |
| -1861.266        | 2251.593 |

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 1.41e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```
[462]: new_reason_for_change_combined_regression = pd.DataFrame({'gender_Female': [1,0,1,0],
                                                                  'gender_Male': [0,1,0,1],
                                                                  'race_grouping_white': [1,1,0,0],
                                                                  'race_grouping_person_of_color': [0,0,1,1]})
new_reason_for_change_combined_regression['predicted'] = result49.
new_reason_for_change_combined_regression.predict(new_reason_for_change_combined_regression)
```

```
   gender_Female  gender_Male  race_grouping_white  \
0       1          0           0               1
1       0          1           1
2       1          0           0
3       0          1           0
```

```
   race_grouping_person_of_color  predicted
0       0                1686.81
1       0                1394.51
2       1                1595.16
3       1                1302.87
```
model50 = sm.ols(data=merit_raises_combined_salaried_regression, formula =
~'base_pay_change ~ gender_Female + gender_Male + age_group_5_25_under +
~age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 +
~age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 +
~age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over')
result50 = model50.fit()
result50.summary()
---

Intercept 950.6651 89.652 10.604 0.000 772.996 1128.334
gender_Female 634.1486 117.932 5.377 0.000 400.435 867.862
gender_Male 316.5165 127.637 2.480 0.015 63.570 569.463
age_group_5_25_under 48.4184 912.882 0.053 0.958 -1760.699 1857.536
age_group_5_25to29 253.1740 238.716 1.061 0.291 -219.905 726.253
age_group_5_30to34 -206.1156 256.682 -0.803 0.424 -714.800 302.568
age_group_5_35to39 477.0562 269.916 1.767 0.080 -57.855 1011.967
age_group_5_40to44 710.8937 356.916 1.991 0.049 3.470 1418.318
age_group_5_45to49 -108.7731 226.626 -0.480 0.632 -557.893 340.347
age_group_5_50to54 -185.8002 285.282 -0.651 0.516 -751.162 379.562
age_group_5_55to59 134.3662 400.874 0.335 0.738 -660.073 928.805
age_group_5_60to64 -172.5545 304.690 -0.566 0.572 -776.378 431.269
age_group_5_65_over 0 0 nan nan 0 0

Omnibus: 49.406 Durbin-Watson: 2.042
Prob(Omnibus): 0.000 Jarque-Bera (JB): 118.315
Skew: 1.654 Prob(JB): 2.03e-26
Kurtosis: 6.566 Cond. No. inf

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 0. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

[464]: model51 = sm.ols(data=meritraises.combined_salaried_regression, formula = 'base_pay_change ~ race_grouping_white + race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over')

259
```python
result51 = model51.fit()
result51.summary()
```

```
[464]: <class 'statsmodels.iolib.summary.Summary'>
```

```
### OLS Regression Results

<table>
<thead>
<tr>
<th>Dep. Variable:</th>
<th>base_pay_change</th>
<th>R-squared:</th>
<th>0.103</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model:</td>
<td>OLS</td>
<td>Adj. R-squared:</td>
<td>0.021</td>
</tr>
<tr>
<td>Method:</td>
<td>Least Squares</td>
<td>F-statistic:</td>
<td>1.250</td>
</tr>
<tr>
<td>Date:</td>
<td>Wed, 06 Nov 2019</td>
<td>Prob (F-statistic):</td>
<td>0.268</td>
</tr>
<tr>
<td>Time:</td>
<td>10:28:10</td>
<td>Log-Likelihood:</td>
<td>-994.67</td>
</tr>
<tr>
<td>Df Model:</td>
<td>10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Covariance Type:</td>
<td>nonrobust</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

```

```
|             | coef | std err | t    | P>|t| |
|-------------|------|---------|------|------|
| Intercept   | 1096.4061 | 936.124 | 1.171 | 0.244 |
| -758.962    | 2951.774   |         |     |     |
| race_grouping_white | 412.0134 | 1036.127 | 0.398 | 0.692 |
| -1641.556   | 2465.583   |         |     |     |
| race_grouping_person_of_color | 174.9337 | 1048.709 | 0.167 | 0.868 |
| -1903.573   | 2253.441   |         |     |     |
| age_group_5_25_under | -192.8195 | 921.509 | -0.209 | 0.835 |
| -2019.220   | 1633.581   |         |     |     |
| age_group_5_25to29 | 303.5939 | 244.359 | 1.242 | 0.217 |
| -180.717    | 787.905    |         |     |     |
| age_group_5_30to34 | -103.6335 | 272.182 | -0.381 | 0.704 |
| -643.090    | 435.823    |         |     |     |
| age_group_5_35to39 | 445.2756 | 286.369 | 1.555 | 0.123 |
| -122.299    | 1012.850   |         |     |     |
| age_group_5_40to44 | 876.1929 | 361.933 | 2.421 | 0.017 |
| 158.854     | 1593.532   |         |     |     |
| age_group_5_45to49 | -52.8984 | 257.357 | -0.206 | 0.838 |
| -562.972    | 467.175    |         |     |     |
| age_group_5_50to54 | -127.9950 | 302.789 | -0.423 | 0.673 |
| -728.112    | 472.122    |         |     |     |
| age_group_5_55to59 | 289.7871 | 405.978 | 0.714 | 0.477 |
| -514.849    | 1094.423   |         |     |     |
| age_group_5_60to64 | -341.0970 | 314.230 | -1.086 | 0.280 |
| -963.891    | 281.697    |         |     |     |
| age_group_5_65_over | 0 | 0 | nan | nan |
```

260
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 0. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
model52 = sm.ols(data=merit_raises_combined_salaried_regression, formula = 'base_pay_change ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over')
result52 = model52.fit()
result52.summary()
```

### OLS Regression Results

```
OLS Regression Results
```

```
Dep. Variable:     base_pay_change   R-squared:     0.115
Model:                OLS           Adj. R-squared:  0.025
Method:              Least Squares   F-statistic:   1.273
Date:       Wed, 06 Nov 2019   Prob (F-statistic):      0.250
Time:       10:28:10          Log-Likelihood:  -993.87
Df Model:                11
Covariance Type:            nonrobust

==============================================================================
                 coef    std err          t      P>|t|    [0.025    0.975]
------------------------------------------------------------------------------
     Intercept   687.2307     646.791     1.063    0.290     -594.822    1969.284
gender_Female   485.3983     334.200     1.452    0.149     -177.044    1147.841
gender_Male    201.8324     353.734     0.571    0.569        72.846    350.919
```

```
| Race Grouping                  | Coef.   | Std. Err. | t-value | Pr(>|t|) |
|-------------------------------|---------|-----------|---------|---------|
| White                         | 486.8437| 1035.835  | 0.470   | 0.639   |
| Person of Color               | 300.9616| 1051.736  | 0.286   | 0.775   |
| Age Group 5-25 Under          | -60.3067| 924.108   | -0.065  | 0.948   |
| Age Group 5-25 to 29          | 227.3710| 242.088   | 0.939   | 0.350   |
| Age Group 5-30 to 34          | -224.4915| 270.950   | -0.829  | 0.409   |
| Age Group 5-35 to 39          | 494.6194| 285.778   | 1.731   | 0.086   |
| Age Group 5-40 to 44          | 705.9406| 370.014   | 1.908   | 0.059   |
| Age Group 5-45 to 49          | -92.0339| 246.733   | -0.373  | 0.710   |
| Age Group 5-50 to 54          | -225.4246| 297.524   | -0.758  | 0.450   |
| Age Group 5-55 to 59          | 121.6681| 412.223   | 0.295   | 0.768   |
| Age Group 5-60 to 64          | -260.1117| 320.639   | -0.811  | 0.419   |
| Age Group 5-65+               | 0       | 0         | nan     | nan     |

---

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 0. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
[466]: model53 = sm.ols(data=merit_raises_combined_salaried_regression, formula = 'performance_rating ~ gender_Female + gender_Male')
result53 = model53.fit()
result53.summary()
```

---

OLS Regression Results
----------------------------------------
```python
model54 = sm.ols(data=merit_raises_combined_salaried_regression, formula='performance_rating ~ race_grouping_white + race_grouping_person_of_color')
result54 = model54.fit()
result54.summary()
```

```bash
Dep. Variable:    performance_rating  R-squared:       0.001  
Model:                      OLS  Adj. R-squared:     -0.016  
Method:                Least Squares  F-statistic:       0.7373  
Date:             Wed, 06 Nov 2019  Prob (F-statistic):    0.392  
No. Observations:     118  AIC:              67.10  
Df Residuals:         116  BIC:              72.64  
Df Model:              1  
Covariance Type:  nonrobust  

==============================================================================
           coef     std err          t      P>|t|      [0.025  0.975]
  Intercept    2.2810     0.020    114.520      0.000     2.242   2.320
  gender_Female   1.1662     0.030     39.292      0.000     1.107   1.225
  gender_Male    1.1148     0.033     33.572      0.000     1.049   1.181

Omni:            5.156  Durbin-Watson:    1.775  
Prob(Omni):     0.076  Jarque-Bera (JB):    5.147  
Skew:             0.509  Prob(JB):        0.0763  
Kurtosis:       2.899  Cond. No.        8.49e+15  
```

**Warnings:**

[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

[2] The smallest eigenvalue is 2.5e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

---

### Code Snippet

```python
model54 = sm.ols(data=merit_raises_combined_salaried_regression, formula='performance_rating ~ race_grouping_white + race_grouping_person_of_color')
result54 = model54.fit()
result54.summary()
```
No. Observations: 118  AIC: 69.69
Df Residuals: 115  BIC: 78.00
Df Model: 2
Covariance Type: nonrobust

==============================================================================
                      coef    std err          t      P>|t|      [0.025     0.975]
==============================================================================
Intercept            3.4000      0.321     10.591      0.000     2.764      4.036
race_grouping_white   0.0351      0.323      0.109      0.914    -0.605     0.675
race_grouping_person_of_color  0.0116      0.325      0.036      0.971    -0.632     0.655

==============================================================================
Omnibus: 5.821  Durbin-Watson: 1.789
Prob(Omnibus): 0.054  Jarque-Bera (JB): 5.895
Skew: 0.544  Prob(JB): 0.0525
Kurtosis: 2.885  Cond. No. 23.4

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.

[468]: model55 = sm.ols(data=merit_raises_combined_salaried_regression, formula = 'performance_rating ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color')
result55 = model55.fit()
result55.summary()

[468]: <class 'statsmodels.iolib.summary.Summary'>

==============================================================================
 OLS Regression Results
==============================================================================
 Dep. Variable: performance_rating  R-squared: 0.007
 Model: OLS  Adj. R-squared: -0.019
 Method: Least Squares  F-statistic: 0.2609
 Date: Wed, 06 Nov 2019  Prob (F-statistic): 0.853
 No. Observations: 118  AIC: 71.04
 Df Residuals: 114  BIC: 82.12
 Df Model: 3
 Covariance Type: nonrobust
|                     | coef  | std err | t      | P>|t| |
|---------------------|-------|---------|--------|-----|
| Intercept           | 2.2502| 0.215   | 10.448 | 0.000|
| gender_Female       | 1.1498| 0.109   | 10.532 | 0.000|
| gender_Male         | 1.1005| 0.115   | 9.577  | 0.000|
| race_grouping_white | 0.0511| 0.324   | 0.158  | 0.875|
| race_grouping_person_of_color | 0.0391 | 0.327 | 0.120 | 0.905|

Omnibus: 5.075 Durbin-Watson: 1.776
Prob(Omnibus): 0.079 Jarque-Bera (JB): 5.057
Skew: 0.505 Prob(JB): 0.0798
Kurtosis: 2.901 Cond. No. 1.02e+16

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 2.32e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```
model56 = sm.ols(data=merit_raises_combined_salaried_regression, formula='performance_rating ~ gender_Female + gender_Male + age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over')
result56 = model56.fit()
result56.summary()
```

```
<class 'statsmodels.iolib.summary.Summary'>
```

```
OLS Regression Results
```

<table>
<thead>
<tr>
<th>Dep. Variable:</th>
<th>performance_rating</th>
<th>R-squared:</th>
<th>0.120</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model:</td>
<td>OLS</td>
<td>Adj. R-squared:</td>
<td>0.046</td>
</tr>
<tr>
<td>Method:</td>
<td>Least Squares</td>
<td>F-statistic:</td>
<td>1.629</td>
</tr>
<tr>
<td>Date:</td>
<td>Wed, 06 Nov 2019</td>
<td>Prob (F-statistic):</td>
<td>0.116</td>
</tr>
<tr>
<td>No. Observations:</td>
<td>118</td>
<td>AIC:</td>
<td>68.83</td>
</tr>
</tbody>
</table>
Df Residuals: 108  BIC: 96.53
Df Model: 9
Covariance Type: nonrobust

==============================================================================
            coef    std err      t      P>|t|      [0.025      0.975]
---         ------  ------- ------      ------  --------  --------
 Intercept  2.1358    0.028  76.644      0.000     2.081     2.191
 gender_Female  1.0716    0.037  28.823      0.000     0.998     1.145
 gender_Male  1.0643    0.040  26.328      0.000     0.984     1.144
 age_group_5_25_under  0.0999    0.283   0.353    0.725    -0.461     0.661
 age_group_5_25to29  0.1698    0.075   2.260      0.026     0.021     0.319
 age_group_5_30to34  0.1758    0.082   2.141      0.035     0.013     0.339
 age_group_5_35to39  0.2692    0.084   3.205      0.002     0.103     0.436
 age_group_5_40to44  0.2676    0.111   2.415      0.017     0.048     0.487
 age_group_5_45to49  0.1212    0.070   1.724      0.088    -0.018     0.261
 age_group_5_50to54  0.4327    0.088   4.891      0.000     0.257     0.608
 age_group_5_55to59  0.4592    0.124   3.692      0.000     0.213     0.706
 age_group_5_60to64  0.1404    0.095   1.481      0.141    -0.047     0.328
 age_group_5_65_over  0.000    0.000   nan       nan    inf     nan

Omni: 8.372  Durbin-Watson: 1.928
Prob(Omnibus): 0.015  Jarque-Bera (JB): 8.098
Skew: 0.609  Prob(JB): 0.0174
Kurtosis: 3.404  Cond. No. inf

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 0. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.
model57 = sm.ols(data=merit_raises_combined_salaried_regression, formula='performance_rating ~ race_grouping_white + race_grouping_person_of_color +
age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 +
age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
age_group_5_50to64 + age_group_5_55to59 + age_group_5_60to64 +
age_group_5_65_over')
result57 = model57.fit()
result57.summary()

<class 'statsmodels.iolib.summary.Summary'>

```
OLS Regression Results
==============================================================================
Dep. Variable: performance_rating  R-squared:                       0.120
Model:                 OLS  Adj. R-squared:                  0.038
Method:               Least Squares  F-statistic:                1.463
Date:               Wed, 06 Nov 2019  Prob (F-statistic):         0.163
No. Observations:         118  AIC:                              70.72
Df Residuals:            107  BIC:                             101.2
Df Model:                 10
Covariance Type:        nonrobust
```

```
=================================================================================
                       coef    std err          t      P>|t|    95.0% Conf. Int.
---------------------- -------- -------- -------- -------- ------------------------
Intercept             3.1237    0.2900    10.7800      0.000      2.5489,     3.6987
race_grouping_white  -0.0231    0.3210    -0.0720      0.943     -0.6589,     0.6133
race_grouping_person_of_color  -0.0434    0.3250   -0.1330      0.894     -0.6879,     0.6011
age_group_5_25_under   0.1994    0.2853     0.7000      0.486     -0.3648,     0.7640
age_group_5_25to29    0.2763    0.0771     3.5930      0.000      0.1237,     0.4287
age_group_5_30to34    0.2889    0.0863     3.3670      0.001      0.1194,     0.4583
age_group_5_35to39    0.3831    0.0892     4.3130      0.000      0.2073,     0.5554
age_group_5_40to44    0.3820    0.1123     3.4110      0.001      0.1600,     0.6044
age_group_5_45to49    0.2362    0.0802     2.9590      0.004      0.0780,     0.3944
age_group_5_50to64    0.3570    0.1033     3.4560      0.001      0.1524,     0.5617
age_group_5_55to59    0.4240    0.1099     3.8630      0.000      0.2056,     0.6424
age_group_5_60to64    0.3784    0.1088     3.4760      0.001      0.1627,     0.5940
age_group_5_65_over   0.4521    0.1269     3.5740      0.000      0.2002,     0.7040
```
age_group_5_50to54  0.5425  0.094  5.792  0.000
0.357  0.728
age_group_5_55to59  0.5728  0.126  4.561  0.000
0.324  0.822
age_group_5_60to64  0.2427  0.097  2.497  0.014
0.050  0.435
age_group_5_65_over 0  0  nan  nan

Omnibus: 8.442 Durbin-Watson: 1.937
Prob(Omnibus): 0.015 Jarque-Bera (JB): 8.190
Skew: 0.616 Prob(JB): 0.0167
Kurtosis: 3.387 Cond. No. inf

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 0. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

[471]: model58 = sm.ols(data=merit_raises_combined_salaried_regression, formula='performance_rating ~ gender_Female + gender_Male + race_grouping_white +
race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 +
age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 +
age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 +
age_group_5_60to64 + age_group_5_65_over')
result58 = model58.fit()
result58.summary()

[471]: <class 'statsmodels.iolib.summary.Summary'>

OLS Regression Results
==============================================================================
Dep. Variable:    performance_rating     R-squared:           0.120
Model:              OLS                  Adj. R-squared:      0.029
Method:             Least Squares        F-statistic:        1.318
Date:               Wed, 06 Nov 2019     Prob (F-statistic): 0.225
No. Observations:   118                  AIC:                72.72
Df Residuals:       106                  BIC:                106.0
Df Model:           11
Covariance Type:    nonrobust
==============================================================================
                 coef    std err          t      P>|t|    [0.025  0.975]
------------------------------------------------------------------------------

268
---

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>2.1538</td>
<td>0.202</td>
<td>10.675</td>
<td>0.000</td>
</tr>
<tr>
<td>gender_Female</td>
<td>1.0778</td>
<td>0.104</td>
<td>10.346</td>
<td>0.000</td>
</tr>
<tr>
<td>gender_Male</td>
<td>1.0760</td>
<td>0.111</td>
<td>9.684</td>
<td>0.000</td>
</tr>
<tr>
<td>race_grouping_white</td>
<td>-0.0226</td>
<td>0.323</td>
<td>-0.070</td>
<td>0.944</td>
</tr>
<tr>
<td>race_grouping_person_of_color</td>
<td>-0.0425</td>
<td>0.329</td>
<td>-0.129</td>
<td>0.897</td>
</tr>
<tr>
<td>age_group_5_25_under</td>
<td>0.0927</td>
<td>0.288</td>
<td>0.322</td>
<td>0.748</td>
</tr>
<tr>
<td>age_group_5_25to29</td>
<td>0.1683</td>
<td>0.077</td>
<td>2.198</td>
<td>0.030</td>
</tr>
<tr>
<td>age_group_5_30to34</td>
<td>0.1805</td>
<td>0.087</td>
<td>2.070</td>
<td>0.041</td>
</tr>
<tr>
<td>age_group_5_35to39</td>
<td>0.2759</td>
<td>0.089</td>
<td>3.093</td>
<td>0.003</td>
</tr>
<tr>
<td>age_group_5_40to44</td>
<td>0.2734</td>
<td>0.116</td>
<td>2.366</td>
<td>0.020</td>
</tr>
<tr>
<td>age_group_5_45to49</td>
<td>0.1285</td>
<td>0.077</td>
<td>1.667</td>
<td>0.098</td>
</tr>
<tr>
<td>age_group_5_50to54</td>
<td>0.4344</td>
<td>0.093</td>
<td>4.687</td>
<td>0.000</td>
</tr>
<tr>
<td>age_group_5_55to59</td>
<td>0.4643</td>
<td>0.129</td>
<td>3.609</td>
<td>0.000</td>
</tr>
<tr>
<td>age_group_5_60to64</td>
<td>0.1358</td>
<td>0.100</td>
<td>1.353</td>
<td>0.179</td>
</tr>
<tr>
<td>age_group_5_65_over</td>
<td>0</td>
<td>0</td>
<td>nan</td>
<td>nan</td>
</tr>
<tr>
<td>age_group_5_65_over</td>
<td>0</td>
<td>0</td>
<td>nan</td>
<td>nan</td>
</tr>
</tbody>
</table>

---

Omnibus: 8.393  Durbin-Watson: 1.936  
Prob(Omnibus): 0.015  Jarque-Bera (JB): 8.134  
Skew: 0.614  Prob(JB): 0.0171 
Kurtosis: 3.385  Cond. No. inf 

---

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 0. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

---

269
```python
commercial_hourly_regression =
commercial_hourly_regression = pd.get_dummies(commercial_hourly_regression,
                                                 columns=['gender', 'race_ethnicity', 'age_group_5', 'years_of_service_grouped', 'dept', 'desk', 'tier', 'race_grouping', 'years_of_service_grouped']
rename(columns={'race_grouping_person_of_color': 'race_grouping_person_of_color', 'age_group_5_25_under': 'age_group_5_25-29', 'age_group_5_25to29': 'age_group_5_30-34', 'age_group_5_30to34': 'age_group_5_35-39', 'age_group_5_35to39': 'age_group_5_40-44', 'age_group_5_40to44': 'age_group_5_45-49', 'age_group_5_45to49': 'age_group_5_50-54', 'age_group_5_50to54': 'age_group_5_55-59', 'age_group_5_55to59': 'age_group_5_60-64', 'age_group_5_60to64': 'age_group_5_65+', 'age_group_5_65_over': 'age_group_5_65_over', 'tier_Tier_1': 'tier_Tier_1', 'tier_Tier_2': 'tier_Tier_2', 'tier_Tier_3': 'tier_Tier_3', 'tier_Tier_4': 'tier_Tier_4', 'years_of_service_grouped_0': 'years_of_service_grouped_0', 'years_of_service_grouped_1-2': 'years_of_service_grouped_1-2', 'years_of_service_grouped_1to2': 'years_of_service_grouped_3-5', 'years_of_service_grouped_3to5': 'years_of_service_grouped_6-10', 'years_of_service_grouped_6to10': 'years_of_service_grouped_11-15', 'years_of_service_grouped_6to10': 'years_of_service_grouped_16-20', 'years_of_service_grouped_11to15': 'years_of_service_grouped_21-25', 'years_of_service_grouped_16to20': 'years_of_service_grouped_21-25', 'years_of_service_grouped_21to25': 'years_of_service_grouped_25+'})

import statsmodels.formula.api as sm
model59 = sm.ols(data=commercial_hourly_regression, formula = 'current_base_pay ~ gender_Female + gender_Male')
result59 = model59.fit()
result59.summary()
```

```
<class 'statsmodels.iolib.summary.Summary'>

OLS Regression Results
==============================================================================
Dep. Variable:  current_base_pay   R-squared:   0.085
Model:                  OLS       Adj. R-squared: 0.078
Method:          Least Squares   F-statistic: 13.41
Date:                      Wed, 06 Nov 2019 Prob (F-statistic): 0.000350
Time:                  10:28:11   Log-Likelihood: -482.21
No. Observations:      147       AIC:         968.4
Df Residuals:          145       BIC:         974.4
Df Model:                1
Covariance Type:        nonrobust
==============================================================================
```
```
[474]: model60 = sm.ols(data=commercial_hourly_regression, formula = 'current_base_pay ~ race_grouping_white + race_grouping_person_of_color')
result60 = model60.fit()
result60.summary()

[474]: <class 'statsmodels.iolib.summary.Summary'>
```

```
OLS Regression Results
==============================================================================
Dep. Variable:       current_base_pay    R-squared:                      0.105
Model:               OLS                   Adj. R-squared:                0.093
Method:              Least Squares       F-statistic:                   8.479
Date:                Wed, 06 Nov 2019   Prob (F-statistic):            0.000330
Time:                10:28:11           Log-Likelihood:                -480.53
No. Observations:    147                 AIC:                           967.1
Df Residuals:        144                 BIC:                           976.0
Df Model:            2
Covariance Type:     nonrobust
==============================================================================
 coef    std err      t    P>|t|     [0.025    0.975]
---------------------------------------------------------------
-Intercept     18.4963     0.356   51.935      0.000     17.792     19.200
gender_Female  11.2044     0.562   19.938      0.000     10.094     12.315
gender_Male    7.2918     0.564   12.923      0.000     6.177     8.407
```

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 3.84e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```
[42]: model60 = sm.ols(data=commercial_hourly_regression, formula = 'current_base_pay ~ race_grouping_white + race_grouping_person_of_color')
result60 = model60.fit()
result60.summary()
```

```
##

```
[474]: model60 = sm.ols(data=commercial_hourly_regression, formula = 'current_base_pay ~ race_grouping_white + race_grouping_person_of_color')
result60 = model60.fit()
result60.summary()

[474]: <class 'statsmodels.iolib.summary.Summary'>
```
```python
[475]: model61 = sm.ols(data=commercial_hourly_regression, formula = 'current_base_pay ~ gender_Female + gender_Male + race_grouping_white +
race_grouping_person_of_color')
result61 = model61.fit()
result61.summary()
```

```
[475]: <class 'statsmodels.iolib.summary.Summary'>
```

```
+---------------------------------------------------------------------+
<table>
<thead>
<tr>
<th>OLS Regression Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dep. Variable: current_base_pay</td>
</tr>
<tr>
<td>Model: OLS</td>
</tr>
<tr>
<td>Method: Least Squares</td>
</tr>
<tr>
<td>Date: Wed, 06 Nov 2019</td>
</tr>
<tr>
<td>Time: 10:28:11</td>
</tr>
<tr>
<td>No. Observations: 147</td>
</tr>
<tr>
<td>Df Residuals: 143</td>
</tr>
<tr>
<td>Df Model: 3</td>
</tr>
<tr>
<td>Covariance Type: nonrobust</td>
</tr>
</tbody>
</table>
+---------------------------------------------------------------------+

---

| coef     | std err | t      | P>|t| |
|----------|----------|--------|------|
| [0.025 0.975] |

---

<table>
<thead>
<tr>
<th>Intercept</th>
<th>15.9980</th>
<th>2.3970</th>
<th>6.673</th>
<th>0.000</th>
</tr>
</thead>
<tbody>
<tr>
<td>race_grouping_white</td>
<td>8.8969</td>
<td>3.8370</td>
<td>2.319</td>
<td>0.022</td>
</tr>
<tr>
<td>race_grouping_person_of_color</td>
<td>4.4273</td>
<td>3.7640</td>
<td>1.176</td>
<td>0.241</td>
</tr>
<tr>
<td>gender_Female</td>
<td>9.8826</td>
<td>1.3700</td>
<td>7.213</td>
<td>0.000</td>
</tr>
<tr>
<td>gender_Male</td>
<td>6.1154</td>
<td>1.2350</td>
<td>4.952</td>
<td>0.000</td>
</tr>
</tbody>
</table>
```
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 7.26e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
new_commercial_hourly_regression = pd.DataFrame({'gender_Female': [1, 0, 1, 0],
                                                 'gender_Male': [0, 1, 0, 1],
                                                 'race_grouping_white': [1, 1, 0, 0],
                                                 'race_grouping_person_of_color': [0, 0, 1, 1],
                                                 'age': [40, 40, 40, 40]})
new_commercial_hourly_regression['predicted'] = result61.predict(new_commercial_hourly_regression)
```

```python
model62 = sm.ols(data=new_commercial_hourly_regression, formula='current_base_pay ~ gender_Female + gender_Male + age_group_5_25_under + age_group_5_25to29 +
                       age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 +
                       age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 +
                       age_group_5_60to64 + age_group_5_65_over')
result62 = model62.fit()
result62.summary()
```

```python
<class 'statsmodels.iolib.summary.Summary'>
```

OLS Regression Results
Dep. Variable: current_base_pay  R-squared: 0.173
Model: OLS  Adj. R-squared: 0.113
Method: Least Squares  F-statistic: 2.851
Date: Wed, 06 Nov 2019  Prob (F-statistic): 0.00298
Time: 10:28:11  Log-Likelihood: -474.72
No. Observations: 147  AIC: 971.4
Df Model: 10  Covariance Type: nonrobust

==============================================================================
                coef     std err      t      P>|t|     [0.025    [0.975]
Intercept    17.3253     0.3390    51.121     0.000    16.655    17.995
gender_Female  10.5396     0.5690    18.510     0.000     9.414    11.666
gender_Male   6.7857     0.5680    11.943     0.000     5.662     7.909
age_group_5_25_under  0.1649     1.8060     0.091     0.927    -3.407     3.737
age_group_5_25to29  3.3425     1.3310     2.510     0.013     0.710     5.975
age_group_5_30to34  3.3398     1.9850     1.683     0.095    -0.585     7.264
age_group_5_35to39  6.3753     1.6730     3.811     0.000     3.067     9.683
age_group_5_40to44  1.1498     1.4970     0.768     0.444    -1.810     4.109
age_group_5_45to49  2.7692     1.4820     1.868     0.064    -0.162     5.701
age_group_5_50to54 -0.2028     1.4540    -0.139     0.889    -3.078     2.672
age_group_5_55to59  0.7987     1.5210     0.525     0.600    -2.210     3.807
age_group_5_60to64 -0.6104     1.6690    -0.366     0.715    -3.910     2.690
age_group_5_65_over  0.1982     1.8010     0.110     0.913    -3.364     3.760

==============================================================================
                Omnibus: 38.981  Durbin-Watson: 1.332
Prob(Omnibus): 0.000  Jarque-Bera (JB): 78.932
Kurtosis: 5.723  Cond. No. 1.06e+16

274
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 2.11e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```
model63 = sm.ols(data=commercial_hourly_regression, formula = 'current_base_pay ~ race_grouping_white + race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over')
result63 = model63.fit()
result63.summary()
```

```python
[478]: <class 'statsmodels.iolib.summary.Summary'>
```
age_group_5_35to39  5.6706  1.664  3.408  0.001
2.380    8.962
age_group_5_40to44  2.9884  1.481  2.019  0.046
0.060    5.916
age_group_5_45to49  2.8627  1.518  1.886  0.061
-0.140    5.865
age_group_5_50to54 -0.8331  1.476 -0.564  0.573
-3.752    2.086
age_group_5_55to59  0.8815  1.550  0.569  0.570
-2.183    3.946
age_group_5_60to64 -0.2342  1.701 -0.138  0.891
-3.599    3.130
age_group_5_65_over -0.3411  1.822 -0.187  0.852
-3.944    3.262
==============================================================================
Omnibus: 34.622 Durbin-Watson: 1.280
Prob(Omnibus): 0.000 Jarque-Bera (JB): 62.556
Skew: 1.095 Prob(JB): 2.61e-14
Kurtosis: 5.328 Cond. No. 8.18e+15
==============================================================================

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specifed.
[2] The smallest eigenvalue is 3.72e-30. This might indicate that there are
strong multicollinearity problems or that the design matrix is singular.

```
[479]: model64 = sm.ols(data=commercial_hourly_regression, formula = 'current_base_pay ~ gender_Female + gender_Male + race_grouping_white +
               race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 +
               age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 +
               age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 +
               age_group_5_60to64 + age_group_5_65_over')
result64 = model64.fit()
result64.summary()
```

```
[479]: <class 'statsmodels.iolib.summary.Summary'>
```

```
Df Model: 12  
Covariance Type: nonrobust  

+----------------------------------------+----------------------------------+------------+--------------------+----------------+
<p>|                                      | coef    | std err | t       | P&gt;|t| |
|----------------------------------------|---------|---------|---------|----------------|</p>
<table>
<thead>
<tr>
<th>[0.025 0.975]</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Intercept</td>
<td>13.7789</td>
<td>2.270</td>
<td>6.071</td>
<td>0.000</td>
</tr>
<tr>
<td>9.290</td>
<td>18.268</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>gender_Female</td>
<td>8.6081</td>
<td>1.331</td>
<td>6.469</td>
<td>0.000</td>
</tr>
<tr>
<td>5.976</td>
<td>11.240</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>gender_Male</td>
<td>5.1708</td>
<td>1.164</td>
<td>4.441</td>
<td>0.000</td>
</tr>
<tr>
<td>2.868</td>
<td>7.473</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>race_grouping_white</td>
<td>8.7881</td>
<td>3.728</td>
<td>2.357</td>
<td>0.020</td>
</tr>
<tr>
<td>1.414</td>
<td>16.162</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>race_grouping_person_of_color</td>
<td>4.5545</td>
<td>3.686</td>
<td>1.236</td>
<td>0.219</td>
</tr>
<tr>
<td>-2.735</td>
<td>11.844</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>age_group_5_25_under</td>
<td>-0.6527</td>
<td>1.749</td>
<td>-0.373</td>
<td>0.710</td>
</tr>
<tr>
<td>-4.113</td>
<td>2.807</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>age_group_5_25to29</td>
<td>2.2409</td>
<td>1.296</td>
<td>1.729</td>
<td>0.086</td>
</tr>
<tr>
<td>-0.323</td>
<td>4.805</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>age_group_5_30to34</td>
<td>3.6872</td>
<td>1.918</td>
<td>1.922</td>
<td>0.057</td>
</tr>
<tr>
<td>-0.107</td>
<td>7.481</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>age_group_5_35to39</td>
<td>5.5001</td>
<td>1.608</td>
<td>3.421</td>
<td>0.001</td>
</tr>
<tr>
<td>2.320</td>
<td>8.680</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>age_group_5_40to44</td>
<td>1.7499</td>
<td>1.436</td>
<td>1.219</td>
<td>0.225</td>
</tr>
<tr>
<td>-1.090</td>
<td>4.590</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>age_group_5_45to49</td>
<td>2.7592</td>
<td>1.450</td>
<td>1.903</td>
<td>0.059</td>
</tr>
<tr>
<td>-0.109</td>
<td>5.628</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>age_group_5_50to54</td>
<td>-0.6665</td>
<td>1.420</td>
<td>-0.469</td>
<td>0.640</td>
</tr>
<tr>
<td>-3.476</td>
<td>2.143</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>age_group_5_55to59</td>
<td>0.2605</td>
<td>1.474</td>
<td>0.177</td>
<td>0.860</td>
</tr>
<tr>
<td>-2.655</td>
<td>3.177</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>age_group_5_60to64</td>
<td>-0.5089</td>
<td>1.624</td>
<td>-0.313</td>
<td>0.754</td>
</tr>
<tr>
<td>-3.721</td>
<td>2.703</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>age_group_5_65_over</td>
<td>-0.5907</td>
<td>1.741</td>
<td>-0.339</td>
<td>0.735</td>
</tr>
<tr>
<td>-4.035</td>
<td>2.854</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------------------------------------</td>
<td>---------</td>
<td>---------</td>
<td>---------</td>
<td>----------------</td>
</tr>
<tr>
<td>Omnibus: 29.883</td>
<td>Durbin-Watson: 1.446</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Prob(Omnibus): 0.000</td>
<td>Jarque-Bera (JB): 47.654</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Kurtosis: 4.917</td>
<td>Cond. No.: 1.31e+16</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Warnings: [1] Standard Errors assume that the covariance matrix of the errors is correctly
The smallest eigenvalue is 1.88e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
# (480) merit_raises_combined_hourly_regression =
# merit_raises_combined[(merit_raises_combined['dept'] == 'Commercial') &
# (merit_raises_combined['pay_rate_type'] == 'Hourly')]
# merit_raises_combined_hourly_regression = pd.
# get_dummies(merit_raises_combined_hourly_regression,
# columns=['gender','race_grouping','age_group_5'])

# (481) merit_raises_combined_hourly_regression =
# merit_raises_combined_hourly_regression.
# rename(columns={'race_grouping_person of color':
# 'race_grouping_person_of_color','age_group_5_<25':
# 'age_group_5_25_under','age_group_5_25-29':
# 'age_group_5_25to29','age_group_5_30-34':
# 'age_group_5_30to34','age_group_5_35-39':
# 'age_group_5_35to39','age_group_5_40-44':
# 'age_group_5_40to44','age_group_5_45-49':
# 'age_group_5_45to49','age_group_5_50-54':
# 'age_group_5_50to54','age_group_5_55-59':
# 'age_group_5_55to59','age_group_5_60-64':
# 'age_group_5_60to64','age_group_5_65+':'age_group_5_65_over'})
model65 = sm.ols(data=merit_raises_combined_hourly_regression, formula =
# 'base_pay_change ~ gender_Female + gender_Male')
result65 = model65.fit()
result65.summary()
```

```
<class 'statsmodels.iolib.summary.Summary'>
```

```python
# (481) <class 'statsmodels.iolib.summary.Summary'>
```

### OLS Regression Results

<table>
<thead>
<tr>
<th>Dep. Variable:</th>
<th>base_pay_change</th>
<th>R-squared:</th>
<th>0.064</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model:</td>
<td>OLS</td>
<td>Adj. R-squared:</td>
<td>0.060</td>
</tr>
<tr>
<td>Method:</td>
<td>Least Squares</td>
<td>F-statistic:</td>
<td>17.78</td>
</tr>
<tr>
<td>Date:</td>
<td>Wed, 06 Nov 2019</td>
<td>Prob (F-statistic):</td>
<td>3.43e-05</td>
</tr>
<tr>
<td>Time:</td>
<td>10:28:12</td>
<td>Log-Likelihood:</td>
<td>35.988</td>
</tr>
<tr>
<td>Df Residuals:</td>
<td>260</td>
<td>BIC:</td>
<td>-60.84</td>
</tr>
<tr>
<td>Df Model:</td>
<td>1</td>
<td>Covariance Type:</td>
<td>nonrobust</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>================================</td>
</tr>
<tr>
<td></td>
<td>coef</td>
<td>std err</td>
<td>t</td>
</tr>
<tr>
<td></td>
<td>0.975</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
|                  | coef   | std err |    t   | P>|t| |
|------------------|--------|---------|--------|-----|
| Intercept        | 0.2835 | 0.010   | 28.425 | 0.000 |
| race_grouping_white | 0.1859 | 0.014   | 13.893 | 0.000 |
| gender_Male      | 0.1895 | 0.014   | 13.893 | 0.000 |
| gender_Female    | 0.0791 | 0.014   | 5.668  | 0.000 |
| Intercept        | 0.2686 | 0.009   | 30.779 | 0.000 |
| race_grouping_white | 0.107  | 0.009   | 30.779 | 0.000 |
| gender_Male      | 0.1895 | 0.014   | 13.893 | 0.000 |
| gender_Female    | 0.0791 | 0.014   | 5.668  | 0.000 |

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 3.34e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
[482]: model66 = sm.ols(data=merit_raises_combined_hourly_regression, formula = 'base_pay_change ~ race_grouping_white + race_grouping_person_of_color')
result66 = model66.fit()
result66.summary()

[482]: <class 'statsmodels.iolib.summary.Summary'>

OLS Regression Results
==============================================================================
Dep. Variable:    base_pay_change  R-squared:             0.032
Model:                   OLS  Adj. R-squared:           0.029
Method:                Least Squares  F-statistic:        8.727
Date:            Wed, 06 Nov 2019   Prob (F-statistic):    0.00342
No. Observations:         262  AIC:                   -59.30
Df Residuals:             260  BIC:                   -52.16
Df Model:                1
Covariance Type:        nonrobust
==============================================================================
coef    std err    t    P>|t|    [0.025    0.975]
[        ]
Intercept        0.2835     0.010   28.425 0.000
gender_Male      0.1895     0.014   13.893 0.000
race_grouping_white  0.1859   0.014    10.443 0.000
```
0.151 0.221
race_grouping_person_of_color 0.0976 0.013 7.264 0.000
0.071 0.124

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 2.75e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```
model67 = sm.ols(data=merit_raises_combined_hourly_regression, formula='base_pay_change ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color')
result67 = model67.fit()
result67.summary()
```

```
OLS Regression Results

==============================================================================
Dep. Variable: base_pay_change R-squared: 0.101
Model: OLS Adj. R-squared: 0.094
Method: Least Squares F-statistic: 14.58
Date: Wed, 06 Nov 2019 Prob (F-statistic): 9.99e-07
Time: 10:28:12 Log-Likelihood: 41.300
No. Observations: 262 AIC: -76.60
Df Residuals: 259 BIC: -65.90
Df Model: 2
Covariance Type: nonrobust
==============================================================================
coef  std err  t  P>|t|
[0.025  0.975]

---------------------------------------------------------------------
Intercept             0.2123     0.007    29.381    0.000
gender_Female        0.1634     0.013    12.262    0.000
        0.022     0.075
new_reason_for_change_combined_regression = pd.DataFrame({'gender_Female': [1, 0, 1, 0], 'gender_Male': [0, 1, 0, 1], 'race_grouping_white': [1, 1, 0, 0], 'race_grouping_person_of_color': [0, 0, 1, 1]})
new_reason_for_change_combined_regression['predicted'] = result67['predict(new_reason_for_change_combined_regression)']

model68 = sm.ols(data=merit_raises_combined_hourly_regression, formula='base_pay_change ~ gender_Female + gender_Male + age_group_5_25_under +
age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 +
age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 +
age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over')
result68 = model68.fit()
result68.summary()
| Dep. Variable: | base_pay_change | R-squared: | 0.127 |
| Model: | OLS | Adj. R-squared: | 0.092 |
| Method: | Least Squares | F-statistic: | 3.651 |
| Date: | Wed, 06 Nov 2019 | Prob (F-statistic): | 0.000145 |
| Time: | 10:28:12 | Log-Likelihood: | 45.112 |
| No. Observations: | 262 | AIC: | -68.22 |
| Df Residuals: | 251 | BIC: | -28.97 |
| Df Model: | 10 |
| Covariance Type: | nonrobust |

| coef | std err | t | P>|t| | [0.025 |
|---|---|---|---|---|
| Intercept | 0.2639 | 0.009 | 27.788 | 0.000 | 0.245 |
| gender_Female | 0.1855 | 0.015 | 12.286 | 0.000 | 0.156 |
| gender_Male | 0.0784 | 0.015 | 5.216 | 0.000 | 0.049 |
| age_group_5_25_under | 0.1763 | 0.086 | 2.062 | 0.040 | 0.008 |
| age_group_5_25to29 | -0.0140 | 0.040 | -0.354 | 0.724 | -0.092 |
| age_group_5_30to34 | 0.0412 | 0.048 | 0.853 | 0.395 | -0.054 |
| age_group_5_35to39 | 0.1336 | 0.043 | 3.111 | 0.002 | 0.049 |
| age_group_5_40to44 | 0.0340 | 0.039 | 0.876 | 0.382 | -0.042 |
| age_group_5_45to49 | 0.0274 | 0.036 | 0.755 | 0.451 | -0.044 |
| age_group_5_50to54 | -0.0362 | 0.035 | -1.028 | 0.305 | -0.105 |
| age_group_5_55to59 | -0.0117 | 0.034 | -0.345 | 0.730 | -0.079 |
| age_group_5_60to64 | -0.0325 | 0.036 | -0.914 | 0.362 | -0.102 |
| age_group_5_65_over | -0.0542 | 0.041 | -1.313 | 0.190 | -0.135 |

Omnibus: 112.509 Durbin-Watson: 1.803
Prob(Omnibus): 0.000 Jarque-Bera (JB): 472.182
Skew: 1.770 Prob(JB): 2.93e-103
Kurtosis: 8.543 Cond. No. 1.43e+16
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 2.07e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
model69 = sm.ols(data=merit_raises_combined_hourly_regression, formula='base_pay_change ~ race_grouping_white + race_grouping_person_of_color +
              age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 +
              age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
              age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
              age_group_5_65_over')
result69 = model69.fit()
result69.summary()
```

```
OLS Regression Results
==============================================================================
Dep. Variable:          base_pay_change   R-squared:          0.106
Model:                 OLS               Adj. R-squared:      0.070
Method:                 Least Squares   F-statistic:        2.975
Date:                   Wed, 06 Nov 2019 Prob (F-statistic): 0.00147
Time:                   10:28:12       Log-Likelihood:     41.998
No. Observations:       262             AIC:                -62.00
Df Residuals:           251             BIC:                -22.74
Df Model:               10              Covariance Type:     nonrobust
==============================================================================
                 coef   std err          t      P>|t|    [0.025   0.975]
-----------------------------------------------
Intercept       0.2759    0.011       26.094     0.000    0.255    0.297
race_grouping_white  0.1811   0.018       9.892     0.000    0.145    0.217
race_grouping_person_of_color  0.0948   0.014       6.724     0.000    0.067    0.123
age_group_5_25_under   0.1520   0.086       1.764     0.079   -0.016    0.322
age_group_5_25to29   -0.0136   0.041      -0.334     0.738   -0.094    0.067
age_group_5_30to34   -0.016     0.048      -0.440     0.662   -0.010    0.080
age_group_5_35to39   -0.016     0.044      -0.367     0.714   -0.006    0.004
age_group_5_40to44   -0.016     0.044      -0.367     0.714   -0.006    0.004
age_group_5_45to49   -0.016     0.044      -0.367     0.714   -0.006    0.004
age_group_5_50to54   -0.016     0.044      -0.367     0.714   -0.006    0.004
age_group_5_55to59   -0.016     0.044      -0.367     0.714   -0.006    0.004
age_group_5_60to64   -0.016     0.044      -0.367     0.714   -0.006    0.004
age_group_5_65_over   -0.016     0.044      -0.367     0.714   -0.006    0.004
age_group_5_over     -0.016     0.044      -0.367     0.714   -0.006    0.004
```
<table>
<thead>
<tr>
<th>age_group_5_25_under</th>
<th>0.0950</th>
<th>0.043</th>
<th>2.196</th>
<th>0.029</th>
</tr>
</thead>
<tbody>
<tr>
<td>age_group_5_25to29</td>
<td>0.0733</td>
<td>0.038</td>
<td>1.923</td>
<td>0.056</td>
</tr>
<tr>
<td>age_group_5_30to34</td>
<td>0.0523</td>
<td>0.037</td>
<td>1.425</td>
<td>0.155</td>
</tr>
<tr>
<td>age_group_5_35to39</td>
<td>0.0574</td>
<td>0.035</td>
<td>1.937</td>
<td>0.164</td>
</tr>
<tr>
<td>age_group_5_40to44</td>
<td>0.0577</td>
<td>0.035</td>
<td>1.989</td>
<td>0.174</td>
</tr>
<tr>
<td>age_group_5_45to49</td>
<td>0.0371</td>
<td>0.032</td>
<td>1.176</td>
<td>0.241</td>
</tr>
<tr>
<td>age_group_5_50to54</td>
<td>-0.0010</td>
<td>0.036</td>
<td>-0.029</td>
<td>0.977</td>
</tr>
<tr>
<td>age_group_5_55to59</td>
<td>-0.069</td>
<td>0.067</td>
<td>-1.162</td>
<td>0.246</td>
</tr>
<tr>
<td>age_group_5_60to64</td>
<td>-0.0415</td>
<td>0.036</td>
<td>-1.162</td>
<td>0.246</td>
</tr>
<tr>
<td>age_group_5_65_over</td>
<td>-0.0828</td>
<td>0.041</td>
<td>-2.034</td>
<td>0.043</td>
</tr>
</tbody>
</table>

**Omnibus:** 113.194  
**Durbin-Watson:** 1.862  
**Prob(Omnibus):** 0.000  
**Skew:** 1.805  
**Kurtosis:** 8.343  
**Condition No.:** 9.13e+15

**Warnings:**
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 5.49e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
model70 = sm.ols(data=merit_raises_combined_hourly_regression, formula='base_pay_change ~ gender_Female + gender_Male + race_grouping_white +
race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 +
age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 +
age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 +
age_group_5_60to64 + age_group_5_65_over')
result70 = model70.fit()
result70.summary()
```

```bash
<class 'statsmodels.iolib.summary.Summary'>
```

**OLS Regression Results**

<table>
<thead>
<tr>
<th>Dep. Variable:</th>
<th>base_pay_change</th>
<th>R-squared:</th>
<th>0.162</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model:</td>
<td>OLS</td>
<td>Adj. R-squared:</td>
<td>0.126</td>
</tr>
<tr>
<td>Method:</td>
<td>Least Squares</td>
<td>F-statistic:</td>
<td>4.407</td>
</tr>
<tr>
<td>Date:</td>
<td>Wed, 06 Nov 2019</td>
<td>Prob (F-statistic):</td>
<td>4.68e-06</td>
</tr>
<tr>
<td>Time:</td>
<td>10:28:12</td>
<td>Log-Likelihood:</td>
<td>50.540</td>
</tr>
<tr>
<td>No. Observations:</td>
<td>262</td>
<td>AIC:</td>
<td>-77.08</td>
</tr>
<tr>
<td>Df Residuals:</td>
<td>250</td>
<td>BIC:</td>
<td>-34.26</td>
</tr>
</tbody>
</table>
```
Df Model: 11
Covariance Type: nonrobust

| coef | std err | t   | P>|t| |
|------|---------|-----|-----|
| 0.2117 | 0.008  | 27.065 | 0.000 |
| 0.1960 | 0.227  |       |     |
| 0.1637 | 0.015  | 11.063 | 0.000 |
| 0.1350 | 0.193  |       |     |
| 0.0480 | 0.014  | 3.323  | 0.001 |
| 0.0200 | 0.076  |       |     |
| 0.0121 | 0.188  |       |     |
| 0.0569 | 0.014  | 4.096  | 0.000 |
| 0.0300 | 0.084  |       |     |
| 0.0150 | 0.346  |       |     |
| 0.1807 | 0.084  | 2.154  | 0.032 |
| -0.0568 | 0.041  | -1.401 | 0.162 |
| -0.1370 | 0.023  |       |     |
| 0.0430 | 0.047  | 0.906  | 0.366 |
| -0.0500 | 0.136  |       |     |
| 0.1123 | 0.042  | 2.650  | 0.009 |
| 0.0290 | 0.196  |       |     |
| 0.0289 | 0.038  | 0.757  | 0.450 |
| -0.0460 | 0.104  |       |     |
| 0.0324 | 0.036  | 0.904  | 0.367 |
| -0.0380 | 0.103  |       |     |
| 0.0345 | 0.036  | 1.171  | 0.243 |
| -0.1090 | 0.028  |       |     |
| -0.0405 | 0.035  | -0.356 | 0.072 |
| -0.0780 | 0.054  |       |     |
| -0.0269 | 0.035  | -0.766 | 0.444 |
| -0.0960 | 0.042  |       |     |
| -0.0495 | 0.041  | -1.217 | 0.225 |
| -0.1300 | 0.031  |       |     |

Omnibus: 103.112  Durbin-Watson: 1.862
Prob(Omnibus): 0.000  Jarque-Bera (JB): 381.604
Skew: 1.654  Prob(JB): 1.37e-83
Kurtosis: 7.900  Cond. No. 2.58e+16

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly
specified.
[2] The smallest eigenvalue is 8.79e-31. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
model71 = sm.ols(data=merit_raises_combined_hourly_regression, formula = 'performance_rating ~ gender_Female + gender_Male')
result71 = model71.fit()
result71.summary()
```

[488]: <class 'statsmodels.iolib.summary.Summary'>

```
OLS Regression Results
================================================================================
Dep. Variable: performance_rating  R-squared:          0.064
Model: OLS                  Adj. R-squared:      0.061
Method: Least Squares      F-statistic:       17.83
Date: Wed, 06 Nov 2019     Prob (F-statistic): 3.34e-05
No. Observations: 261      AIC:              10.62
Df Residuals: 259          BIC:              17.75
Df Model: 1                  Covariance Type: nonrobust
================================================================================
                     coef     std err          t      P>|t|      [0.025
Intercept        2.1932     0.010       215.915      0.000       2.173
- gender_Female   1.1609     0.016        73.044      0.000       1.130
  1.192
  gender_Male     1.0322     0.016        63.618      0.000       1.000
  1.064
==============================================================================
Omnibus:          16.892     Durbin-Watson:    1.674
Prob(Omnibus):    0.000     Jarque-Bera (JB):  18.305
Skew:             0.633      Prob(JB):        0.000106
Kurtosis:         3.288      Cond. No.      2.38e+15
```

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 6.93e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.
```
```python
model72 = sm.ols(data=merit_raises_combined_hourly_regression, formula = 'performance_rating ~ race_grouping_white + race_grouping_person_of_color')
result72 = model72.fit()
result72.summary()
```

```
OLS Regression Results
=========================================
Dep. Variable:  performance_rating  R-squared:   0.009
Model:              OLS  Adj. R-squared:  0.005
Method:                Least Squares  F-statistic:  2.318
Date:       Wed, 06 Nov 2019  Prob (F-statistic):  0.129
Time:    10:28:12  Log-Likelihood:   -10.836
No. Observations:  261  AIC:   25.67
Df Residuals:      259  BIC:   32.80
Df Model:              1
Covariance Type:    nonrobust

=================================================================================================
                      coef    std err          t      P>|t|    [0.025  0.975]
------------------------------------------------------------------------------------------------
Intercept             2.2028    0.012   187.636      0.000    2.180    2.226
race_grouping_white   1.1282    0.021    53.858      0.000    1.087    1.169
race_grouping_person_of_color  1.0746    0.016    67.923      0.000    1.043    1.106

=================================================================================================
Omniatus:             13.746  Durbin-Watson:     1.519
Prob(Omnibus):         0.001  Jarque-Bera (JB):   14.917
Skew:                0.585  Prob(JB):           0.000577
Kurtosis:              2.976  Cond. No.           5.99e+15
```

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 1.19e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
model73 = sm.ols(data=merit_raises_combined_hourly_regression, formula = 'performance_rating ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color')
result73 = model73.fit()
```

```
```
```python
result73.summary()
```

```
[490]: <class 'statsmodels.iolib.summary.Summary'>
```

```python
print(result73.summary())
```

```python
OLS Regression Results
==============================================================================
Dep. Variable: performance_rating   R-squared: 0.076
Model: OLS  Adj. R-squared: 0.069
Method: Least Squares  F-statistic: 10.56
Date: Wed, 06 Nov 2019  Prob (F-statistic): 3.89e-05
Time: 10:28:13  Log-Likelihood: -1.7265
Df Residuals: 258  BIC: 20.15
Df Model: 2  
Covariance Type: nonrobust
==============================================================================
                               coef    std err          t      P>|t|      
Intercept                    1.6517    0.0090      193.876  0.000
gender_Female                0.8915    0.0160      56.645  0.000
gender_Male                  0.7603    0.0160      48.075  0.000
race_grouping_white          0.8561    0.0190      44.198  0.000
race_grouping_person_of_color 0.7956    0.0160      51.040  0.000
==============================================================================
 Omnibus: 17.639  Durbin-Watson: 1.701
 Prob(Omnibus): 0.000  Jarque-Bera (JB): 19.204
 Skew: 0.640  Prob(JB): 6.76e-05
 Kurtosis: 3.356  Cond. No. 3.19e+15
```

```
Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 5.46e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.
```

[491]:
```
model74 = sm.ols(data=merit_raises_combined_hourly_regression, formula = 'performance_rating ~ gender_Female + gender_Male + age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over')
result74 = model74.fit()
result74.summary()

[491]: <class 'statsmodels.iolib.summary.Summary'>

OLS Regression Results
==============================================================================
Dep. Variable:  performance_rating   R-squared:                   0.121
Model:                OLS       Adj. R-squared:             0.086
Method:     Least Squares     F-statistic:               3.439
Date:              Wed, 06 Nov 2019     Prob (F-statistic):           0.000303
No. Observations:       261             AIC:            12.36
Df Residuals:            250             BIC:            51.57
Df Model:                  10
Covariance Type:    nonrobust
==============================================================================
                 coef    std err          t      P>|t|      [0.025    0.975]
--------------------------- ----------- ------ ------- ------ ------------- -------
Intercept        2.0534      0.011  185.289    0.000     2.032    2.075
gender_Female    1.0766      0.018  60.998    0.000     1.042    1.111
gender_Male      0.9768      0.018  55.653    0.000     0.942    1.011
age_group_5_25_under  0.1699    0.100   1.703    0.090    -0.027    0.366
age_group_5_25to29  0.2144      0.046   4.637    0.000     0.123    0.305
age_group_5_30to34  0.1876      0.056   3.330    0.001     0.077    0.299
age_group_5_35to39  0.2790      0.050   5.572    0.000     0.180    0.378
age_group_5_40to44  0.2975      0.045   6.572    0.000     0.208    0.387
age_group_5_45to49  0.1887      0.042   4.459    0.000     0.105    0.272
age_group_5_50to54  0.2140      0.041   5.215    0.000     0.133    0.295

289
model75 = sm.ols(data=merit_raises_combined_hourly_regression, formula = 'performance_rating ~ race_grouping_white + race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over')
result75 = model75.fit()
result75.summary()

```
OLS Regression Results
==============================================================================
Dep. Variable: performance_rating   R-squared:                       0.094
Model: OLS   Adj. R-squared:               0.058
Method: Least Squares   F-statistic:          2.601
Date: Wed, 06 Nov 2019   Prob (F-statistic):      0.00509
Time: 10:28:13   Log-Likelihood:       0.91884
No. Observations: 261   AIC:                 20.16
Df Residuals: 250   BIC:                 59.37
Df Model: 10   Covariance Type: nonrobust
==============================================================================
               coef    std err          t      P>|t|    [0.025   0.975]
---------------------------             ---------------------------
age_group_5_55to59     0.2644    0.0400    6.595   0.000     0.185
age_group_5_60to64     0.1640    0.0410    3.958   0.000     0.082
age_group_5_65_over    0.0739    0.0480    1.535   0.126    -0.021

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 4.15e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.
```
<table>
<thead>
<tr>
<th>Intercept</th>
<th>2.0590</th>
<th>0.012</th>
<th>166.409</th>
<th>0.000</th>
</tr>
</thead>
<tbody>
<tr>
<td>race_grouping_white</td>
<td>1.0502</td>
<td>0.021</td>
<td>49.016</td>
<td>0.000</td>
</tr>
<tr>
<td>race_grouping_person_of_color</td>
<td>1.0088</td>
<td>0.017</td>
<td>61.107</td>
<td>0.000</td>
</tr>
<tr>
<td>age_group_5_25_under</td>
<td>0.1440</td>
<td>0.101</td>
<td>1.428</td>
<td>0.154</td>
</tr>
<tr>
<td>age_group_5_25to29</td>
<td>0.2284</td>
<td>0.048</td>
<td>4.798</td>
<td>0.000</td>
</tr>
<tr>
<td>age_group_5_30to34</td>
<td>0.2191</td>
<td>0.057</td>
<td>3.872</td>
<td>0.000</td>
</tr>
<tr>
<td>age_group_5_35to39</td>
<td>0.2498</td>
<td>0.051</td>
<td>4.934</td>
<td>0.000</td>
</tr>
<tr>
<td>age_group_5_40to44</td>
<td>0.3326</td>
<td>0.045</td>
<td>7.460</td>
<td>0.000</td>
</tr>
<tr>
<td>age_group_5_45to49</td>
<td>0.2070</td>
<td>0.043</td>
<td>4.816</td>
<td>0.000</td>
</tr>
<tr>
<td>age_group_5_50to54</td>
<td>0.2125</td>
<td>0.042</td>
<td>5.100</td>
<td>0.000</td>
</tr>
<tr>
<td>age_group_5_55to59</td>
<td>0.2703</td>
<td>0.041</td>
<td>6.634</td>
<td>0.000</td>
</tr>
<tr>
<td>age_group_5_60to64</td>
<td>0.1514</td>
<td>0.042</td>
<td>3.619</td>
<td>0.000</td>
</tr>
<tr>
<td>age_group_5_65_over</td>
<td>0.0440</td>
<td>0.048</td>
<td>0.924</td>
<td>0.357</td>
</tr>
</tbody>
</table>

Omnibus: 9.530 Durbin-Watson: 1.609
Prob(Omnibus): 0.009 Jarque-Bera (JB): 9.632
Skew: 0.464 Prob(JB): 0.00810
Kurtosis: 3.151 Cond. No. 6.30e+15

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 1.15e-29. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.

```python
model76 = sm.ols(data=merit_raises_combined_hourly_regression, formula='performance_rating ~ gender_Female + gender_Male + race_grouping_white +
race_grouping_person_of_color + age_group_5_25_under + age_group_5_25to29 +
age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 +
age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 +
age_group_5_60to64 + age_group_5_65_over')
result76 = model76.fit()
```
```python
result76.summary()
```

```
[493]: <class 'statsmodels.iolib.summary.Summary'>

OLS Regression Results
=========================================================
Dep. Variable: performance_rating  R-squared: 0.128
Model: OLS  Adj. R-squared: 0.090
Method: Least Squares  F-statistic: 3.330
Date: Wed, 06 Nov 2019  Prob (F-statistic): 0.000269
Time: 10:28:13  Log-Likelihood: 5.9135
No. Observations: 261  AIC: 12.17
Df Residuals: 249  BIC: 54.95
Df Model: 11
Covariance Type: nonrobust

=======================================================================================================================
                 coef     std err          t      P>|t|      [0.025     0.975
--------------------------------------------------------------------------------
Intercept         1.5701      0.009   169.243      0.000   1.552      1.588
gender_Female     0.8372      0.018    47.621      0.000   0.803      0.872
gender_Male       0.7329      0.017    42.711      0.000   0.699      0.767
race_grouping_white  0.8109      0.020    40.043      0.000   0.771      0.851
race_grouping_person_of_color  0.7592      0.016    46.090      0.000   0.727      0.792
age_group_5_25_under  0.1266      0.099     1.273      0.204  -0.069      0.322
age_group_5_25to29   0.1463      0.048     3.046      0.003   0.098      0.241
age_group_5_30to34   0.1430      0.056     2.541      0.012  -0.032      0.264
age_group_5_35to39   0.2222      0.050     4.421      0.000   0.123      0.321
age_group_5_40to44   0.2493      0.045     5.514      0.000   0.159      0.338
age_group_5_45to49   0.1458      0.042     3.433      0.001   0.062      0.229
age_group_5_50to54   0.1662      0.041     4.053      0.000   0.085      0.247
age_group_5_55to59   0.2185      0.040     5.448      0.000   0.139      0.297
```
<table>
<thead>
<tr>
<th>age_group_5_60to64</th>
<th>0.1214</th>
<th>0.042</th>
<th>2.917</th>
<th>0.004</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.039</td>
<td>0.203</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>age_group_5_65_over</td>
<td>0.0308</td>
<td>0.048</td>
<td>0.639</td>
<td>0.524</td>
</tr>
<tr>
<td>-0.064</td>
<td>0.126</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Omnibus: 13.454  Durbin-Watson: 1.737
Prob(Omnibus): 0.001  Jarque-Bera (JB): 14.029
Skew: 0.544  Prob(JB): 0.000899
Kurtosis: 3.329  Cond. No. 1.70e+16

Warnings:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The smallest eigenvalue is 2.02e-30. This might indicate that there are strong multicollinearity problems or that the design matrix is singular.
Appendix B: Analysis in R

Washington Post Newspaper Guild Pay Study 2019

This is the study of Washington Post Guild members’ salaries based on data turned over by management of The Washington Post on July 2, 2019, pursuant to a request by members of the Guild. Management turned over two Excel files: one file detailing the salaries of current guild members working for The Post (as of the date of transmission) and one file detailing the salaries of past guild members who worked for The Post and have left the organization in the past five years.

What follows is an attempt to understand pay at The Washington Post. No individual analysis should be taken on its own to mean that disparities in pay do or do not exist. This study will start with summary analysis of trends and will dive deeper as the study goes on.

The only data manipulation done prior to analysis was taking the data out of Excel and putting the files into CSV files, converting dates from ‘MM/DD/YYYY’ to ‘YYYY-MM-DD’ and removing commas from monetary columns where values exceeded 1,000.

Importing data

```r
library(tidyverse)
library(dplyr)
library(lubridate)
library(data.table)
library(fastDummies)

df <- read_csv('csvs/active_wd.csv')
df2 <- read_csv('csvs/terminated_wd.csv')
```

Add fields for analysis

Add age field

```r
data_date <- as.Date('2019-07-02')

df <- df %>% mutate(date_of_birth = ymd(date_of_birth),
                   age = floor(decimal_date(data_date) - decimal_date(date_of_birth)))
df2 <- df2 %>% mutate(date_of_birth = ymd(date_of_birth),
                    age = floor(decimal_date(data_date) - decimal_date(date_of_birth)))
```

Add years of service field

```r
df <- df %>% mutate(hire_date = ymd(hire_date),
                    years_of_service = floor(decimal_date(data_date) - decimal_date(hire_date)))
df2 <- df2 %>% mutate(hire_date = ymd(hire_date),
                    years_of_service = floor(decimal_date(data_date) - decimal_date(hire_date)))
```
Add field for 5-year age groups

agebreaks5 <- c(0, 25, 30, 35, 40, 45, 50, 55, 60, 65, 100)

setDT(df)[, age_group_5 := cut(age, 
  breaks = agebreaks5, 
  right = FALSE, 
  labels = agelabels5)]

setDT(df2)[, age_group_5 := cut(age, 
  breaks = agebreaks5, 
  right = FALSE, 
  labels = agelabels5)]

Add field for 10-year age groups

agebreaks10 <- c(0, 25, 35, 45, 55, 65, 100)
agelabels10 <- c('<25', '25-34', '35-44', '45-54', '55-64', '65+')

setDT(df)[, age_group_10 := cut(age, 
  breaks = agebreaks10, 
  right = FALSE, 
  labels = agelabels10)]

setDT(df2)[, age_group_10 := cut(age, 
  breaks = agebreaks10, 
  right = FALSE, 
  labels = agelabels10)]

Add field for years-of-service groups

yosbreaks <- c(0, 1, 3, 6, 11, 16, 21, 26, 100)
yoslabels <- c('0', '1-2', '3-5', '6-10', '11-15', '16-20', '21-25', '25+')

setDT(df)[, years_of_service_grouped := cut(years_of_service, 
  breaks = yosbreaks, 
  right = FALSE, 
  labels = yoslabels)]

setDT(df2)[, years_of_service_grouped := cut(years_of_service, 
  breaks = yosbreaks, 
  right = FALSE, 
  labels = yoslabels)]

Group departments

df <- df %>%
  mutate(dept = case_when(
    department == 'News' ~ 'News',
    department == 'Editorial' ~ 'News',
    department == 'Client Solutions' ~ 'Commercial',
    department == 'Circulation' ~ 'Commercial',
  )

2
df <- Group desks

df2 <- df2 %>%
mutate(dept = case_when(
  department == 'Finance' ~ 'Commercial',
  department == 'Marketing' ~ 'Commercial',
  department == 'WP News Media Services' ~ 'Commercial',
  department == 'Production' ~ 'Commercial',
  department == 'Public Relations' ~ 'Commercial',
  department == 'Administration' ~ 'Commercial',
  department == 'Product' ~ 'Commercial',
  TRUE ~ 'Other'))

df2 <- df2 %>%
mutate(dept = case_when(
  department == 'News' ~ 'News',
  department == 'Editorial' ~ 'News',
  department == 'News Service and Syndicate' ~ 'News',
  department == 'Audience Development and Insights' ~ 'Commercial',
  department == 'Client Solutions' ~ 'Commercial',
  department == 'Customer Care and Logistics' ~ 'Commercial',
  department == 'Finance' ~ 'Commercial',
  department == 'Legal' ~ 'Commercial',
  department == 'Marketing' ~ 'Commercial',
  department == 'WP News Media Services' ~ 'Commercial',
  department == 'Production' ~ 'Commercial',
  department == 'Public Relations' ~ 'Commercial',
  department == 'Washington Post Live' ~ 'Commercial',
  department == 'Product' ~ 'Commercial',
  TRUE ~ 'Other'))

Group desks

df <- df %>%
mutate(desk = case_when(
  cost_center_current == '110000 News Operations' ~ 'Operations',
  cost_center_current == '110001 News Digital Operations' ~ 'Operations',
  cost_center_current == '110610 Audience Development and Engagement' ~ 'Audience Development and Engagement',
  cost_center_current == '110620 News Audio' ~ 'Audio',
  cost_center_current == '110604 Presentation Design' ~ 'Design',
  cost_center_current == '110605 Presentation' ~ 'Design',
  cost_center_current == '110664 News National Apps' ~ 'Emerging News Products',
  cost_center_current == '110665 News The Lily' ~ 'Emerging News Products',
  cost_center_current == '110666 News Snapchat' ~ 'Emerging News Products',
  cost_center_current == '110667 News By The Way' ~ 'Emerging News Products',
  cost_center_current == '113210 Economy and Business' ~ 'Financial',
  cost_center_current == '114000 Foreign Administration' ~ 'Foreign',
  cost_center_current == '114095 News Foreign Brazil' ~ 'Foreign',
  cost_center_current == '114100 Foreign Latam' ~ 'Foreign',
  cost_center_current == '114220 News Foreign Istanbul' ~ 'Foreign',
  cost_center_current == '114235 Foreign Western Europe' ~ 'Foreign',
  cost_center_current == '114300 News Foreign West Africa' ~ 'Foreign',
  cost_center_current == '114415 Foreign Hong Kong' ~ 'Foreign',
  cost_center_current == '114405 Foreign Beijing Bureau' ~ 'Foreign',
  cost_center_current == '114105 Foreign Mexico Bureau' ~ 'Foreign',
  cost_center_current == '114005 Foreign Beirut Bureau' ~ 'Foreign',
  cost_center_current == '114400 Foreign India Bureau' ~ 'Foreign',
  TRUE ~ 'Other'))

3
df2 <- df2 %>%
  mutate(desk = case_when(
    cost_center_current == '114410 Foreign Tokyo Bureau' ~ 'Foreign',
    cost_center_current == '114205 Foreign Islamabad Bureau' ~ 'Foreign',
    cost_center_current == '114305 Foreign Nairobi Bureau' ~ 'Foreign',
    cost_center_current == '114240 Foreign Rome Bureau' ~ 'Foreign',
    cost_center_current == '114200 Foreign London Bureau' ~ 'Foreign',
    cost_center_current == '114230 Foreign Moscow Bureau' ~ 'Foreign',
    cost_center_current == '114225 Foreign Cairo Bureau' ~ 'Foreign',
    cost_center_current == '114215 Foreign Berlin Bureau' ~ 'Foreign',
    cost_center_current == '110603 Presentation Graphics' ~ 'Graphics',
    cost_center_current == '110450 Investigative' ~ 'Investigative',
    cost_center_current == '112300 Local Politics and Government' ~ 'Local',
    cost_center_current == '110601 Multiplatform Desk' ~ 'Multiplatform',
    cost_center_current == '110500 Magazine' ~ 'National',
    cost_center_current == '113200 National Politics and Government' ~ 'National',
    cost_center_current == '113205 National Security' ~ 'National',
    cost_center_current == '113215 News National Health & Science' ~ 'National',
    cost_center_current == '113220 National Enterprise' ~ 'National',
    cost_center_current == '113235 National America' ~ 'National',
    cost_center_current == '113240 News National Environment' ~ 'National',
    cost_center_current == '110006 News Content & Research' ~ 'News Content and Research',
    cost_center_current == '110455 News Logistics' ~ 'News Logistics',
    cost_center_current == '110410 Book World' ~ 'Outlook',
    cost_center_current == '110460 Outlook' ~ 'Outlook',
    cost_center_current == '110475 Polling' ~ 'Polling',
    cost_center_current == '110015 Sports Main' ~ 'Sports',
    cost_center_current == '110300 Style' ~ 'Style',
    cost_center_current == '110435 Food' ~ 'Style',
    cost_center_current == '110485 Travel' ~ 'Style',
    cost_center_current == '110495 Local Living' ~ 'Style',
    cost_center_current == '110505 Weekend' ~ 'Style',
    cost_center_current == '110600 Universal Desk' ~ 'Universal Desk',
    cost_center_current == '110652 News Video - General' ~ 'Video',
    cost_center_current == '110663 Wake Up Report' ~ 'Other',
    cost_center_current == '115000 Editorial Administration' ~ 'Editorial',
    TRUE ~ 'non-newsroom'))
df <- df %>%
  mutate(tier = case_when(
    desk == 'National' ~ 'Tier 1',
    desk == 'Foreign' ~ 'Tier 1',
    desk == 'Financial' ~ 'Tier 1',
    desk == 'Investigative' ~ 'Tier 1',
    desk == 'Style' ~ 'Tier 2',
    desk == 'Local' ~ 'Tier 2',
    TRUE ~ 'non-newsroom'))

Group desks by median salary ranges
df <- df %>%
  group_by(race)

df2 <- df %>%
  mutate(race_grouping = case_when(  
    desk == 'Graphics' ~ 'Tier 2',
    desk == 'Universal Desk' ~ 'Tier 2',
    desk == 'Sports' ~ 'Tier 2',
    desk == 'Outlook' ~ 'Tier 2',
    desk == 'Editorial' ~ 'Tier 2',
    desk == 'Audio' ~ 'Tier 3',
    desk == 'Polling' ~ 'Tier 3',
    desk == 'Design' ~ 'Tier 3',
    desk == 'Operations' ~ 'Tier 3',
    desk == 'Multiplatform' ~ 'Tier 3',
    desk == 'Video' ~ 'Tier 3',
    desk == 'Audience Development and Engagement' ~ 'Tier 3',
    desk == 'News Logistics' ~ 'Tier 4',
    desk == 'News Content and Research' ~ 'Tier 4',
    desk == 'Emerging News Products' ~ 'Tier 4',
    desk == 'Other' ~ 'Tier 4',
    TRUE ~ 'Other'))

df2 <- df2 %>%
  mutate(tier = case_when(  
    desk == 'National' ~ 'Tier 1',
    desk == 'Foreign' ~ 'Tier 1',
    desk == 'Financial' ~ 'Tier 1',
    desk == 'Investigative' ~ 'Tier 1',
    desk == 'Style' ~ 'Tier 2',
    desk == 'Local' ~ 'Tier 2',
    desk == 'Graphics' ~ 'Tier 2',
    desk == 'Universal Desk' ~ 'Tier 2',
    desk == 'Sports' ~ 'Tier 2',
    desk == 'Outlook' ~ 'Tier 2',
    desk == 'Editorial' ~ 'Tier 2',
    desk == 'Audio' ~ 'Tier 3',
    desk == 'Polling' ~ 'Tier 3',
    desk == 'Design' ~ 'Tier 3',
    desk == 'Operations' ~ 'Tier 3',
    desk == 'Multiplatform' ~ 'Tier 3',
    desk == 'Video' ~ 'Tier 3',
    desk == 'Audience Development and Engagement' ~ 'Tier 3',
    desk == 'News Logistics' ~ 'Tier 4',
    desk == 'News Content and Research' ~ 'Tier 4',
    desk == 'Emerging News Products' ~ 'Tier 4',
    desk == 'Other' ~ 'Tier 4',
    TRUE ~ 'Other'))

Group race and ethnicity

df <- df %>%
  mutate(race_grouping = case_when(  
    race_ethnicity == 'White (United States of America)' ~ 'white',
    race_ethnicity == 'Black or African American (United States of America)' ~ 'person of color',
    race_ethnicity == 'Asian (United States of America)' ~ 'person of color',
    race_ethnicity == 'Hispanic or Latino (United States of America)' ~ 'person of color',
    race_ethnicity == 'Two or More Races (United States of America)' ~ 'person of color',
    TRUE ~ 'other'))
Employee pay change grouping

```r
race_ethnicity <- 'American Indian or Alaska Native (United States of America)' - 'person of color'
race_ethnicity <- 'Native Hawaiian or Other Pacific Islander (United States of America)' - 'person of color'
 TRUE - 'unknown')

df2 <- df2 %>%
  mutate(race_grouping = case_when(
    race_ethnicity == 'White (United States of America)' ~ 'white',
    race_ethnicity == 'Black or African American (United States of America)' ~ 'person of color',
    race_ethnicity == 'Asian (United States of America)' ~ 'person of color',
    race_ethnicity == 'Hispanic or Latino (United States of America)' ~ 'person of color',
    race_ethnicity == 'Two or More Races (United States of America)' ~ 'person of color',
    race_ethnicity == 'American Indian or Alaska Native (United States of America)' ~ 'person of color',
    TRUE ~ 'unknown'))
```

```r
reason_for_change1 <- df[, c('business_process_reason1', 'base_pay_change1', 'effective_date1', 'pay_rate_type1')]
reason_for_change2 <- df[, c('business_process_reason2', 'base_pay_change2', 'effective_date2', 'pay_rate_type2')]
reason_for_change3 <- df[, c('business_process_reason3', 'base_pay_change3', 'effective_date3', 'pay_rate_type3')]
reason_for_change4 <- df[, c('business_process_reason4', 'base_pay_change4', 'effective_date4', 'pay_rate_type4')]
reason_for_change5 <- df[, c('business_process_reason5', 'base_pay_change5', 'effective_date5', 'pay_rate_type5')]
reason_for_change6 <- df[, c('business_process_reason6', 'base_pay_change6', 'effective_date6', 'pay_rate_type6')]
reason_for_change7 <- df[, c('business_process_reason7', 'base_pay_change7', 'effective_date7', 'pay_rate_type7')]
reason_for_change8 <- df[, c('business_process_reason8', 'base_pay_change8', 'effective_date8', 'pay_rate_type8')]
reason_for_change9 <- df[, c('business_process_reason9', 'base_pay_change9', 'effective_date9', 'pay_rate_type9')]
reason_for_change10 <- df[, c('business_process_reason10', 'base_pay_change10', 'effective_date10', 'pay_rate_type10')]
reason_for_change11 <- df[, c('business_process_reason11', 'base_pay_change11', 'effective_date11', 'pay_rate_type11')]
reason_for_change12 <- df[, c('business_process_reason12', 'base_pay_change12', 'effective_date12', 'pay_rate_type12')]
reason_for_change13 <- df[, c('business_process_reason13', 'base_pay_change13', 'effective_date13', 'pay_rate_type13')]
reason_for_change14 <- df[, c('business_process_reason14', 'base_pay_change14', 'effective_date14', 'pay_rate_type14')]
reason_for_change15 <- df[, c('business_process_reason15', 'base_pay_change15', 'effective_date15', 'pay_rate_type15')]
reason_for_change16 <- df[, c('business_process_reason16', 'base_pay_change16', 'effective_date16', 'pay_rate_type16')]
reason_for_change17 <- df[, c('business_process_reason17', 'base_pay_change17', 'effective_date17', 'pay_rate_type17')]
reason_for_change18 <- df[, c('business_process_reason18', 'base_pay_change18', 'effective_date18', 'pay_rate_type18')]
reason_for_change19 <- df[, c('business_process_reason19', 'base_pay_change19', 'effective_date19', 'pay_rate_type19')]
reason_for_change20 <- df[, c('business_process_reason20', 'base_pay_change20', 'effective_date20', 'pay_rate_type20')]
reason_for_change21 <- df[, c('business_process_reason21', 'base_pay_change21', 'effective_date21', 'pay_rate_type21')]
reason_for_change22 <- df[, c('business_process_reason22', 'base_pay_change22', 'effective_date22', 'pay_rate_type22')]
reason_for_change23 <- df[, c('business_process_reason23', 'base_pay_change23', 'effective_date23', 'pay_rate_type23')]
reason_for_change24 <- df[, c('business_process_reason24', 'base_pay_change24', 'effective_date24', 'pay_rate_type24')]
reason_for_change25 <- df[, c('business_process_reason25', 'base_pay_change25', 'effective_date25', 'pay_rate_type25')]
reason_for_change26 <- df[, c('business_process_reason26', 'base_pay_change26', 'effective_date26', 'pay_rate_type26')]
reason_for_change27 <- df[, c('business_process_reason27', 'base_pay_change27', 'effective_date27', 'pay_rate_type27')]
reason_for_change28 <- df[, c('business_process_reason28', 'base_pay_change28', 'effective_date28', 'pay_rate_type28')]
reason_for_change29 <- df[, c('business_process_reason29', 'base_pay_change29', 'effective_date29', 'pay_rate_type29')]
reason_for_change30 <- df[, c('business_process_reason30', 'base_pay_change30', 'effective_date30', 'pay_rate_type30')]
reason_for_change31 <- df[, c('business_process_reason31', 'base_pay_change31', 'effective_date31', 'pay_rate_type31')]

names(reason_for_change1) <- c('business_process_reason', 'base_pay_change', 'effective_date', 'pay_rate_type')
names(reason_for_change2) <- c('business_process_reason', 'base_pay_change', 'effective_date', 'pay_rate_type')
names(reason_for_change3) <- c('business_process_reason', 'base_pay_change', 'effective_date', 'pay_rate_type')
names(reason_for_change4) <- c('business_process_reason', 'base_pay_change', 'effective_date', 'pay_rate_type')
```

reason_for_change_combined <- rbind(reason_for_change1, reason_for_change2, reason_for_change3, reason_for_change4, reason_for_change5)

Employee performance evaluation grouping

fifteen1 <- df[, c('2015_annual_performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept')]
fifteen2 <- df2[, c('2015_annual_performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept')]
sixteen1 <- df[, c('2016_annual_performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept')]
sixteen2 <- df2[, c('2016_annual_performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept')]
seventeen1 <- df[, c('2017_annual_performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept')]
seventeen2 <- df2[, c('2017_annual_performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept')]
eighteen1 <- df[, c('2018_annual_performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept')]
eighteen2 <- df2[, c('2018_annual_performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept')]

names(fifteen1) <- c('performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept')
names(fifteen2) <- c('performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept')
names(sixteen1) <- c('performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept')
names(sixteen2) <- c('performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept')
names(seventeen1) <- c('performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept')
names(seventeen2) <- c('performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept')
names(eighteen1) <- c('performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept')
names(eighteen2) <- c('performance_rating', 'gender', 'race_ethnicity', 'race_grouping', 'dept')

ratings_combined <- rbind(fifteen1, fifteen2, sixteen1, sixteen2, seventeen1, seventeen2, eighteen1, eighteen2)
Create departmental data frames

```r
news_salaried <- filter(df, dept == 'News', pay_rate_type == 'Salaried')
news_hourly <- filter(df, dept == 'News', pay_rate_type == 'Hourly')
commercial_salaried <- filter(df, dept == 'Commercial', pay_rate_type == 'Salaried')
commercial_hourly <- filter(df, dept == 'Commercial', pay_rate_type == 'Hourly')
```

```r
news_salaried2 <- filter(df2, dept == 'News', pay_rate_type == 'Salaried')
news_hourly2 <- filter(df2, dept == 'News', pay_rate_type == 'Hourly')
commercial_salaried2 <- filter(df2, dept == 'Commercial', pay_rate_type == 'Salaried')
commercial_hourly2 <- filter(df2, dept == 'Commercial', pay_rate_type == 'Hourly')
```

Suppress Results

Suppress results where there are less than five employees

```r
suppress <- function(results) {
  results <- filter(results, count >= 5)
  return(results)
}
```

Suppress results and order them by count of employees

```r
suppress_count <- function(results) {
  results <- filter(results, count >= 5)
  results <- results[order(-results$count),]
  return(results)
}
```

Suppress results and order them by median salary of employees

```r
suppress_median <- function(results) {
  results <- filter(results, count >= 5)
  results <- results[order(-results$median),]
  return(results)
}
```

Summary Analysis

Employee counts

```r
current_employee_count = nrow(df)
terminated_employee_count = nrow(df2)
```

```r
cat('Total employees in data: ', current_employee_count + terminated_employee_count, '\n')
```

## Total employees in data: 1489

```r
cat('Current employees: ', current_employee_count, '\n')
```

## Current employees: 950

```r
cat('Terminated employees: ', terminated_employee_count, '\n')
```

## Terminated employees: 539
current_salaried_employee_count <- nrow(filter(df,pay_rate_type == 'Salaried'))
terminated_salaried_employee_count <- nrow(filter(df2,pay_rate_type == 'Salaried'))

cat('Total salaried employees in data:', current_salaried_employee_count + terminated_salaried_employee_count, '\n')
## Total salaried employees in data: 989

cat('Current salaried employees: ', current_salaried_employee_count, '\n')
## Current salaried employees: 707

cat('Terminated salaried employees: ', terminated_salaried_employee_count, '\n')
## Terminated salaried employees: 282

current_hourly_employee_count <- nrow(filter(df,pay_rate_type == 'Hourly'))
terminated_hourly_employee_count <- nrow(filter(df2,pay_rate_type == 'Hourly'))

cat('Total hourly employees in data: ', current_hourly_employee_count + terminated_hourly_employee_count, '\n')
## Total hourly employees in data: 500

cat('Current hourly employees: ', current_hourly_employee_count, '\n')
## Current hourly employees: 243

cat('Terminated hourly employees: ', terminated_hourly_employee_count, '\n')
## Terminated hourly employees: 257

Salary information

current_mean_salary = mean(df$current_base_pay[df$pay_rate_type == 'Salaried'])
cat('The mean yearly pay for current salaried employees is $', current_mean_salary, '\n')
## The mean yearly pay for current salaried employees is $ 112383

current_median = median(df$current_base_pay[df$pay_rate_type == 'Salaried'])
cat('The median yearly pay for current salaried employees is $', current_median)  
## The median yearly pay for current salaried employees is $ 99903.95

current_mean_hourly <- mean(df$current_base_pay[df$pay_rate_type == 'Hourly'])
cat('The mean rate for current hourly employees at The Washington Post is $', current_mean_hourly, '\n')
## The mean rate for current hourly employees at The Washington Post is $ 30.19712

current_median_hourly <- median(df$current_base_pay[df$pay_rate_type == 'Hourly'])
cat('The median rate for current hourly employees at The Washington Post is $', current_median_hourly)
## The median rate for current hourly employees at The Washington Post is $ 29.23

Employee gender

current_employee_gender <- df %>% group_by(gender)
current_employee_gender <- current_employee_gender %>% summarise(  
  count = length(current_base_pay)
)
suppress(current_employee_gender)
# A tibble: 2 x 2
#  gender count
#  <chr> <int>
# 1 Female 507
# 2 Male 443

terminated_employee_gender <- df2 %>% group_by(gender)
terminated_employee_gender <- terminated_employee_gender %>% summarise(
  count = length(current_base_pay)
)
suppress(terminated_employee_gender)

# A tibble: 2 x 2
#  gender count
#  <chr> <int>
# 1 Female 293
# 2 Male 246

current_median_gender <- filter(df, pay_rate_type == 'Salaried') %>% group_by(gender)
current_median_gender <- current_median_gender %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_median_gender)

# A tibble: 2 x 3
#  gender count median
#  <chr> <int> <dbl>
# 1 Female 370 91816.
# 2 Male 337 109928.

current_median_hourly_gender <- filter(df, pay_rate_type == 'Hourly') %>% group_by(gender)
current_median_hourly_gender <- current_median_hourly_gender %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_median_hourly_gender)

# A tibble: 2 x 3
#  gender count median
#  <chr> <int> <dbl>
# 1 Female 137 30.8
# 2 Male 106 25.8

current_age_gender_salaried <- filter(df, pay_rate_type == 'Salaried') %>% group_by(gender)
current_age_gender_salaried %>% summarise(
  median_age = median(age)
)

# A tibble: 2 x 2
#  gender median_age
#  <chr>     <dbl>
# 1 Female     35
# 2 Male      41
Employee race and ethnicity

current_employee_race_ethnicity <- df %>% group_by(race_ethnicity)
current_employee_race_ethnicity <- current_employee_race_ethnicity %>% summarise(
  count = length(current_base_pay)
)
suppress_count(current_employee_race_ethnicity)

## # A tibble: 7 x 2
## race_ethnicity count
## <chr>          <int>
## 1 White (United States of America) 612
## 2 Black or African American (United States of America) 157
## 3 Asian (United States of America) 77
## 4 Hispanic or Latino (United States of America) 45
## 5 <NA>          22
## 6 Two or More Races (United States of America) 18
## 7 Prefer Not to Disclose (United States of America) 14

terminated_employee_race_ethnicity <- df2 %>% group_by(race_ethnicity)
terminated_employee_race_ethnicity <- terminated_employee_race_ethnicity %>% summarise(
  count = length(current_base_pay)
)
suppress_count(terminated_employee_race_ethnicity)

## # A tibble: 6 x 2
## race_ethnicity count
## <chr>          <int>
## 1 White (United States of America) 291
## 2 Black or African American (United States of America) 162
## 3 Asian (United States of America) 46
## 4 Hispanic or Latino (United States of America) 20
## 5 <NA>          11
## 6 Two or More Races (United States of America) 7

current_median_race <- filter(df, pay_rate_type == 'Salaried') %>% group_by(race_ethnicity)
current_median_race <- current_median_race %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_median_race)

## # A tibble: 7 x 3
## race_ethnicity count median
## <chr>          <int> <dbl>
## 1 <NA>          21 140000
## 2 White (United States of America) 505 102880
## 3 Black or African American (United States of America) 62 91881.
## 4 Asian (United States of America) 59 90780
## 5 Prefer Not to Disclose (United States of America) 10 82140
## 6 Hispanic or Latino (United States of America) 33 82000
## 7 Two or More Races (United States of America) 14 79860

current_median_hourly_race <- filter(df, pay_rate_type == 'Hourly') %>% group_by(race_ethnicity)
current_median_hourly_race <- current_median_hourly_race %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)

12
median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_median_hourly_race)

## # A tibble: 4 x 3
## race_ethnicity  count  median
## <chr>           <int> <dbl>
## 1 White (United States of America) 107 32.7
## 2 Asian (United States of America)  18 27.3
## 3 Hispanic or Latino (United States of America) 12 25.6
## 4 Black or African American (United States of America) 95 25.2

current_age_race_salaried <- filter(df, pay_rate_type == 'Salaried') %>% group_by(race_ethnicity) current_age_race_salaried %>% summarise(
  median_age = median(age)
)

## # A tibble: 9 x 2
## race_ethnicity  median_age
## <chr>             <dbl>
## 1 American Indian or Alaska Native (United States of America) 49.5
## 2 Asian (United States of America) 33
## 3 Black or African American (United States of America) 41.5
## 4 Hispanic or Latino (United States of America) 37
## 5 Native Hawaiian or Other Pacific Islander (United States of A~ 43
## 6 Prefer Not to Disclose (United States of America) 31.5
## 7 Two or More Races (United States of America) 28
## 8 White (United States of America) 39
## 9 <NA> 36

current_age_race_hourly <- filter(df, pay_rate_type == 'Hourly') %>% group_by(race_ethnicity) current_age_race_hourly %>% summarise(
  median_age = median(age)
)

## # A tibble: 8 x 2
## race_ethnicity  median_age
## <chr>             <dbl>
## 1 American Indian or Alaska Native (United States of America) 53.5
## 2 Asian (United States of America) 32
## 3 Black or African American (United States of America) 47
## 4 Hispanic or Latino (United States of America) 29.5
## 5 Prefer Not to Disclose (United States of America) 30
## 6 Two or More Races (United States of America) 26.5
## 7 White (United States of America) 39
## 8 <NA> 31

Employee gender x race/ethnicity

current_employee_race_gender <- df %>% group_by(race_ethnicity, gender) current_employee_race_gender %>% summarise(
  count = length(current_base_pay)
)
suppress(current_employee_race_gender)
```r
## # A tibble: 14 x 3
## # Groups: race_ethnicity [7]
##  race_ethnicity gender count
## <chr>        <chr> <int>
## 1 Asian (United States of America) Female 53
## 2 Asian (United States of America) Male 24
## 3 Black or African American (United States of America) Female 80
## 4 Black or African American (United States of America) Male 77
## 5 Hispanic or Latino (United States of America) Female 24
## 6 Hispanic or Latino (United States of America) Male 21
## 7 Prefer Not to Disclose (United States of America) Female 6
## 8 Prefer Not to Disclose (United States of America) Male 8
## 9 Two or More Races (United States of America) Female 12
## 10 Two or More Races (United States of America) Male 6
## 11 White (United States of America) Female 318
## 12 White (United States of America) Male 294
## 13 <NA> Female 11
## 14 <NA> Male 11

current_salaried_race_gender <- filter(df, pay_rate_type == 'Salaried') %>% group_by(race_ethnicity, gender) %>% summarise(count = length(current_base_pay)),
suppress(current_salaried_race_gender)

## # A tibble: 7 x 3
## # Groups: race_ethnicity [4]
##  race_ethnicity gender count
## <chr>        <chr> <int>
## 1 Asian (United States of America) Female 11
## 2 Asian (United States of America) Male 11
```
## 2 Asian (United States of America) Male 7
## 3 Black or African American (United States of America) Female 49
## 4 Black or African American (United States of America) Male 46
## 5 Hispanic or Latino (United States of America) Female 8
## 6 White (United States of America) Female 63
## 7 White (United States of America) Male 44

current_median_race_gender <- filter(df, pay_rate_type == 'Salaried') %>% group_by(race_ethnicity, gender) 
  %>% summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )
suppress(current_median_race_gender)

## # A tibble: 14 x 4
## # Groups: race_ethnicity [7]
## race_ethnicity gender count median
## <chr> <chr> <int> <dbl>
## 1 Asian (United States of America) Female 42 9.11e4
## 2 Asian (United States of America) Male 17 9.04e4
## 3 Black or African American (United States of America) Female 31 8.78e4
## 4 Black or African American (United States of America) Male 31 9.99e4
## 5 Hispanic or Latino (United States of America) Female 16 8.02e4
## 6 Hispanic or Latino (United States of America) Male 17 9.08e4
## 7 Prefer Not to Disclose (United States of America) Female 5 7.30e4
## 8 Prefer Not to Disclose (United States of America) Male 5 8.83e4
## 9 Two or More Races (United States of America) Female 16 8.02e4
## 10 Two or More Races (United States of America) Male 5 9.49e4
## 11 White (United States of America) Female 255 9.58e4
## 12 White (United States of America) Male 250 1.11e5
## 13 <NA> Female 10 1.38e5
## 14 <NA> Male 11 1.40e5

current_median_hourly_race_gender <- filter(df, pay_rate_type == 'Hourly') %>% group_by(race_ethnicity, gender) 
  %>% summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )
suppress(current_median_hourly_race_gender)

## # A tibble: 7 x 4
## # Groups: race_ethnicity [4]
## race_ethnicity gender count median
## <chr> <chr> <int> <dbl>
## 1 Asian (United States of America) Female 11 28.3
## 2 Asian (United States of America) Male 7 26.3
## 3 Black or African American (United States of America) Female 49 26.8
## 4 Black or African American (United States of America) Male 46 23.2
## 5 Hispanic or Latino (United States of America) Female 8 28.2
## 6 White (United States of America) Female 63 33.5
## 7 White (United States of America) Male 44 31.0

Employee age
current_employee_age_5 <- df %>% group_by(age_group_5)
current_employee_age_5 <- current_employee_age_5 %>% summarise(
  count = length(current_base_pay)
)
suppress(current_employee_age_5)

## # A tibble: 10 x 2
## #  age_group_5 count
## <fct>     <int>
## 1 <25       59
## 2 25-29    171
## 3 30-34    139
## 4 35-39    125
## 5 40-44    98
## 6 45-49    79
## 7 50-54    106
## 8 55-59    84
## 9 60-64    56
##10 65+      33
terminated_employee_age_5 <- df2 %>% group_by(age_group_5)
terminated_employee_age_5 <- terminated_employee_age_5 %>% summarise(
  count = length(current_base_pay)
)
suppress(terminated_employee_age_5)

## # A tibble: 10 x 2
## #  age_group_5 count
## <fct>     <int>
## 1 <25       7
## 2 25-29    118
## 3 30-34    115
## 4 35-39    56
## 5 40-44    53
## 6 45-49    40
## 7 50-54    33
## 8 55-59    42
## 9 60-64    29
##10 65+      44
current_employee_age_10 <- df %>% group_by(age_group_10)
current_employee_age_10 <- current_employee_age_10 %>% summarise(
  count = length(current_base_pay)
)
suppress(current_employee_age_10)

## # A tibble: 6 x 2
## #  age_group_10 count
## <fct>    <int>
## 1 <25     59
## 2 25-34   310
## 3 35-44   223
## 4 45-54   185
## 5 55-64   140
## 6 65+     33
terminated_employee_age_10 <- df2 %>% group_by(age_group_10) %>% summarise(count = length(current_base_pay))

suppress(terminated_employee_age_10)

## # A tibble: 6 x 2
## #  age_group_10 count
## <fct>      <int>
## 1 <25        7
## 2 25-34     233
## 3 35-44     109
## 4 45-54     73
## 5 55-64     71
## 6 65+       44

current_median_age_5 <- filter(df, pay_rate_type == 'Salaried') %>% group_by(age_group_5) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))

suppress(current_median_age_5)

## # A tibble: 10 x 3
## #  age_group_5 count median
## <fct>      <int>  <dbl>
## 1 <25       34     64640
## 2 25-29     126    80000
## 3 30-34     119    92500
## 4 35-39     104   105301.
## 5 40-44     72     125924.
## 6 45-49     56     99502.
## 7 50-54     80     110845.
## 8 55-59     61    139717.
## 9 60-64     38    113134.
## 10 65+      17    153061

current_median_hourly_age_5 <- filter(df, pay_rate_type == 'Hourly') %>% group_by(age_group_5) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))

suppress(current_median_hourly_age_5)

## # A tibble: 10 x 3
## #  age_group_5 count median
## <fct>      <int> <dbl>
## 1 <25       25     25.6
## 2 25-29     45     30.8
## 3 30-34     20     30.6
## 4 35-39     21     31.2
## 5 40-44     26     29.5
## 6 45-49     23     31.3
## 7 50-54     26     27.2
## 8 55-59     23     27.0

17
## A tibble: 6 x 3
## age_group_10 count median
## <fct> <int> <dbl>
## 1 <25 34 64640
## 2 25-34 245 85500
## 3 35-44 176 115118.
## 4 45-54 136 108202.
## 5 55-64 99 127059.
## 6 65+ 17 153061

## A tibble: 6 x 3
## age_group_10 count median
## <fct> <int> <dbl>
## 1 <25 25 25.6
## 2 25-34 65 30.8
## 3 35-44 47 30.8
## 4 45-54 49 28.3
## 5 55-64 41 26.5
## 6 65+ 16 27.3

Employee department

## A tibble: 2 x 2
## dept count
## <chr> <int>
## 1 News 670
## 2 Commercial 280
## A tibble: 9 x 2
##  department count
##  <chr>   <int>
## 1 News     632
## 2 Client Solutions  164
## 3 Circulation    49
## 4 Editorial      38
## 5 Finance        31
## 6 Marketing      11
## 7 WP News Media Services  9
## 8 Production      6
## 9 Public Relations   5

`current_employee_dept_salary <- filter(df, pay_rate_type == 'Salaried') %>% group_by(dept) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))`

`suppress_median(current_employee_dept_salary)`

## A tibble: 2 x 3
##  dept   count  median
##  <chr>   <int>    <dbl>
## 1 News   574 104670.
## 2 Commercial 133  86105.

`current_employee_department_salary <- filter(df, pay_rate_type == 'Salaried') %>% group_by(department) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))`

`suppress_median(current_employee_department_salary)`

## A tibble: 7 x 3
##  department  count  median
##  <chr>       <int>   <dbl>
## 1 Editorial   33  105000
## 2 News        541  104560.
## 3 Finance     8  90576.
## 4 WP News Media Services 9  86105.
## 5 Client Solutions 102  85634.
## 6 Marketing   7  81196.
## 7 Production   5  71665.

`current_employee_dept_hourly <- filter(df, pay_rate_type == 'Hourly') %>% group_by(dept) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))`

`suppress_median(current_employee_dept_hourly)`

## A tibble: 2 x 3
##  dept     count median
##  <chr>   <int>   <dbl>
## 1 News    96  33.0
## 2 Commercial 147  26.3
current_employee_department_hourly <- filter(df, pay_rate_type == 'Hourly') %>% group_by(department)
current_employee_department_hourly %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_employee_department_hourly)

Employee cost center

current_employee_desk <- df %>% group_by(desk)
current_employee_desk %>% summarise(
  count = length(current_base_pay)
)
suppress_count(current_employee_desk)

current_employee_cost_center <- df %>% group_by(cost_center_current)
current_employee_cost_center %>% summarise(
  count = length(current_base_pay)
)
suppress_count(current_employee_cost_center)
## A tibble: 50 x 2
## cost_center_current count
## <chr> <int>
## 1 112300 Local Politics and Government 70
## 2 113200 National Politics and Government 63
## 3 110652 News Video - General 50
## 4 110015 Sports Main 48
## 5 110601 Multiplatform Desk 42
## 6 110300 Style 39
## 7 119065 Dispatch Operations (Night Circulation) 39
## 8 113210 Economy and Business 38
## 9 115000 Editorial Administration 38
## 10 110605 Presentation 24
## # ... with 40 more rows

current_employee_desk_salary <- filter(df, pay_rate_type == 'Salaried') %>%
group_by(desk)
current_employee_desk_salary <- current_employee_desk_salary %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_employee_desk_salary)

## A tibble: 19 x 3
## desk count median
## <chr> <int> <dbl>
## 1 National 106 149520.
## 2 Foreign 25 135000
## 3 Financial 38 133510.
## 4 Investigative 13 129780
## 5 Style 45 107171.
## 6 Local 65 105780
## 7 Editorial 33 105000
## 8 Graphics 15 100780
## 9 Universal Desk 8 100444.
## 10 Sports 37 100000
## 11 Outlook 6 99938.
## 12 Audio 7 92000
## 13 Design 45 88065.
## 14 Operations 6 87890
## 15 non-newsroom 133 86105.
## 16 Multiplatform 26 86104
## 17 Video 46 84250
## 18 Audience Development and Engagement 16 83530
## 19 Emerging News Products 30 75000

current_employee_cost_center_salary <- filter(df, pay_rate_type == 'Salaried') %>%
group_by(cost_center_current)
current_employee_cost_center_salary <- current_employee_cost_center_salary %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_employee_cost_center_salary)

## A tibble: 35 x 3
## cost_center_current count median
## <chr> <int> <dbl>
## 1 112300 Local Politics and Government 70 149520.
## 2 113200 National Politics and Government 63 135000
## 3 110652 News Video - General 50 133510.
## 4 110015 Sports Main 48 129780
## 5 110601 Multiplatform Desk 42 107171.
## 6 110300 Style 39 105780
## 7 119065 Dispatch Operations (Night Circulation) 39 100444.
## 8 113210 Economy and Business 38 100000
## 9 115000 Editorial Administration 38 99938.
## 10 110605 Presentation 24 92000
## 11 119065 Dispatch Operations (Night Circulation) 39 88065.
## 12 113210 Economy and Business 38 87890
## 13 115000 Editorial Administration 38 86105.
## 14 110605 Presentation 24 86104
## 15 110300 Style 39 84250
## 16 110652 News Video - General 50 83530
## 17 112300 Local Politics and Government 70 75000

21
```r
# A tibble: 11 x 3
#  desk      count   median
#  <chr>    <int>    <dbl>
#1  Audio      6    39.7
#2 Universal Desk 8    38.7
#3 Audience Development and Engagement 7    37.6
#4 Multiplatform 16   34.1
#5 Editorial 5    32.3
#6 National   12   31.7
#7 Local      5    26.5
#8 non-newsroom 147   26.3
#9 Style      9    21.8
#10 Sports    11    20.9
#11 Operations 7    15.6
```

```r
current_employee_cost_center_hourly <- filter(df, pay_rate_type == 'Hourly') %>%
group_by(cost_center_current) %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_employee_cost_center_hourly)
```

```r
# A tibble: 17 x 3
#  cost_center_current     count median
#  <chr>                    <int> <dbl>
#1  110620 News Audio       6  39.7
#2  110600 Universal Desk   8  38.7
#3  110610 Audience Development and Engagement 7  37.6
#4  129100 Community        5  35.0
#5  110601 Multiplatform Desk 16  34.1
#6  115000 Editorial Administration 5  32.3
#7  126060 Circulation Accounting 9  30.5
#8  113200 National Politics and Government 8  30.5
#9  126020 Revenue Administration 14  28.8
#10 117210 Production Creative 5  28.1
#11 112300 Local Politics and Government 5  26.5
```
Employee years of service

```r
current_employee_yos <- df %>%
group_by(years_of_service_grouped)
current_employee_yos <- current_employee_yos %>%
  summarise(
    count = length(current_base_pay)
  )
suppress(current_employee_yos)

# # A tibble: 8 x 2
# # years_of_service_grouped count
# <fct> <int>
# 1 0 138
# 2 1-2 223
# 3 3-5 195
# 4 6-10 109
# 5 11-15 80
# 6 16-20 102
# 7 21-25 46
# 8 25+ 57

terminated_employee_yos <- df2 %>%
group_by(years_of_service_grouped)
terminated_employee_yos <- terminated_employee_yos %>%
  summarise(
    count = length(current_base_pay)
  )
suppress(terminated_employee_yos)

# # A tibble: 8 x 2
# # years_of_service_grouped count
# <fct> <int>
# 1 0 8
# 2 1-2 78
# 3 3-5 197
# 4 6-10 119
# 5 11-15 52
# 6 16-20 44
# 7 21-25 12
# 8 25+ 29

current_employee_yos_salary <- filter(df, pay_rate_type == 'Salaried') %>%
group_by(years_of_service_grouped)
current_employee_yos_salary <- current_employee_yos_salary %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )
suppress(current_employee_yos_salary)

# # A tibble: 8 x 3
# # years_of_service_grouped count  median
# <fct> <int>   <dbl>
# 1 0 138 24.7
# 2 1-2 223 24.3
# 3 3-5 195 22.4
# 4 6-10 109 20.9
# 5 11-15 80 20.5
# 6 16-20 102 15.6
```
```r
# # A tibble: 8 x 3
# years_of_service_grouped count median
# <fct> <int> <dbl>
# 1 0 42 27.7
# 2 1-2 59 31.7
# 3 3-5 23 27.0
# 4 6-10 34 29.2
# 5 11-15 24 32.4
# 6 16-20 28 27.8
# 7 21-25 14 31.1
# 8 25+ 19 26.8
```

```r
# # A tibble: 16 x 3
# # Groups: years_of_service_grouped [8]
# years_of_service_grouped gender count
# <fct> <chr> <int>
# 1 0 Female 82
# 2 0 Male 56
# 3 1-2 Female 132
# 4 1-2 Male 91
# 5 3-5 Female 96
# 6 3-5 Male 99
# 7 6-10 Female 51
# 8 6-10 Male 58
# 9 11-15 Female 41
# 10 11-15 Male 39
# 11 16-20 Female 48
# 12 16-20 Male 54
# 13 21-25 Female 25
# 14 21-25 Male 21
# 15 25+ Female 32
# 16 25+ Male 25
```
current_employee_yos_gender_salary <- filter(df, pay_rate_type == 'Salaried') %>% group_by(years_of_service_grouped, gender)
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_employee_yos_gender_salary)

## A tibble: 16 x 4
## Groups: years_of_service_grouped [8]
##   years_of_service_grouped gender count median
##    <fct>       <chr>  <int>  <dbl>
## 1    0          Female  61  80000
## 2    0           Male  35 100000
## 3    1-2         Female  96  85780
## 4    1-2         Male  68  96738.
## 5    3-5         Female  88  89725.
## 6    3-5         Male  84  95265.
## 7    6-10        Female  38  99500.
## 8    6-10        Male  37 117844.
## 9    9 11-15    Female  28  98142.
##10   10 11-15    Male  28 126911.
##11   11 16-20   Female  31 121140
##12   12 16-20   Male  43 127059.
##13   13 21-25  Female  13 134780
##14   14 21-25  Male   19  99012.
##15   15 25+     Female  15 139831.
##16   16 25+     Male  23 127476.

current_employee_yos_gender_hourly <- filter(df, pay_rate_type == 'Hourly') %>% group_by(years_of_service_grouped, gender)
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_employee_yos_gender_hourly)

## A tibble: 14 x 4
## Groups: years_of_service_grouped [8]
##   years_of_service_grouped gender count median
##    <fct>       <chr>  <int>  <dbl>
## 1    0          Female  21  29.2
## 2    0           Male  21  22.0
## 3    1-2         Female  36  31.9
## 4    1-2         Male  23  26.0
## 5    3-5         Female  8  34.8
## 6    3-5         Male  15  23.0
## 7    6-10        Female  13  30.8
## 8    6-10        Male  21  25.2
## 9    9 11-15    Female  13  34.7
##10   10 11-15    Male  11  29.9
##11   11 16-20   Female  17  25.1
##12   12 16-20   Male  11  30.2
##13   13 21-25  Female  12  30.3
##14   14 25+     Female  17  27.7
current_employee_yos_race <- df %>% group_by(years_of_service_grouped, race_ethnicity)
current_employee_yos_race <- current_employee_yos_race %>% summarise(
  count = length(current_base_pay)
)
suppress(current_employee_yos_race)

## # A tibble: 31 x 3
## # Groups: years_of_service_grouped [8]
## #   years_of_service_grouped race_ethnicity count
##   <fct>                        <chr>        <int>
## 1 0                             Asian (United States of America) 15
## 2 0                             Black or African American (United States of America) 20
## 3 0                             Hispanic or Latino (United States of America) 10
## 4 0                             Prefer Not to Disclose (United States of America) 8
## 5 0                             Two or More Races (United States of America) 6
## 6 0                             White (United States of America) 77
## 7 1-2                           Asian (United States of America) 20
## 8 1-2                           Black or African American (United States of America) 30
## 9 1-2                           Hispanic or Latino (United States of America) 12
## 10 1-2                          Two or More Races (United States of America) 6
## # ... with 21 more rows

current_employee_yos_race_salary <- filter(df, pay_rate_type == 'Salaried') %>% group_by(years_of_service_grouped, race_ethnicity)
current_employee_yos_race_salary <- current_employee_yos_race_salary %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_employee_yos_race_salary)

## # A tibble: 25 x 4
## # Groups: years_of_service_grouped [8]
## #   years_of_service_grouped race_ethnicity count median
##   <fct>                        <chr>        <int> <dbl>
## 1 0                             Asian (United States of America) 11 77000
## 2 0                             Black or African American (United States of America) 5 87000
## 3 0                             Hispanic or Latino (United States of America) 5 75000
## 4 0                             White (United States of America) 65 90000
## 5 1-2                           Asian (United States of America) 16 87780
## 6 1-2                           Black or African American (United States of America) 12 89780
## 7 1-2                           Hispanic or Latino (United States of America) 7 82000
## 8 1-2                           Two or More Races (United States of America) 5 68000
## 9 1-2                           White (United States of America) 115 92780
## 10 1-2                          <NA> 5 140280
## # ... with 15 more rows

current_employee_yos_race_hourly <- filter(df, pay_rate_type == 'Hourly') %>% group_by(years_of_service_grouped, race_ethnicity)
current_employee_yos_race_hourly <- current_employee_yos_race_hourly %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_employee_yos_race_hourly)

## # A tibble: 18 x 4
## # Groups: years_of_service_grouped [8]
## #   years_of_service_grouped race_ethnicity count median
##   <fct>                        <chr>        <int> <dbl>
### Employee performance evaluations

```r
fifteen <- rbind(fifteen1, fifteen2)
fifteenrating_gender <- fifteen %>% group_by(gender)
fifteenrating_gender %>% summarise(
  median = median(performance_rating, na.rm = TRUE)
)
```

```
## # A tibble: 2 x 2
## gender median
## <chr>   <dbl>
## 1 Female 3.4
## 2 Male   3.4
```

```r
sixteen <- rbind(sixteen1, sixteen2)
sixteenrating_gender <- sixteen %>% group_by(gender)
sixteenrating_gender %>% summarise(
  median = median(performance_rating, na.rm = TRUE)
)
```

```
## # A tibble: 2 x 2
## gender median
## <chr>   <dbl>
## 1 Female 3.3
## 2 Male   3.3
```

```r
seventeen <- rbind(seventeen1, seventeen2)
seventeenrating_gender <- seventeen %>% group_by(gender)
seventeenrating_gender %>% summarise(
  median = median(performance_rating, na.rm = TRUE)
)
```

```
## # A tibble: 2 x 2
## gender median
## <chr>   <dbl>
## 1 Female 3.4
## 2 Male   3.4
```
# 2 Male 3.4

eighteen <- rbind(eighteen1, eighteen2)
eighteenrating_gender <- eighteen %>% group_by(gender)
eighteenrating_gender %>% summarise(
  median = median(performance_rating, na.rm = TRUE)
)

## A tibble: 2 x 2
##  gender median
##  <chr>    <dbl>
## 1 Female   3.4
## 2 Male     3.4

fifteen <- rbind(fifteen1, fifteen2)
fifteenrating_race_ethnicity <- fifteen %>% group_by(race_ethnicity)
fifteenrating_race_ethnicity %>% summarise(
  median = median(performance_rating, na.rm = TRUE)
)

## A tibble: 9 x 2
## race_ethnicity median
## <chr>           <dbl>
## 1 American Indian or Alaska Native (United States of America) 3.5
## 2 Asian (United States of America) 3.4
## 3 Black or African American (United States of America) 3.2
## 4 Hispanic or Latino (United States of America) 3.2
## 5 Native Hawaiian or Other Pacific Islander (United States of America) 3.25
## 6 Prefer Not to Disclose (United States of America) 3.3
## 7 Two or More Races (United States of America) 3.3
## 8 White (United States of America) 3.4
## 9 <NA> 3.7

sixteen <- rbind(sixteen1, sixteen2)
sixteenrating_race_ethnicity <- sixteen %>% group_by(race_ethnicity)
sixteenrating_race_ethnicity %>% summarise(
  median = median(performance_rating, na.rm = TRUE)
)

## A tibble: 9 x 2
## race_ethnicity median
## <chr>           <dbl>
## 1 American Indian or Alaska Native (United States of America) 3.25
## 2 Asian (United States of America) 3.35
## 3 Black or African American (United States of America) 3.2
## 4 Hispanic or Latino (United States of America) 3.1
## 5 Native Hawaiian or Other Pacific Islander (United States of America) 3.7
## 6 Prefer Not to Disclose (United States of America) 3.3
## 7 Two or More Races (United States of America) 3.2
## 8 White (United States of America) 3.4
## 9 <NA> 3.75

seventeen <- rbind(seventeen1, seventeen2)
seventeenrating_race_ethnicity <- seventeen %>% group_by(race_ethnicity)
seventeenrating_race_ethnicity %>% summarise(
  median = median(performance_rating, na.rm = TRUE)
)

## A tibble: 9 x 2
## race_ethnicity median
## <chr>           <dbl>
## 1 American Indian or Alaska Native (United States of America) 3.75
## 2 Asian (United States of America) 3.35
## 3 Black or African American (United States of America) 3.2
## 4 Hispanic or Latino (United States of America) 3.1
## 5 Native Hawaiian or Other Pacific Islander (United States of America) 3.7
## 6 Prefer Not to Disclose (United States of America) 3.3
## 7 Two or More Races (United States of America) 3.2
## 8 White (United States of America) 3.4
## 9 <NA> 3.75

28
```
# A tibble: 9 x 2
# race_ethnicity median
# <chr>           <dbl>
## 1 American Indian or Alaska Native (United States of America) 3.55
## 2 Asian (United States of America) 3.4
## 3 Black or African American (United States of America) 3.2
## 4 Hispanic or Latino (United States of America) 3.3
## 5 Native Hawaiian or Other Pacific Islander (United States of America) 3.4
## 6 Prefer Not to Disclose (United States of America) 3.5
## 7 Two or More Races (United States of America) 3.4
## 8 White (United States of America) 3.5
## 9 <NA> 3.6

eighteen <- rbind(eighteen1,eighteen2)
eighteenrating_race_ethnicity <— eighteen %>% group_by(race_ethnicity)  
eighteenrating_race_ethnicity %>% summarise(
  median = median(performance_rating, na.rm = TRUE)
)

# A tibble: 9 x 2
# race_ethnicity median
# <chr>           <dbl>
## 1 American Indian or Alaska Native (United States of America) 3.55
## 2 Asian (United States of America) 3.4
## 3 Black or African American (United States of America) 3.2
## 4 Hispanic or Latino (United States of America) 3.3
## 5 Native Hawaiian or Other Pacific Islander (United States of America) 3.4
## 6 Prefer Not to Disclose (United States of America) 3.5
## 7 Two or More Races (United States of America) 3.4
## 8 White (United States of America) 3.5
## 9 <NA> 3.6

fifteen <- rbind(fifteen1,fifteen2)
fifteenrating_gender_race <— fifteen %>% group_by(race_ethnicity, gender)  
fifteenrating_gender_race %>% summarise(
  median = median(performance_rating, na.rm = TRUE)
)

# A tibble: 18 x 3
# Groups: race_ethnicity [9]
# race_ethnicity gender median
# <chr>       <chr>  <dbl>
## 1 American Indian or Alaska Native (United States of America) Female 3.5
## 2 American Indian or Alaska Native (United States of America) Male 3.4
## 3 Asian (United States of America) Female 3.4
## 4 Asian (United States of America) Male 3.5
## 5 Black or African American (United States of America) Female 3.2
## 6 Black or African American (United States of America) Male 3
## 7 Hispanic or Latino (United States of America) Female 3.3
## 8 Hispanic or Latino (United States of America) Male 3.2
## 9 Native Hawaiian or Other Pacific Islander (United States of America) Female 3.2
## 10 Native Hawaiian or Other Pacific Islander (United States of America) Male 3.3
## 11 Prefer Not to Disclose (United States of America) Female 3.3
## 12 Prefer Not to Disclose (United States of America) Male NA
## 13 Two or More Races (United States of America) Female 3.3
## 14 Two or More Races (United States of America) Male 2.75
```

# A tibble: 18 x 3
# Groups: race_ethnicity [9]
   race_ethnicity gender median
   <chr>        <chr>  <dbl>
1 American Indian or Alaska Native (United States of Americ~ Female  3.3
2 American Indian or Alaska Native (United States of Americ~ Male   3.2
3 Asian (United States of America) Female  3.4
4 Asian (United States of America) Male   3.3
5 Black or African American (United States of America) Female  3.25
6 Black or African American (United States of America) Male   3.15
7 Hispanic or Latino (United States of America) Female  3.15
8 Hispanic or Latino (United States of America) Male   3.1
9 Native Hawaiian or Other Pacific Islander (United States ~ Female  4.1
10 Native Hawaiian or Other Pacific Islander (United States ~ Male   3.3
11 Prefer Not to Disclose (United States of America) Female  3.3
12 Prefer Not to Disclose (United States of America) Male   NA
13 Two or More Races (United States of America) Female  3.2
14 Two or More Races (United States of America) Male   2.7
15 White (United States of America) Female  3.4
16 White (United States of America) Male   3.4
17 <NA> Female  3.6
18 <NA> Male   NA

# A tibble: 18 x 3
# Groups: race_ethnicity [9]
   race_ethnicity gender median
   <chr>        <chr>  <dbl>
1 American Indian or Alaska Native (United States of Americ~ Female  3.7
2 American Indian or Alaska Native (United States of Americ~ Male   3.1
3 Asian (United States of America) Female  3.4
4 Asian (United States of America) Male   3.3
5 Black or African American (United States of America) Female  3.2
6 Black or African American (United States of America) Male   3.1
7 Hispanic or Latino (United States of America) Female  3.3
8 Hispanic or Latino (United States of America) Male   3.3
9 Native Hawaiian or Other Pacific Islander (United States ~ Female  4.0
10 Native Hawaiian or Other Pacific Islander (United States ~ Male   3.0
11 Prefer Not to Disclose (United States of America) Female  3.5
12 Prefer Not to Disclose (United States of America) Male   3.2
## 13 Two or More Races (United States of America)  Female  3.25
## 14 Two or More Races (United States of America)  Male  3.5
## 15 White (United States of America)  Female  3.4
## 16 White (United States of America)  Male  3.4
## 17 <NA>  Female  3.65
## 18 <NA>  Male  3.5

eighteen <- rbind(eighteen1,eighteen2)
eighteenrating_gender_race <- eighteen %>% group_by(race_ethnicity, gender)
eighteenrating_gender_race %>% summarise(
  median = median(performance_rating, na.rm = TRUE)
)

## A tibble: 18 x 3
## Groups: race_ethnicity [9]
## race_ethnicity gender median
## <chr> <chr> <dbl>
## 1 American Indian or Alaska Native (United States of Americ~ Female  3.7
## 2 American Indian or Alaska Native (United States of Americ~ Male  3.2
## 3 Asian (United States of America)  Female  3.4
## 4 Asian (United States of America)  Male  3.4
## 5 Black or African American (United States of America)  Female  3.3
## 6 Black or African American (United States of America)  Male  3.3
## 7 Hispanic or Latino (United States of America)  Female  3.3
## 8 Hispanic or Latino (United States of America)  Male  3.3
## 9 Native Hawaiian or Other Pacific Islander (United States ~ Female NA
## 10 Native Hawaiian or Other Pacific Islander (United States ~ Male  3.4
## 11 Prefer Not to Disclose (United States of America)  Female  3.55
## 12 Prefer Not to Disclose (United States of America)  Male  3.3
## 13 Two or More Races (United States of America)  Female  3.3
## 14 Two or More Races (United States of America)  Male  3.35
## 15 White (United States of America)  Female  3.4
## 16 White (United States of America)  Male  3.5
## 17 <NA>  Female  3.6
## 18 <NA>  Male  3.4

Employee pay changes

reason_for_change <- reason_for_change_combined %>% group_by(business_process_reason)
reason_for_change <- reason_for_change %>% summarise(
  count = length(business_process_reason)
)
suppress_count(reason_for_change)

## A tibble: 19 x 2
## <chr> <int>
## 1 <NA> 16810
## 2 Request Compensation Change > Adjustment > Contract Increase 2451
## 3 Merit > Performance > Annual Performance Appraisal 1729
## 4 Data Change > Data Change > Change Job Details 673
## 5 Transfer > Transfer > Move to another Manager 533
## 6 Request Compensation Change > Adjustment > Change Plan Assignment 435
## 7 Request Compensation Change > Adjustment > Market Adjustment 384
## 8 Promotion > Promotion > Promotion 359
reason_for_change_gender <- reason_for_change_combined %>% group_by(business_process_reason, gender) 
  %>% summarise(count = length(business_process_reason)) 
suppress_count(reason_for_change_gender)

reason_for_change_race <- reason_for_change_combined %>% group_by(business_process_reason, race_ethnicity) 
  %>% summarise(count = length(business_process_reason)) 
suppress_count(reason_for_change_race)

# A tibble: 34 x 3
# Groups: business_process_reason [19]


# A tibble: 83 x 3
# Groups: business_process_reason [18]


# ... with 73 more rows

32
reason_for_change_race_gender <- reason_for_change_combined %>% group_by(business_process_reason, race_ethnicity, gender) 
reason_for_change_race_gender <- reason_for_change_race_gender %>% summarise(
  count = length(business_process_reason)
)
suppress_count(reason_for_change_race_gender)

## # A tibble: 122 x 4
## # Groups: business_process_reason, race_ethnicity [70]
## business_process_reason race_ethnicity gender count
## <chr> <chr> <chr> <int>
## 1 <NA> White (United States of A~ Female 5391
## 2 <NA> White (United States of A~ Male 4836
## 3 <NA> Black or African American~ Male 1827
## 4 <NA> Black or African American~ Female 1680
## 5 <NA> Asian (United States of A~ Female 1022
## 6 Request Compensation Change > A~ White (United States of A~ Female 794
## 7 Request Compensation Change > A~ White (United States of A~ Male 762
## 8 Merit > Performance > Annual Pe~ White (United States of A~ Male 564
## 9 Merit > Performance > Annual Pe~ White (United States of A~ Female 545
## 10 <NA> Hispanic or Latino (Unite~ Female 414
## # ... with 112 more rows

News

Gender

current_news_gender_salaried <- news_salaried %>% group_by(gender)
current_news_gender_salaried <- current_news_gender_salaried %>% summarise(
  count = length(current_base_pay)
)
suppress(current_news_gender_salaried)

## # A tibble: 2 x 2
## gender count
## <chr> <int>
## 1 Female 284
## 2 Male 290

current_news_gender_hourly <- news_hourly %>% group_by(gender)
current_news_gender_hourly <- current_news_gender_hourly %>% summarise(
  count = length(current_base_pay)
)
suppress(current_news_gender_hourly)

## # A tibble: 2 x 2
## gender count
## <chr> <int>
## 1 Female 63
## 2 Male 33

current_news_gender_salaried_median <- news_salaried %>% group_by(gender)
current_news_gender_salaried_median <- current_news_gender_salaried_median %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
```r
suppress(current_news_gender_salaried_median)

## # A tibble: 2 x 3
## #  gender count median
## # <chr> <int> <dbl>
## 1 Female 284 95595.
## 2 Male 290 116065.

current_news_gender_hourly_median <- news_hourly %>%
  group_by(gender) %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )
suppress(current_news_gender_hourly_median)

## # A tibble: 2 x 3
## #  gender count median
## # <chr> <int> <dbl>
## 1 Female 63 32.8
## 2 Male 33 33.3

current_news_gender_age_salaried <- news_salaried %>%
  group_by(gender) %>%
  summarise(median_age = median(age))

## # A tibble: 2 x 2
## #  gender median_age
## # <chr> <dbl>
## 1 Female 35
## 2 Male 41

current_news_gender_age_hourly <- news_hourly %>%
  group_by(gender) %>%
  summarise(median_age = median(age))

## # A tibble: 2 x 2
## #  gender median_age
## # <chr> <dbl>
## 1 Female 31
## 2 Male 36

current_news_gender_age_5_salary <- news_salaried %>%
  group_by(age_group_5, gender) %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )
suppress(current_news_gender_age_5_salary)

## # A tibble: 20 x 4
## # Groups: age_group_5 [10]
## #  age_group_5 gender count median
## # <fct> <chr> <int> <dbl>
## 1 <25 Female 19 64280
## 2 <25 Male 5 72000
## 3 25-29 Female 60 80000
```
## 4 25-29 Male 31 85500
## 5 30-34 Female 57 87000
## 6 30-34 Male 46 97828.
## 7 35-39 Female 38 98892.
## 8 35-39 Male 48 116030
## 9 40-44 Female 22 133200.
## 10 40-44 Male 41 125000
## 11 45-49 Female 29 108864.
## 12 45-49 Male 41 126280.
## 13 50-54 Female 22 145655.
## 14 50-54 Male 29 147780
## 15 55-59 Female 20 117295.
## 16 55-59 Male 23 99725
## 17 60-64 Female 12 129325.
## 18 60-64 Male 16 131217.
## 19 65+ Female 5 157095.
## 20 65+ Male 10 156260.

current_news_gender_age_5_hourly <- news_hourly %>% group_by(age_group_5, gender)
current_news_gender_age_5_hourly <- current_news_gender_age_5_hourly %>%% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_news_gender_age_5_hourly)

## # A tibble: 9 x 4
## # Groups: age_group_5 [8]
## # age_group_5 gender count median
## <fct> <chr> <int> <dbl>
## 1 <25 Female 12 31.4
## 2 25-29 Female 17 31.2
## 3 25-29 Male 6 21.0
## 4 30-34 Male 7 33.7
## 5 35-39 Female 5 31.9
## 6 40-44 Female 5 41.4
## 7 45-49 Female 5 44.5
## 8 50-54 Female 6 40.2
## 9 55-59 Male 5 34.9

current_news_gender_age_10_salary <- news_salaried %>% group_by(age_group_10, gender)
current_news_gender_age_10_salary <- current_news_gender_age_10_salary %>%% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_news_gender_age_10_salary)

## # A tibble: 12 x 4
## # Groups: age_group_10 [6]
## # age_group_10 gender count median
## <fct> <chr> <int> <dbl>
## 1 <25 Female 19 64280
## 2 <25 Male 5 72000
## 3 25-34 Female 117 83147.
## 4 25-34 Male 77 92500
## 5 35-44 Female 60 105691.

35
## 6 35-44 Male 89 118785
## 7 45-54 Female 49 108864.
## 8 45-54 Male 64 117982.
## 9 55-64 Female 34 140424.
## 10 55-64 Male 45 146542.
## 11 65+ Female 5 157095.
## 12 65+ Male 10 156260.

current_news_gender_age_10_hourly <- news_hourly %>%
group_by(age_group_10, gender)
current_news_gender_age_10_hourly <- current_news_gender_age_10_hourly %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_news_gender_age_10_hourly)

## # A tibble: 8 x 4
## # Groups: age_group_10 [5]
## # age_group_10 gender count median
## <fct> <chr> <int> <dbl>
## 1 <25 Female 12 31.4
## 2 25-34 Female 21 31.2
## 3 25-34 Male 13 30.8
## 4 35-44 Female 10 33.1
## 5 35-44 Male 7 35.9
## 6 45-54 Female 11 41.4
## 7 55-64 Female 5 42.1
## 8 55-64 Male 7 33.4

current_news_gender_salaried_under_40 <- filter(news_salaried, age < 40) %>%
group_by(gender)
current_news_gender_salaried_under_40 <- current_news_gender_salaried_under_40 %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_news_gender_salaried_under_40)

## # A tibble: 2 x 3
## gender count median
## <chr> <int> <dbl>
## 1 Female 174 84030
## 2 Male 130 95890

current_news_gender_salaried_over_40 <- filter(news_salaried, age > 39) %>%
group_by(gender)
current_news_gender_salaried_over_40 <- current_news_gender_salaried_over_40 %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_news_gender_salaried_over_40)

## # A tibble: 2 x 3
## gender count median
## <chr> <int> <dbl>
## 1 Female 110 126000
## 2 Male 160 127765.
count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_news_gender_hourly_under_40)

## # A tibble: 2 x 3
## gender count median
## <chr> <int> <dbl>
## 1 Female 38 31.4
## 2 Male 18 32.0

current_news_gender_hourly_over_40 <- filter(news_hourly, age > 39) %>%
group_by(gender)
current_news_gender_hourly_over_40 <- current_news_gender_hourly_over_40 %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_news_gender_hourly_over_40)

## # A tibble: 2 x 3
## gender count median
## <chr> <int> <dbl>
## 1 Female 25 41.4
## 2 Male 15 33.4

Race and ethnicity

current_news_race_salaried <- news_salaried %>%
group_by(race_ethnicity)
current_news_race_salaried <- current_news_race_salaried %>%
summarise(
  count = length(current_base_pay)
)
suppress_count(current_news_race_salaried)

## # A tibble: 7 x 2
## race_ethnicity count
## <chr>          <int>
## 1 White (United States of America)  406
## 2 Black or African American (United States of America)  48
## 3 Asian (United States of America)  46
## 4 Hispanic or Latino (United States of America)  28
## 5 <NA>          21
## 6 Two or More Races (United States of America)  14
## 7 Prefer Not to Disclose (United States of America)  8

current_news_race_hourly <- news_hourly %>%
group_by(race_ethnicity)
current_news_race_hourly <- current_news_race_hourly %>%
summarise(
  count = length(current_base_pay)
)
suppress_count(current_news_race_hourly)

## # A tibble: 3 x 2
## race_ethnicity count
## <chr>          <int>
## 1 White (United States of America)  64
## 2 Black or African American (United States of America)  13
## 3 Asian (United States of America)  11
current_news_race_group_salaried <- news_salaried %>% group_by(race_grouping)
current_news_race_group_salaried <- current_news_race_group_salaried %>% summarize(
  count = length(current_base_pay)
)
suppress_count(current_news_race_group_salaried)

## A tibble: 3 x 2
## race_grouping  count
## <chr>        <int>
## 1 white        406
## 2 person of color 139
## 3 unknown      29

current_news_race_group_hourly <- news_hourly %>% group_by(race_grouping)
current_news_race_group_hourly <- current_news_race_group_hourly %>% summarize(
  count = length(current_base_pay)
)
suppress_count(current_news_race_group_hourly)

## A tibble: 2 x 2
## race_grouping  count
## <chr>        <int>
## 1 white       64
## 2 person of color 30

current_news_race_salaried_median <- news_salaried %>% group_by(race_ethnicity)
current_news_race_salaried_median <- current_news_race_salaried_median %>% summarize(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_news_race_salaried_median)

## A tibble: 7 x 3
## race_ethnicity  count  median
## <chr>            <int>   <dbl>
## 1 <NA>            21  140000
## 2 White (United States of America) 406 106212.
## 3 Black or African American (United States of America) 48  97276.
## 4 Asian (United States of America) 46  95205.
## 5 Hispanic or Latino (United States of America) 28  82890
## 6 Prefer Not to Disclose (United States of America) 8  82140
## 7 Two or More Races (United States of America) 14  79860

current_news_race_hourly_median <- news_hourly %>% group_by(race_ethnicity)
current_news_race_hourly_median <- current_news_race_hourly_median %>% summarize(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_news_race_hourly_median)

## A tibble: 3 x 3
## race_ethnicity  count  median
## <chr>            <int>   <dbl>
## 1 White (United States of America) 64  33.6
## 2 Asian (United States of America) 11  31.7
## 3 Black or African American (United States of America) 13  29.4
current_news_race_group_salaried_median <- news_salaried %>% group_by(race_grouping) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))

suppress_median(current_news_race_group_salaried_median)

## # A tibble: 3 x 3
## race_grouping  count  median
## <chr>       <int>     <dbl>
## 1 unknown   29 134780
## 2 white  406 106212.
## 3 person of color 139 92080

current_news_race_group_hourly_median <- news_hourly %>% group_by(race_grouping) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))

suppress_median(current_news_race_group_hourly_median)

## # A tibble: 2 x 3
## race_grouping  count  median
## <chr>       <int>     <dbl>
## 1 white  64 33.6
## 2 person of color 30 30.1

current_news_race_age_salaried <- news_salaried %>% group_by(race_ethnicity) %>% summarise(median_age = median(age))

## # A tibble: 9 x 2
## race_ethnicity median_age
## <chr>          <dbl>
## 1 American Indian or Alaska Native (United States of America) 49.5
## 2 Asian (United States of America) 33
## 3 Black or African American (United States of America) 39.5
## 4 Hispanic or Latino (United States of America) 37
## 5 Native Hawaiian or Other Pacific Islander (United States of America) 43
## 6 Prefer Not to Disclose (United States of America) 30.5
## 7 Two or More Races (United States of America) 28
## 8 White (United States of America) 40
## 9 <NA> 36

current_news_race_age_hourly <- news_hourly %>% group_by(race_ethnicity) %>% summarise(median_age = median(age))

## # A tibble: 8 x 2
## race_ethnicity median_age
## <chr>          <dbl>
## 1 American Indian or Alaska Native (United States of America) 69
## 2 Asian (United States of America) 36
## 3 Black or African American (United States of America) 28
## 4 Hispanic or Latino (United States of America) 26
## 5 Prefer Not to Disclose (United States of America) 23
## 6 Two or More Races (United States of America) 22.5
## 7 White (United States of America) 39.5
## 8 <NA> 31

```r
current_news_race_age_5_salary <- news_salaried %>%
group_by(age_group_5, race_ethnicity)

current_news_race_age_5_salary <- current_news_race_age_5_salary %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )
suppress(current_news_race_age_5_salary)
```

## A tibble: 25 x 4
## Groups: age_group_5 [10]
##   age_group_5 race_ethnicity count median
##   <fct>       <chr>       <int>  <dbl>
## 1 <25         Asian (United States of America) 5  65780
## 2 <25         White (United States of America) 12 65140
## 3 25-29       Asian (United States of America) 11  77000
## 4 25-29       Black or African American (United States of America) 6  81000
## 5 25-29       Two or More Races (United States of America) 6  75690
## 6 25-29       White (United States of America)  59  81757
## 7 30-34       Asian (United States of America) 10  95780
## 8 30-34       Black or African American (United States of America) 9  88133
## 9 30-34       Hispanic or Latino (United States of America) 6  80596
## 10 30-34      White (United States of America)  66  92640
## # ... with 15 more rows

```r
current_news_race_age_5_hourly <- news_hourly %>%
group_by(age_group_5, race_ethnicity)

current_news_race_age_5_hourly <- current_news_race_age_5_hourly %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )
suppress(current_news_race_age_5_hourly)
```

## A tibble: 10 x 4
## Groups: age_group_5 [9]
##   age_group_5 race_ethnicity count median
##   <fct>       <chr>       <int>  <dbl>
## 1 <25         White (United States of America)  7 18.5
## 2 25-29       Black or African American (United States of America) 8 30.1
## 3 25-29       White (United States of America) 11 30.8
## 4 30-34       White (United States of America)  9 33.7
## 5 35-39       White (United States of America)  5 34.7
## 6 40-44       White (United States of America)  7 41.4
## 7 45-49       White (United States of America)  5 44.5
## 8 50-54       White (United States of America)  6 40.2
## 9 55-59       White (United States of America)  6 33.9
## 10 60-64      White (United States of America)  5 38.8

```r
current_news_race_age_10_salary <- news_salaried %>%
group_by(age_group_10, race_ethnicity)

current_news_race_age_10_salary <- current_news_race_age_10_salary %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )
```
suppress(current_news_race_age_10_salary)

```r
## # A tibble: 21 x 4
## # Groups: age_group_10 [6]
## # age_group_10 race_ethnicity  count median
## <fct> <chr> <int> <dbl>
## 1 <25 Asian (United States of America) 5 6.58e4
## 2 <25 White (United States of America) 12 6.51e4
## 3 25-34 Asian (United States of America) 21 8.60e4
## 4 25-34 Black or African American (United States of America) 15 8.70e4
## 5 25-34 Hispanic or Latino (United States of America) 10 8.12e4
## 6 25-34 Prefer Not to Disclose (United States of America) 5 7.85e4
## 7 25-34 Two or More Races (United States of America) 9 7.64e4
## 8 25-34 White (United States of America) 125 8.60e4
## 9 25-34 <NA> 9 1.16e5
## # ... with 11 more rows
```

```r
current_news_race_age_10_hourly <- news_hourly %>%
group_by(age_group_10, race_ethnicity) %>%
summarise(count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE))
suppress(current_news_race_age_10_hourly)
```

```r
## # A tibble: 6 x 4
## # Groups: age_group_10 [5]
## # age_group_10 race_ethnicity  count median
## <fct> <chr> <int> <dbl>
## 1 <25 White (United States of America) 7 18.5
## 2 25-34 Black or African American (United States of America) 8 30.1
## 3 25-34 White (United States of America) 20 31.3
## 4 35-44 White (United States of America) 12 35.3
## 5 45-54 White (United States of America) 11 41.4
## 6 55-64 White (United States of America) 11 34.9
```

```r
current_news_race_group_age_5_salary <- news_salaried %>%
group_by(age_group_5, race_grouping) %>%
summarise(count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE))
suppress(current_news_race_group_age_5_salary)
```

```r
## # A tibble: 21 x 4
## # Groups: age_group_5 [10]
## # age_group_5 race_grouping  count median
## <fct> <chr> <int> <dbl>
## 1 <25 person of color 11 63780
## 2 <25 white 12 65140
## 3 25-29 person of color 27 80000
## 4 25-29 unknown 5 88280
## 5 25-29 white 59 81757.
## 6 30-34 person of color 28 86983.
## 7 30-34 unknown 9 108000
## 8 30-34 white 66 92640
```
## 9 35-39 person of color 23 99238.
## 10 35-39 white 61 105780
## # ... with 11 more rows

```r
current_news_race_group_age_5_hourly <- news_hourly %>%
  group_by(age_group_5, race_grouping) %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )
```

```r
# A tibble: 11 x 4
# Groups: age_group_5 [9]
## age_group_5 race_grouping count median
## <fct>      <chr>     <int> <dbl>
## 1 <25       person of color 6 29.5
## 2 <25       white       7 18.5
## 3 25-29     person of color 12 27.1
## 4 25-29     white       11 30.8
## 5 30-34     white       9 33.7
## 6 35-39     white       5 34.7
## 7 40-44     white       7 41.4
## 8 45-49     white       5 44.5
## 9 50-54     white       6 40.2
## 10 55-59    white       6 33.9
## 11 60-64    white       5 38.8
```

```r
current_news_race_group_age_10_salary <- news_salaried %>%
  group_by(age_group_10, race_grouping) %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )
```

```r
# A tibble: 13 x 4
# Groups: age_group_10 [6]
## age_group_10 race_grouping count median
## <fct>      <chr>     <int> <dbl>
## 1 <25       person of color 11 63780
## 2 <25       white       12 65140
## 3 25-34     person of color 55 83340
## 4 25-34     unknown    14 106890
## 5 25-34     white      125 86000
## 6 35-44     person of color 38 102890
## 7 35-44     unknown    7 140280
## 8 35-44     white      104 115258.
## 9 45-54     person of color 26 106932.
## 10 45-54    white      84 116687.
## 11 55-64    person of color 8 140424.
## 12 55-64    white      68 140052.
## 13 65+      white      13 159300
```

```r
current_news_race_group_age_10_hourly <- news_hourly %>%
  group_by(age_group_10, race_grouping) %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )
```

42
### # A tibble: 8 x 4
### # Groups: age_group_10 [5]
### age_group_10  race_grouping  count  median
### <fct>         <chr>        <int>  <dbl>
### 1 <25          person of color 6   29.5
### 2 <25          white          7   18.5
### 3 25-34       person of color 13  29.1
### 4 25-34       white          20  31.3
### 5 35-44       person of color  5   23.9
### 6 35-44       white          12  35.3
### 7 45-54       white          11  41.4
### 8 55-64       white          11  34.9

current_news_race_salaried_under_40 <- filter(news_salaried, age < 40) %>%
group_by() %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_news_race_salaried_under_40)

### # A tibble: 7 x 3
### race_ethnicity  count  median
### <chr>          <int>  <dbl>
### 1 <NA>          11  125000
### 2 White (United States of America) 198  90780
### 3 Black or African American (United States of America) 24  87970.
### 4 Asian (United States of America)  33  87000
### 5 Hispanic or Latino (United States of America) 19  79618.
### 6 Prefer Not to Disclose (United States of America)  6  77750
### 7 Two or More Races (United States of America) 13  76380

current_news_race_salaried_over_40 <- filter(news_salaried, age > 39) %>%
group_by() %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_news_race_salaried_over_40)

### # A tibble: 5 x 3
### race_ethnicity  count  median
### <chr>          <int>  <dbl>
### 1 <NA>          10  151408.
### 2 White (United States of America) 208 128484.
### 3 Hispanic or Latino (United States of America) 9  126580
### 4 Asian (United States of America) 13 111761.
### 5 Black or African American (United States of America) 24 109396.

current_news_race_hourly_under_40 <- filter(news_hourly, age < 40) %>%
group_by() %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_news_race_hourly_under_40)
# A tibble: 3 x 3
## race_ethnicity count median
## <chr>      <int> <dbl>
## 1 White (United States of America) 32 32.0
## 2 Black or African American (United States of America) 10 29.9
## 3 Asian (United States of America) 7  25.0

current_news_race_hourly_over_40 <- filter(news_hourly, age > 39) %>%
group_by(race_ethnicity) %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_news_race_hourly_over_40)

# A tibble: 1 x 3
## race_ethnicity count median
## <chr>      <int> <dbl>
## 1 White (United States of America) 32 39.9

Gender x race/ethnicity

current_news_race_gender_salaried <- news_salaried %>%
group_by(race_ethnicity, gender) %>%
summarise(
  count = length(current_base_pay)
)
suppress(current_news_race_gender_salaried)

# A tibble: 13 x 3
## # Groups: race_ethnicity [7]
## race_ethnicity gender count
## <chr>      <chr> <int>
## 1 Asian (United States of America) Female 34
## 2 Asian (United States of America) Male  12
## 3 Black or African American (United States of America) Female 24
## 4 Black or African American (United States of America) Male  24
## 5 Hispanic or Latino (United States of America) Female 14
## 6 Hispanic or Latino (United States of America) Male  14
## 7 Prefer Not to Disclose (United States of America) Male  5
## 8 Two or More Races (United States of America) Female  9
## 9 Two or More Races (United States of America) Male   5
##10 White (United States of America) Female 188
##11 White (United States of America) Male  218
##12 <NA> Female  10
##13 <NA> Male   11

current_news_race_gender_hourly <- news_hourly %>%
group_by(race_ethnicity, gender) %>%
summarise(
  count = length(current_base_pay)
)
suppress(current_news_race_gender_hourly)

# A tibble: 5 x 3
## # Groups: race_ethnicity [3]
## race_ethnicity gender count
## <chr>      <chr> <int>
current_news_race_gender_median_salaried <- news_salaried %>%
group_by(race_ethnicity, gender)
%>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_news_race_gender_median_salaried)

current_news_race_gender_hourly_median <- news_hourly %>%
group_by(race_ethnicity, gender)
%>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_news_race_gender_hourly_median)

current_news_race_gender_salaried_under_40 <- filter(news_salaried, age < 40) %>%
group_by(race_ethnicity, gender)
%>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_news_race_gender_salaried_under_40)
## A tibble: 10 x 4
## Groups: race_ethnicity [6]
# race_ethnicity gender count median
# <chr> <chr> <int> <dbl>
1 Asian (United States of America) Female 25 8.60e4
2 Asian (United States of America) Male 8 1.03e5
3 Black or African American (United States of America) Female 16 8.54e4
4 Black or African American (United States of America) Male 8 1.28e5
5 Hispanic or Latino (United States of America) Female 12 8.01e4
6 Hispanic or Latino (United States of America) Male 7 7.50e4
7 Two or More Races (United States of America) Female 9 7.50e4
8 White (United States of America) Female 105 8.58e4
9 White (United States of America) Male 93 9.57e4
10 <NA> Male 7 1.35e5

current_news_race_gender_salaried_over_40 <- filter(news_salaried, age > 39) %>%
  group_by(race_ethnicity, gender)

current_news_race_gender_salaried_over_40 <- current_news_race_gender_salaried_over_40 %>%
  summarise(count = length(current_base_pay),
            median = median(current_base_pay, na.rm = FALSE))
suppress(current_news_race_gender_salaried_over_40)

## A tibble: 7 x 4
## Groups: race_ethnicity [5]
# race_ethnicity gender count median
# <chr> <chr> <int> <dbl>
1 Asian (United States of America) Female 9 111761.
2 Black or African American (United States of America) Female 8 115002.
3 Black or African American (United States of America) Male 16 107464.
4 Hispanic or Latino (United States of America) Male 7 126580
5 White (United States of America) Female 83 122917.
6 White (United States of America) Female 125 130000
7 <NA> Female 6 148572.

current_news_race_gender_hourly_under_40 <- filter(news_hourly, age < 40) %>%
  group_by(race_ethnicity, gender)

current_news_race_gender_hourly_under_40 <- current_news_race_gender_hourly_under_40 %>%
  summarise(count = length(current_base_pay),
            median = median(current_base_pay, na.rm = FALSE))
suppress(current_news_race_gender_hourly_under_40)

## A tibble: 4 x 4
## Groups: race_ethnicity [3]
# race_ethnicity gender count median
# <chr> <chr> <int> <dbl>
1 Asian (United States of America) Female 5 25.0
2 Black or African American (United States of America) Female 6 31.0
3 White (United States of America) Female 21 31.9
4 White (United States of America) Male 11 33.7

current_news_race_gender_hourly_over_40 <- filter(news_hourly, age > 39) %>%
  group_by(race_ethnicity, gender)

current_news_race_gender_hourly_over_40 <- current_news_race_gender_hourly_over_40 %>%
  summarise(count = length(current_base_pay),
            median = median(current_base_pay, na.rm = FALSE))
suppress(current_news_race_gender_hourly_over_40)
Years of service

```r
current_news_yos_salaried <- news_salaried %>%
  group_by(years_of_service_grouped)
current_news_yos_salaried <- current_news_yos_salaried %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))

current_news_yos_hourly <- news_hourly %>%
  group_by(years_of_service_grouped)
current_news_yos_hourly <- current_news_yos_hourly %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))

current_news_yos_gender_salaried <- news_salaried %>%
  group_by(years_of_service_grouped, gender)
current_news_yos_gender_salaried <- current_news_yos_gender_salaried %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))
```

```r
# A tibble: 3 x 3
# Groups: years_of_service_grouped [8]
## years_of_service_grouped count median
## <fct> <int> <dbl>
## 1 0 16 29.5
## 2 1-2 26 32.7
## 3 3-5 9 33.0
## 4 6-10 15 35.9
## 5 11-15 10 36.5
## 6 16-20 11 32.3
## 7 21-25 5 38.9
```
## years_of_service_grouped  gender  count  median
## <fct>     <chr>     <int>    <dbl>
## 1 0        Female  39 80000
## 2 0        Male    26 105000
## 3 1-2      Female  70 87390
## 4 1-2      Male    58 101788.
## 5 3-5      Female  72 88530
## 6 3-5      Male    74 95265.
## 7 6-10     Female  26 100640.
## 8 6-10     Male    34 119562.
## 9 9-11-15  Female  25 98545.
## 10 9-11-15 Male    25 129780
## 11 11-15   Female  28 119826.
## 12 11-15   Male    40 129745.
## 13 13-15   Female  11 134780
## 14 13-15   Male    13 148417.
## 15 25+     Female  13 142280
## 16 25+     Male    20 131793.

current_news_yos_gender_hourly <- news_hourly %>%
group_by(years_of_service_grouped, gender)
current_news_yos_gender_hourly <- current_news_yos_gender_hourly
%>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_news_yos_gender_hourly)

## # A tibble: 9 x 4
## # Groups: years_of_service_grouped [6]
## # years_of_service_grouped  gender  count  median
## <fct>     <chr>     <int>    <dbl>
## 1 0        Female  11 28.2
## 2 0        Male    5 30.8
## 3 1-2      Female  18 32.4
## 4 1-2      Male    8 33.3
## 5 3-5      Male    6 32.5
## 6 6-10     Female  8 31.4
## 7 6-10     Male    7 36.7
## 8 11-15    Female  9 38.4
## 9 16-20    Female  7 42.1

current_news_yos_race_salaried <- news_salaried %>%
group_by(years_of_service_grouped, race_ethnicity)
current_news_yos_race_salaried <- current_news_yos_race_salaried
%>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_news_yos_race_salaried)

## # A tibble: 21 x 4
## # Groups: years_of_service_grouped [8]
## # years_of_service_grouped  race_ethnicity  count  median
## <fct>     <chr>                <int>    <dbl>
## 1 0        Asian (United States of America) 7 7.70e4
## 2 0        White (United States of America) 42 1.00e4
## 3 1-2      Asian (United States of America) 13 8.48e4
## 4 1-2      Black or African American (United St- 10 8.98e4

48
## 5 1-2 Hispanic or Latino (United States of America) 6.29e4
## 6 1-2 Two or More Races (United States of America) 6.80e4
## 7 1-2 White (United States of America) 9.58e4
## 8 1-2 <NA> 1.40e5
## 9 3-5 Asian (United States of America) 9.36e4
## 10 3-5 Black or African American (United States of America) 9.73e4
## ... with 11 more rows

```r
current_news_yos_race_hourly <- news_hourly %>%
  group_by(years_of_service_grouped, race_ethnicity)
current_news_yos_race_hourly <- current_news_yos_race_hourly %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))
```

```r
## # A tibble: 7 x 4
## # Groups: years_of_service_grouped [7]
## years_of_service_grouped race_ethnicity count median
## <fct> <chr> <int> <dbl>
## 1 0 White (United States of America) 6 29.5
## 2 1-2 White (United States of America) 18 32.8
## 3 3-5 White (United States of America) 6 32.5
## 4 6-10 White (United States of America) 9 35.9
## 5 11-15 White (United States of America) 8 39.9
## 6 16-20 White (United States of America) 9 42.1
## 7 21-25 White (United States of America) 5 38.9
```

```r
current_news_yos_race_gender_salaried <- news_salaried %>%
  group_by(years_of_service_grouped, race_ethnicity, gender)
current_news_yos_race_gender_salaried <- current_news_yos_race_gender_salaried %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))
```

```r
## # A tibble: 26 x 5
## # Groups: years_of_service_grouped, race_ethnicity [16]
## years_of_service_grouped race_ethnicity gender count median
## <fct> <chr> <chr> <int> <dbl>
## 1 0 Asian (United States of America) Female 7 7.70e4
## 2 0 White (United States of America) Female 25 8.50e4
## 3 0 White (United States of America) Male 17 1.10e5
## 4 1-2 Asian (United States of America) Female 11 7.70e4
## 5 1-2 Black or African American (United States of America) Female 6 8.58e4
## 6 1-2 Hispanic or Latino (United States of America) Female 5 8.20e4
## 7 1-2 White (United States of America) Female 41 9.08e4
## 8 1-2 White (United States of America) Male 44 9.98e4
## 9 3-5 Asian (United States of America) Female 8 9.36e4
## 10 3-5 Black or African American (United States of America) Female 7 9.61e4
```

```r
current_news_yos_race_gender_hourly <- news_hourly %>%
  group_by(years_of_service_grouped, race_ethnicity, gender)
current_news_yos_race_gender_hourly <- current_news_yos_race_gender_hourly %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))
```

```r
## # A tibble: 7 x 4
## # Groups: years_of_service_grouped [7]
## years_of_service_grouped race_ethnicity gender count median
## <fct> <chr> <chr> <int> <dbl>
## 1 0 Asian (United States of America) Female 7 7.70e4
## 2 0 White (United States of America) Female 25 8.50e4
## 3 0 White (United States of America) Male 17 1.10e5
## 4 1-2 Asian (United States of America) Female 11 7.70e4
## 5 1-2 Black or African American (United States of America) Female 6 8.58e4
## 6 1-2 Hispanic or Latino (United States of America) Female 5 8.20e4
## 7 1-2 White (United States of America) Female 41 9.08e4
## 8 1-2 White (United States of America) Male 44 9.98e4
## 9 3-5 Asian (United States of America) Female 8 9.36e4
## 10 3-5 Black or African American (United States of America) Female 7 9.61e4
```

```r
current_news_yos_race_gender_hourly <- news_hourly %>%
  group_by(years_of_service_grouped, race_ethnicity, gender)
A tibble: 5 x 5
## Groups: years_of_service_grouped, race_ethnicity [4]
## years_of_service_grouped race_ethnicity gender count median
## <fct> <chr>   <chr>     <int> <dbl>
## 1 1-2       White (United States of Amer- Female 12  32.7
## 2 1-2       White (United States of Amer- Male  6  33.3
## 3 6-10      White (United States of Amer- Male  5  35.9
## 4 11-15     White (United States of Amer- Female  7  41.4
## 5 16-20     White (United States of Amer- Female  6  42.4

Age

current_median_news_age_5_salaried <- news_salaried %>% group_by(age_group_5)
current_median_news_age_5_salaried <- current_median_news_age_5_salaried %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_median_news_age_5_salaried)

A tibble: 10 x 3
## age_group_5 count median
## <fct>   <int> <dbl>
## 1 <25    24   64640
## 2 25-29  91   80500
## 3 30-34 103   90780
## 4 35-39  86  105691.
## 5 40-44  63  125769.
## 6 45-49  43  102796.
## 7 50-54  70  115770.
## 8 55-59  51  147780
## 9 60-64  28  131217.
## 10 65+  15  157095.

current_median_news_age_5_hourly <- news_hourly %>% group_by(age_group_5)
current_median_news_age_5_hourly <- current_median_news_age_5_hourly %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_median_news_age_5_hourly)

A tibble: 10 x 3
## age_group_5 count median
## <fct>   <int> <dbl>
## 1 <25    14   29.5
## 2 25-29  23   30.8
## 3 30-34  11   33.7
## 4 35-39   8   33.9
## 5 40-44   9   33.1
## 6 45-49   6   47.4
## 7 50-54   8   36.2
## 8 55-59   7   34.9
## 9 60-64   5   38.8
## 10 65+    5   42.6
current_median_news_age_10_salaried <- news_salaried %>% group_by(age_group_10)  
  count = length(current_base_pay),  
  median = median(current_base_pay, na.rm = FALSE)  
)  
suppress(current_median_news_age_10_salaried)

## # A tibble: 6 x 3  
##   age_group_10 count median  
##     <fct>    <int> <dbl>  
## 1 <25       24    64640  
## 2 25-34     194   85890  
## 3 35-44     149  115237.  
## 4 45-54     113  114803  
## 5 55-64     79   141016.  
## 6 65+       15   157095. 

current_median_news_age_10_hourly <- news_hourly %>% group_by(age_group_10)  
  count = length(current_base_pay),  
  median = median(current_base_pay, na.rm = FALSE)  
)  
suppress(current_median_news_age_10_hourly)

## # A tibble: 6 x 3  
##   age_group_10 count median  
##     <fct>    <int> <dbl>  
## 1 <25       14     29.5  
## 2 25-34     34     31.0  
## 3 35-44     17     33.1  
## 4 45-54     14     41.1  
## 5 55-64     12     35.8  
## 6 65+       5     42.6  

current_news_age_5_yos_salary <- news_salaried %>% group_by(age_group_5, years_of_service_grouped)  
  count = length(current_base_pay),  
  median = median(current_base_pay, na.rm = FALSE)  
)  
suppress(current_news_age_5_yos_salary)

## # A tibble: 39 x 4  
## # Groups: age_group_5 [9]  
##   age_group_5 years_of_service_grouped count median  
##      <fct>                <int> <dbl>  
## 1 <25                    0     9 66000  
## 2 <25                    1-2   13 63780  
## 3 25-29                  0     19 82000  
## 4 25-29                  1-2   30 78500  
## 5 25-29                  3-5   41 81757.  
## 6 30-34                  0     13 87000  
## 7 30-34                  1-2   28 93528.  
## 8 30-34                  3-5   43 88780  
## 9 30-34                  6-10  15 82312.  
## 10 35-39                 0     9 110000
current_news_age_5_yos_hourly <- news_hourly %% group_by(age_group_5, years_of_service_grouped) %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_news_age_5_yos_hourly)

current_news_age_10_yos_salary <- news_salaried %% group_by(age_group_10, years_of_service_grouped) %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_news_age_10_yos_salary)

current_news_age_10_yos_hourly <- news_hourly %% group_by(age_group_10, years_of_service_grouped) %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_news_age_10_yos_hourly)
```r
## 5 25-34 3-5 6 30.0
## 6 35-44 11-15 6 33.9

current_median_news_age_5_gender_salaried <- news_salaried %>%
group_by(age_group_5, gender)
current_median_news_age_5_gender_salaried <-
current_median_news_age_5_gender_salaried %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_median_news_age_5_gender_salaried)

## # A tibble: 20 x 4
## # Groups: age_group_5 [10]
## age_group_5 gender count median
## <fct> <chr> <int> <dbl>
## 1 <25 Female 19 64280
## 2 <25 Male 5 72000
## 3 25-29 Female 60 80000
## 4 25-29 Male 31 85500
## 5 30-34 Female 57 87000
## 6 30-34 Male 46 97828.
## 7 35-39 Female 38 88892.
## 8 35-39 Male 48 116030
## 9 40-44 Female 22 133200.
## 10 40-44 Male 41 125000
## 11 45-49 Female 20 117295.
## 12 45-49 Male 23 99725
## 13 50-54 Female 29 108864.
## 14 50-54 Male 41 126280.
## 15 55-59 Female 22 145655.
## 16 55-59 Male 29 147780
## 17 60-64 Female 12 129325.
## 18 60-64 Male 16 131217.
## 19 65+ Female 5 157095.
## 20 65+ Male 10 156260.

current_median_news_age_5_gender_hourly <- news_hourly %>%
group_by(age_group_5, gender)
current_median_news_age_5_gender_hourly <-
current_median_news_age_5_gender_hourly %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_median_news_age_5_gender_hourly)

## # A tibble: 9 x 4
## # Groups: age_group_5 [8]
## age_group_5 gender count median
## <fct> <chr> <int> <dbl>
## 1 <25 Female 12 31.4
## 2 25-29 Female 17 31.2
## 3 25-29 Male 6 21.0
## 4 30-34 Male 7 33.7
## 5 35-39 Female 5 31.9
## 6 40-44 Female 5 41.4
## 7 45-49 Female 5 44.5
## 8 50-54 Female 6 40.2
## 9 55-59 Male 5 34.9
```
current_median_news_age_10_gender_salaried <- news_salaried %>% group_by(age_group_10, gender)
current_median_news_age_10_gender_salaried <- current_median_news_age_10_gender_salaried %>% summarise(
count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_median_news_age_10_gender_salaried)

## # A tibble: 12 x 4
## # Groups: age_group_10 [6]
## age_group_10 gender count median
## <fct> <chr> <int> <dbl>
## 1 <25 Female 19 64280
## 2 <25 Male 5 72000
## 3 25-34 Female 117 83147.
## 4 25-34 Male 77 92500
## 5 35-44 Female 60 105691.
## 6 35-44 Male 89 118785
## 7 45-54 Female 49 108864.
## 8 45-54 Male 64 117982.
## 9 55-64 Female 34 140424.
## 10 55-64 Male 45 146542.
## 11 65+ Female 5 157095.
## 12 65+ Male 10 156260.

current_median_news_age_10_gender_hourly <- news_hourly %>% group_by(age_group_10, gender)
current_median_news_age_10_gender_hourly <- current_median_news_age_10_gender_hourly %>% summarise(
count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_median_news_age_10_gender_hourly)

## # A tibble: 8 x 4
## # Groups: age_group_10 [5]
## age_group_10 gender count median
## <fct> <chr> <int> <dbl>
## 1 <25 Female 12 31.4
## 2 25-34 Female 21 31.2
## 3 25-34 Male 13 30.8
## 4 35-44 Female 10 33.1
## 5 35-44 Male 7 35.9
## 6 45-54 Female 11 41.4
## 7 55-64 Female 5 42.1
## 8 55-64 Male 7 33.4

current_median_news_age_5_race_salaried <- news_salaried %>% group_by(age_group_5, race_ethnicity)
current_median_news_age_5_race_salaried <- current_median_news_age_5_race_salaried %>% summarise(
count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_median_news_age_5_race_salaried)

## # A tibble: 25 x 4
## # Groups: age_group_5 [10]
## age_group_5 race_ethnicity count median
## <fct> <chr> <int> <dbl>

54
## 1 <25 Asian (United States of America) 5 65780
## 2 <25 White (United States of America) 12 65140
## 3 25-29 Asian (United States of America) 11 77000
## 4 25-29 Black or African American (United States of America) 6 81000
## 5 25-29 Two or More Races (United States of America) 6 75690
## 6 25-29 White (United States of America) 59 81757
## 7 30-34 Asian (United States of America) 10 95780
## 8 30-34 Black or African American (United States of America) 9 88133
## 9 30-34 Hispanic or Latino (United States of America) 6 80596
## 10 30-34 White (United States of America) 66 92640
## ... with 15 more rows

```r
# current_median_news_age_5_race_hourly <- news_hourly %>%
group_by(age_group_5, race_ethnicity)
summarise(count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE))
suppress(current_median_news_age_5_race_hourly)

# A tibble: 10 x 4
# Groups: age_group_5 [9]
  age_group_5 race_ethnicity count median
  <fct> <chr> <int> <dbl>
1 <25 White (United States of America) 7 18.5
2 25-29 Black or African American (United States of America) 8 30.1
3 25-29 White (United States of America) 11 30.8
4 30-34 White (United States of America) 9 33.7
5 35-39 White (United States of America) 5 34.7
6 40-44 White (United States of America) 7 41.4
7 45-49 White (United States of America) 5 44.5
8 50-54 White (United States of America) 6 40.2
9 55-59 White (United States of America) 6 33.9
10 60-64 White (United States of America) 5 38.8

# current_median_news_age_10_race_salaried <- news_salaried %>%
group_by(age_group_10, race_ethnicity)
summarise(count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE))
suppress(current_median_news_age_10_race_salaried)

# A tibble: 21 x 4
# Groups: age_group_10 [6]
  age_group_10 race_ethnicity count median
  <fct> <chr> <int> <dbl>
1 <25 Asian (United States of America) 5 6.58e4
2 <25 White (United States of America) 12 6.51e4
3 25-34 Asian (United States of America) 21 8.60e4
4 25-34 Black or African American (United States of America) 15 8.70e4
5 25-34 Hispanic or Latino (United States of America) 10 8.12e4
6 25-34 Prefer Not to Disclose (United States of America) 5 7.85e4
7 25-34 Two or More Races (United States of America) 9 7.64e4
8 25-34 White (United States of America) 125 8.60e4
9 25-34 <NA> 9 1.16e5
10 35-44 Asian (United States of America) 11 1.08e5
```
```r
# ... with 11 more rows

current_median_news_age_10_race_hourly <- news_hourly %>%
  group_by(age_group_10, race_ethnicity) %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))
suppress(current_median_news_age_10_race_hourly)

## # A tibble: 6 x 4
## # Groups: age_group_10 [5]
## age_group_10 race_ethnicity count median
## <fct> <chr>                 <int> <dbl>
## 1 <25 White (United States of America)  7  18.5
## 2 25-34 Black or African American (United States of Am-  8  30.1
## 3 25-34 White (United States of America) 20  31.3
## 4 35-44 White (United States of America) 12  35.3
## 5 45-54 White (United States of America) 11  41.4
## 6 55-64 White (United States of America) 11  34.9

current_median_news_age_5_race_group_salaried <- news_salaried %>%
  group_by(age_group_5, race_grouping) %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))
suppress(current_median_news_age_5_race_group_salaried)

## # A tibble: 21 x 4
## # Groups: age_group_5 [10]
## age_group_5 race_grouping count median
## <fct> <chr>              <int> <dbl>
## 1 <25 person of color  11 63780
## 2 <25 white            12 65140
## 3 25-29 person of color 27 80000
## 4 25-29 unknown        5 88280
## 5 25-29 white          59 81757.
## 6 30-34 person of color 28 86983.
## 7 30-34 unknown        9 108000
## 8 30-34 white          66 92640
## 9 35-39 person of color 23 99238.
## 10 35-39 white         61 105780
## # ... with 11 more rows

current_median_news_age_5_race_group_hourly <- news_hourly %>%
  group_by(age_group_5, race_grouping) %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))
suppress(current_median_news_age_5_race_group_hourly)

## # A tibble: 11 x 4
## # Groups: age_group_5 [9]
## age_group_5 race_grouping count median
## <fct> <chr>              <int> <dbl>
## 1 <25 person of color  6  29.5
## 2 <25 white            7  18.5
```
## 3 25-29 person of color 12 27.1
## 4 25-29 white 11 30.8
## 5 30-34 white 9 33.7
## 6 35-39 white 5 34.7
## 7 40-44 white 7 41.4
## 8 45-49 white 5 44.5
## 9 50-54 white 6 40.2
## 10 55-59 white 6 33.9
## 11 60-64 white 5 38.8

```
current_median_news_age_10_race_group_salaried <- news_salaried %>%
group_by(age_group_10, race_grouping) %>%
summarise(count = length(current_base_pay),
           median = median(current_base_pay, na.rm = FALSE))
```

## # A tibble: 13 x 4
## # Groups: age_group_10 [6]
##  age_group_10 race_grouping count median
##    <fct>         <chr>  <int> <dbl>
## 1   <25          person of color 11 63780
## 2   <25          white          12 65140
## 3   25-34        person of color 55 83340
## 4   25-34        unknown       14 106890
## 5   25-34        white         125 86000
## 6   35-44        person of color 38 102890
## 7   35-44        unknown       7 140280
## 8   35-44        white         104 115258
## 9   45-54        person of color 26 106932
## 10  45-54        white         84 116687
## 11  55-64        person of color 8 140424
## 12  55-64        white         68 140052
## 13  65+          white         13 159300

```
current_median_news_age_10_race_group_hourly <- news_hourly %>%
group_by(age_group_10, race_grouping) %>%
summarise(count = length(current_base_pay),
           median = median(current_base_pay, na.rm = FALSE))
```

## # A tibble: 8 x 4
## # Groups: age_group_10 [5]
##  age_group_10 race_grouping count median
##    <fct>         <chr>  <int> <dbl>
## 1   <25          person of color 6 29.5
## 2   <25          white        7 18.5
## 3   25-34        person of color 13 29.1
## 4   25-34        white      20 31.3
## 5   35-44        person of color 5 23.9
## 6   35-44        white      12 35.3
## 7   45-54        white      11 41.4
## 8   55-64        white      11 34.9
current_median_news_age_5_race_gender_salaried <- news_salaried %>% group_by(age_group_5, race_ethnicity, gender) %>% 
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_median_news_age_5_race_gender_salaried)

## # A tibble: 30 x 5
## # Groups: age_group_5, race_ethnicity [22]
## age_group_5 race_ethnicity gender count median
## <fct> <chr> <chr> <int> <dbl>
## 1 <25 Asian (United States of America) Female 5 6.58e4
## 2 <25 White (United States of America) Female 9 6.43e4
## 3 25-29 Asian (United States of America) Female 9 7.70e4
## 4 25-29 Black or African American (United States of America) Female 5 8.00e4
## 5 25-29 White (United States of America) Female 38 8.19e4
## 6 25-29 White (United States of America) Male 21 7.68e4
## 7 30-34 Asian (United States of America) Female 8 1.01e5
## 8 30-34 Black or African American (United States of America) Female 5 8.58e4
## 9 30-34 Hispanic or Latino (United States of America) Female 6 8.06e4
## 10 30-34 White (United States of America) Female 32 8.77e4
## # ... with 20 more rows

current_median_news_age_5_race_gender_hourly <- news_hourly %>% group_by(age_group_5, race_ethnicity, gender) %>% 
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_median_news_age_5_race_gender_hourly)

## # A tibble: 5 x 5
## # Groups: age_group_5, race_ethnicity [5]
## age_group_5 race_ethnicity gender count median
## <fct> <chr> <chr> <int> <dbl>
## 1 <25 White (United States of America) Female 5 32
## 2 25-29 White (United States of America) Female 10 31.2
## 3 30-34 White (United States of America) Male 6 34.4
## 4 45-49 White (United States of America) Female 5 44.5
## 5 55-59 White (United States of America) Male 5 34.9

current_median_news_age_10_race_gender_salaried <- news_salaried %>% group_by(age_group_10, race_ethnicity, gender) %>% 
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_median_news_age_10_race_gender_salaried)

## # A tibble: 23 x 5
## # Groups: age_group_10, race_ethnicity [17]
## age_group_10 race_ethnicity gender count median
## <fct> <chr> <chr> <int> <dbl>
## 1 <25 Asian (United States of America) Female 5 6.58e4
## 2 <25 White (United States of America) Female 9 6.43e4
## 3 25-34 Asian (United States of America) Female 17 8.70e4
## 4 25-34 Black or African American (United States of America) Female 10 8.10e4

58
## 5 25-34 Black or African American (United States) Male 5 1.40e5
## 6 25-34 Hispanic or Latino (United States of America) Female 8 8.12e4
## 7 25-34 Two or More Races (United States of America) Female 6 7.57e4
## 8 25-34 White (United States of America) Female 70 9.08e4
## 9 25-34 White (United States of America) Male 55 9.08e4
## 10 25-34 <NA> Male 6 1.32e5
## # ... with 13 more rows

current_median_news_age_10_race_gender_hourly <- news_hourly %>%
group_by(age_group_10, race_ethnicity, gender)
summarise(count = length(current_base_pay),
           median = median(current_base_pay, na.rm = FALSE))
suppress(current_median_news_age_10_race_gender_hourly)

## # A tibble: 7 x 5
## # Groups: age_group_10, race_ethnicity [5]
## age_group_10 race_ethnicity gender count median
## <fct> <chr> <chr> <int> <dbl>
## 1 <25 White (United States of America) Female 5 32
## 2 25-34 White (United States of America) Female 13 30.8
## 3 25-34 White (United States of America) Male 7 33.7
## 4 35-44 White (United States of America) Female 7 34.7
## 5 35-44 White (United States of America) Male 5 35.9
## 6 45-54 White (United States of America) Female 9 44.5
## 7 55-64 White (United States of America) Male 7 33.4

current_median_news_age_5_race_group_gender_salaried <- news_salaried %>%
group_by(age_group_5, race_grouping, gender)
summarise(count = length(current_base_pay),
           median = median(current_base_pay, na.rm = FALSE))
suppress(current_median_news_age_5_race_group_gender_salaried)

## # A tibble: 31 x 5
## # Groups: age_group_5, race_grouping [18]
## age_group_5 race_grouping gender count median
## <fct> <chr> <chr> <int> <dbl>
## 1 <25 person of color Female 10 64390
## 2 <25 white Female 9 64280
## 3 25-29 person of color Female 19 77000
## 4 25-29 person of color Male 8 88540
## 5 25-29 white Female 38 81878.
## 6 25-29 white Male 21 76780
## 7 30-34 person of color Female 22 86373.
## 8 30-34 person of color Male 6 106000
## 9 30-34 unknown Male 6 120390
## 10 30-34 white Female 32 87660
## # ... with 21 more rows

current_median_news_age_5_race_group_gender_hourly <- news_hourly %>%
group_by(age_group_5, race_grouping, gender)
summarise(count = length(current_base_pay),
           median = median(current_base_pay, na.rm = FALSE))
suppress(current_median_news_age_5_race_group_gender_hourly)
## A tibble: 8 x 5
## Groups: age_group_5, race_grouping [7]
## # Groups: age_group_5 race_grouping gender count median
##  <fct> <chr>               <chr> <int> <dbl>
## 1 <25 person of color Female 6 29.5
## 2 <25 white Female 5 32
## 3 25-29 person of color Female 7 31.2
## 4 25-29 person of color Male 5 20.9
## 5 25-29 white Female 10 31.2
## 6 30-34 white Male 6 34.4
## 7 45-49 white Female 5 44.5
## 8 55-59 white Male 5 34.9

```r
current_median_news_age_10_race_group_gender_salaried <- news_salaried %>%
group_by(age_group_10, race_grouping, gender)
summarise(
count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_median_news_age_10_race_group_gender_salaried)
```

## A tibble: 20 x 5
## Groups: age_group_10, race_grouping [12]
## # Groups: age_group_10 race_grouping gender count median
##  <fct> <chr>               <chr> <int> <dbl>
## 1 <25 person of color Female 10 64390
## 2 <25 white Female 9 64280
## 3 25-34 person of color Female 41 82000.
## 4 25-34 person of color Male 14 89540
## 5 25-34 unknown Female 6 92140
## 6 25-34 unknown Male 8 120390
## 7 25-34 white Female 70 84640
## 8 25-34 white Male 55 90780
## 9 35-44 person of color Female 19 100000
## 10 35-44 person of color Male 19 113280
## 11 35-44 white Female 37 105000
## 12 35-44 white Male 67 120780
## 13 45-54 person of color Female 7 108864.
## 14 45-54 person of color Male 19 105000
## 15 45-54 white Female 42 111589.
## 16 45-54 white Male 42 123530.
## 17 55-64 person of color Female 6 142688.
## 18 55-64 white Female 26 130924.
## 19 55-64 white Male 42 147161.
## 20 65+ white Male 9 159458.

```r
current_median_news_age_10_race_group_gender_hourly <- news_hourly %>%
group_by(age_group_10, race_grouping, gender)
summarise(
count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_median_news_age_10_race_group_gender_hourly)
```

## A tibble: 10 x 5
## Groups: age_group_10, race_grouping [7]
## # Groups: age_group_10 race_grouping gender count median
##  <fct> <chr>               <chr> <int> <dbl>
## 1 25 person of color Female 10 7000
## 2 25 white Female 9 8200
## 3 30-34 person of color Female 41 10600
## 4 30-34 person of color Male 19 9080
## 5 30-34 unknown Female 6 92140
## 6 30-34 unknown Male 8 120390
## 7 30-34 white Female 70 84640
## 8 30-34 white Male 55 90780
## 9 40-49 person of color Female 19 100000
## 10 40-49 person of color Male 20 130924.
```
## # A tibble: 18 x 3
##   desk          count median
##   <chr>     <int>    <dbl>
## 1 National  106  149520.
## 2 Foreign  25  135000
## 3 Financial 38  133510.
## 4 Investigative 13  129780
## 5 Style  45  107171.
## 6 Local  65  105780
## 7 Editorial 33  105000
## 8 Graphics 15  100780
## 9 Universal Desk 8  100444.
## 10 Sports  37  100000
## 11 Outlook  6  99938.
## 12 Audio  45  88065.
## 13 Design  45  88065.
## 14 Operations  6  87890
## 15 Multiplatform 26  86104
## 16 Video  46  84250
## 17 Audience Development and Engagement 16  83530
## 18 Emerging News Products 30  75000
```

Desks

```r
# current_news_median_desk_salaried <- news_salaried %>%
  group_by(desk) %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )

# current_news_median_desk_hourly <- news_hourly %>%
  group_by(desk) %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )
```

```
## # A tibble: 10 x 3
##   desk          count median
##   <chr>     <int>    <dbl>
## 1 Audio      6  39.7
## 2 Universal Desk 8  38.7
```
## 3 Audience Development and Engagement 7 37.6
## 4 Multiplatform 16 34.1
## 5 Editorial 5 32.3
## 6 National 12 31.7
## 7 Local 5 26.5
## 8 Style 9 21.8
## 9 Sports 11 20.9
## 10 Operations 7 15.6

current_news_median_desk_gender_salaried <- news_salaried %>%
group_by(desk, gender)
current_news_median_desk_gender_salaried <- current_news_median_desk_gender_salaried %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_news_median_desk_gender_salaried)

## # A tibble: 31 x 4
## # Groups: desk [17]
##   desk gender count median
##   <chr> <chr> <int> <dbl>
## 1 National Male 57 169780
## 2 Foreign Male 14 145390
## 3 Editorial Male 18 140271
## 4 National Female 49 139780
## 5 Financial Male 25 136468
## 6 Investigative Male 8 135030
## 7 Foreign Female 11 129970
## 8 Financial Female 13 125000
## 9 Investigative Female 5 125000
##10 Local Male 31 118850
## # ... with 21 more rows

current_news_median_desk_gender_hourly_salaried <- news_hourly %>%
group_by(desk, gender)
current_news_median_desk_gender_hourly_salaried <- current_news_median_desk_gender_hourly %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_news_median_desk_gender_hourly_salaried)

## # A tibble: 6 x 4
## # Groups: desk [6]
##   desk gender count median
##   <chr> <chr> <int> <dbl>
## 1 Audio Female 5 41.0
## 2 Universal Desk Female 5 35.9
## 3 Multiplatform Female 13 34.7
## 4 Sports Male 8 33.0
## 5 National Female 8 32.7
## 6 Style Female 8 26.7

current_news_median_desk_race_salaried <- news_salaried %>%
group_by(desk, race_ethnicity)
current_news_median_desk_race_salaried <- current_news_median_desk_race_salaried %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_news_median_desk_race_salaried)
```r
## # A tibble: 23 x 4
## # Groups: desk [16]
## desk    race_ethnicity    count  median
## <chr>    <chr>            <int>   <dbl>
## 1 National White (United States of America) 84  1.69e5
## 2 Investigative White (United States of America) 10  1.40e5
## 3 National Black or African American (United States of - 9  1.40e5
## 4 Foreign <NA> 20  1.38e5
## 5 Financial White (United States of America) 29  1.36e5
## 6 National Asian (United States of America) 11  1.26e5
## 7 Editorial White (United States of America) 27  1.20e5
## 8 Style White (United States of America) 38  1.12e5
## 9 Local White (United States of America) 46  1.08e5
## 10 Universal De~ White (United States of America) 5  1.04e5
## # ... with 13 more rows

current_news_median_desk_race_hourly <- news_hourly %>%
group_by(desk, race_ethnicity)

  current_news_median_desk_race_hourly <- current_news_median_desk_race_hourly %>%
    summarise(
      count = length(current_base_pay),
      median = median(current_base_pay, na.rm = FALSE)
    )

suppress_median(current_news_median_desk_race_hourly)

## # A tibble: 5 x 4
## # Groups: desk [5]
## desk    race_ethnicity    count  median
## <chr>    <chr>            <int>   <dbl>
## 1 Style White (United States of America) 5  38.9
## 2 Universal Desk White (United States of America) 6  38.7
## 3 Multiplatform White (United States of America) 12 36.5
## 4 Sports White (United States of America) 9  33.0
## 5 National White (United States of America) 9  32.7

current_news_median_desk_race_gender_salaried <- news_salaried %>%
group_by(desk, race_ethnicity, gender)

  current_news_median_desk_race_gender_salaried <- current_news_median_desk_race_gender_salaried %>%
    summarise(
      count = length(current_base_pay),
      median = median(current_base_pay, na.rm = FALSE)
    )

suppress_median(current_news_median_desk_race_gender_salaried)

## # A tibble: 30 x 5
## # Groups: desk, race_ethnicity [18]
## desk    race_ethnicity gender count  median
## <chr>    <chr>     <chr> <int>   <dbl>
## 1 National White (United States of America) Male 46  1.75e5
## 2 Investigative White (United States of America) Male 6  1.49e5
## 3 Financial White (United States of America) Male 21  1.40e5
## 4 Editorial White (United States of America) Male 16  1.40e5
## 5 Foreign <NA>  Male 11  1.40e5
## 6 National White (United States of America) Female 38  1.40e5
## 7 National Black or African American (United State~ Male 8  1.35e5
## 8 Foreign <NA>  Female 9  1.35e5
## 9 National Asian (United States of America) Female 8  1.33e5
## 10 Sports White (United States of America) Female 6  1.32e5
## # ... with 20 more rows
```
current_news_median_desk_race_gender_hourly <- news_hourly %>% group_by(desk, race_ethnicity, gender) %>% count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE)

suppress_median(current_news_median_desk_race_gender_hourly)

## A tibble: 4 x 5
## # Groups: desk, race_ethnicity [4]
## # Groups: desk, race_grouping [21]
## desk race_ethnicity gender count median
## <chr> <chr> <chr> <int> <dbl>
## 1 Style White (United States of America) Female 5 38.9
## 2 Multiplatform White (United States of America) Female 9 38.4
## 3 Sports White (United States of America) Male 7 33.0
## 4 National White (United States of America) Female 6 32.7

current_news_median_desk_race_gender_hourly <- news_salaried %>% group_by(desk, race_grouping, gender) %>% count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE)

suppress_median(current_news_median_desk_race_group_gender_salaried)

## A tibble: 36 x 5
## # Groups: desk, race_grouping [21]
## # Groups: desk, race_grouping [4]
## desk race_grouping gender count median
## <chr> <chr> <chr> <int> <dbl>
## 1 National white Male 46 175374.
## 2 Investigative white Male 6 149422.
## 3 Financial white Male 21 140387.
## 4 Editorial white Male 16 140271.
## 5 Foreign unknown Male 11 140000
## 6 National white Female 38 139734.
## 7 Foreign unknown Female 9 135000
## 8 National person of color Female 10 132780
## 9 Sports white Female 6 132015.
## 10 National person of color Male 11 130780

## ... with 26 more rows

current_news_median_desk_race_group_gender_salaried <- current_news_median_desk_race_group_gender_salaried %>% group_by(desk, race_grouping, gender) %>% count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE)

suppress_median(current_news_median_desk_race_group_gender_salaried)

## A tibble: 4 x 5
## # Groups: desk, race_grouping [4]
## # Groups: desk, race_grouping [4]
## desk race_grouping gender count median
## <chr> <chr> <chr> <int> <dbl>
## 1 Style white Female 5 38.9
## 2 Multiplatform white Female 9 38.4
## 3 Sports white Male 7 33.0
## 4 National white Female 6 32.7
current_news_median_desk_race_gender_age5_salaried <- news_salaried %>%
group_by(desk, race_ethnicity, gender, age_group_5)
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)

suppress_median(current_news_median_desk_race_gender_age5_salaried)

## # A tibble: 15 x 6
## # Groups: desk, race_ethnicity, gender [8]
##    desk race_ethnicity gender age_group_5 count median
##   <chr> <chr>       <chr> <fct> <int>  <dbl>
## 1   National White Male 40-44  9   1.70e5
## 2   National White Male 30-34  9   1.70e5
## 3   National White Male 50-54  5   1.68e5
## 4   National White Male 55-59  6   1.63e5
## 5   National White Male 40-44  5   1.60e5
## 6   National White Male 35-39 10   1.49e5
## 7   National White Male 35-39  7   1.47e5
## 8   National White Male 35-39  5   1.45e5
## 9   National White Male 55-59  6   1.28e5
##10  National White Male 30-34  5   1.25e5
##11  National White Female 25-29  5   1.25e5
##12  National White Female 35-39  6   1.09e5
##13  National White Female 30-34  5   8.80e4
##14  National White Male 45-49  5   8.73e4
##15  National White Female 25-29  7   7.00e4

current_news_median_desk_race_gender_age5_hourly <- news_hourly %>%
group_by(desk, race_ethnicity, gender, age_group_5)
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)

suppress_median(current_news_median_desk_race_gender_age5_hourly)

## # A tibble: 0 x 6
## # Groups: desk, race_ethnicity, gender [0]
## # ... with 6 variables: desk <chr>, race_ethnicity <chr>, gender <chr>,
## # age_group_5 <fct>, count <int>, median <dbl>

current_news_median_desk_race_group_gender_age5_salaried <- news_salaried %>%
group_by(desk, race_grouping, gender, age_group_5)
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)

suppress_median(current_news_median_desk_race_group_gender_age5_salaried)

## # A tibble: 16 x 6
## # Groups: desk, race_grouping, gender [9]
##    desk race_grouping gender age_group_5 count median
##   <chr> <chr>       <chr> <fct> <int>  <dbl>
## 1   National white Male 40-44  9   170000
## 2   National white Male 30-34  9   169780
## 3   National white Female 50-54  5   167780
## 4   National white Female 55-59  6   162854.
## 5   National white Female 40-44  5   160000

65
## 6 National white Male 35-39 10 148640
## 7 Sports white Male 35-39 7 147300
## 8 Financial white Male 35-39 5 144755
## 9 Local white Male 55-59 6 127655.
## 10 Foreign unknown Male 30-34 5 125000
## 11 National white Female 25-29 5 125000
## 12 National white Female 35-39 6 109390
## 13 Video white Female 30-34 5 88000
## 14 Sports white Male 45-49 5 87278.
## 15 Video person of color Female 25-29 8 76390
## 16 Emerging News Products white Female 25-29 7 70000

current_news_median_desk_race_group_gender_age5_hourly <- news_hourly %>%
group_by(desk, race_grouping, gender, age_group_5)

current_news_median_desk_race_group_gender_age5_hourly <- current_news_median_desk_race_group_gender_age5_hourly %>%
summarise(count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE))

suppress_median(current_news_median_desk_race_group_gender_age5_hourly)

## # A tibble: 0 x 6
## # Groups: desk, race_grouping, gender [0]
## # ... with 6 variables: desk <chr>, race_grouping <chr>, gender <chr>,
## # age_group_5 <fct>, count <int>, median <dbl>

current_news_median_desk_tier_salaried <- news_salaried %>%
group_by(tier)
current_news_median_desk_tier_salaried <- current_news_median_desk_tier_salaried %>%
summarise(count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE))

suppress_median(current_news_median_desk_tier_salaried)

## # A tibble: 4 x 3
## # <chr> <int> <dbl>
## 1 Tier 1 182 140140
## 2 Tier 2 209 105000
## 3 Tier 3 147 85780
## 4 Tier 4 36 75000

current_news_median_desk_tier_gender_salaried <- news_salaried %>%
group_by(tier, gender)
current_news_median_desk_tier_gender_salaried <- current_news_median_desk_tier_gender_salaried %>%
summarise(count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE))

suppress_median(current_news_median_desk_tier_gender_salaried)

## # A tibble: 8 x 4
## # <chr> <chr> <int> <dbl>
## 1 Tier 1 Male 104 150975.
## 2 Tier 1 Female 78 135160.
## 3 Tier 2 Male 112 112755.
## 4 Tier 2 Female 97 99252.
## 5 Tier 3 Male 64 90660.
## 6 Tier 3 Female 83 82010.
```r
## 7 Tier 4 Female 26 75000
## 8 Tier 4 Male 10 74086.

current_news_median_desk_tier_race_salaried <- news_salaried %>%
group_by(tier, race_ethnicity)
current_news_median_desk_tier_race_salaried <- current_news_median_desk_tier_race_salaried %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))

suppress_median(current_news_median_desk_tier_race_salaried)

## # A tibble: 14 x 4
## # Groups: tier 
##   tier race_ethnicity     count median
##   <chr> <chr>                <int> <dbl>
## 1 Tier 1 White (United States of America) 126 1.58e5
## 2 Tier 1 <NA>                21  1.40e5
## 3 Tier 1 Black or African American (United States of America) 12 1.35e5
## 4 Tier 1 Asian (United States of America) 17  1.25e5
## 5 Tier 2 White (United States of America) 159 1.07e5
## 6 Tier 2 Black or African American (United States of America) 16  1.02e5
## 7 Tier 2 Asian (United States of America) 14  9.38e4
## 8 Tier 2 Hispanic or Latino (United States of America) 11  9.21e4
## 9 Tier 2 Two or More Races (United States of America)  6  8.91e4
##10 Tier 3 White (United States of America)  98  8.80e4
##11 Tier 3 Black or African American (United States of America)  9  8.57e4
##12 Tier 3 Hispanic or Latino (United States of America)  8  8.08e4
##13 Tier 3 Asian (United States of America) 13  7.90e4
##14 Tier 4 White (United States of America) 23  7.50e4

current_news_median_desk_tier_race_gender_salaried <- news_salaried %>%
group_by(tier, race_ethnicity, gender)
current_news_median_desk_tier_race_gender_salaried <- current_news_median_desk_tier_race_gender_salaried %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))

current_news_median_desk_tier_race_gender_salaried <- news_salaried %>%
group_by(tier, race_grouping, gender)
current_news_median_desk_tier_race_gender_salaried <- current_news_median_desk_tier_race_gender_salaried %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))

current_news_median_desk_tier_race_group_gender_salaried <- news_salaried %>%
group_by(tier, race_grouping)
current_news_median_desk_tier_race_group_gender_salaried <- current_news_median_desk_tier_race_group_gender_salaried %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))
```

67
```r
suppress_median(current_news_median_desk_tier_race_group_gender_salaried)

## # A tibble: 17 x 5
## # Groups: tier, race_grouping [9]
## tier race_grouping gender count median
## <chr> <chr> <chr> <int> <dbl>
## 1 Tier 1 white Male 74 166149.
## 2 Tier 1 unknown Male 14 137890
## 3 Tier 1 unknown Female 10 137640
## 4 Tier 1 white Female 52 135825.
## 5 Tier 1 person of color Male 16 127890
## 6 Tier 1 person of color Female 10 132580
## 7 Tier 2 white Male 93 117844.
## 8 Tier 2 person of color Male 19 105000
## 9 Tier 2 white Female 66 102424.
## 10 Tier 2 person of color Female 30 93020.
## 11 Tier 3 white Male 43 92500
## 12 Tier 3 person of color Male 19 85692.
## 13 Tier 3 white Female 55 84780
## 14 Tier 3 person of color Female 27 79161.
## 15 Tier 4 person of color Female 10 78500
## 16 Tier 4 white Male 8 75500
## 17 Tier 4 white Female 15 75000

current_news_median_desk_tier_race_gender_age5_salaried <- news_salaried %>%
group_by(tier, race_ethnicity, gender, age_group_5) %>%
summarise(count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE))

## # A tibble: 42 x 6
## # Groups: tier, race_ethnicity, gender [9]
## tier race_ethnicity gender age_group_5 count median
## <chr> <chr> <chr> <fct> <int> <dbl>
## 1 Tier 1 White (United States of America) Male 50-54 5 180040
## 2 Tier 1 White (United States of America) Male 60-64 6 170790.
## 3 Tier 1 White (United States of America) Male 40-44 15 166999.
## 4 Tier 1 White (United States of America) Female 45-49 5 165000
## 5 Tier 1 White (United States of America) Female 55-59 6 162854.
## 6 Tier 1 White (United States of America) Male 55-59 9 160780
## 7 Tier 1 White (United States of America) Female 55-59 5 149030.
## 8 Tier 2 White (United States of America) Male 65+ 6 147473.
## 9 Tier 2 White (United States of America) Male 55-59 16 147161.
## 10 Tier 1 White (United States of America) Female 50-54 8 146280
## # ... with 32 more rows

current_news_median_desk_tier_race_group_gender_age5_salaried <- news_salaried %>%
group_by(tier, race_grouping, gender, age_group_5) %>%
summarise(count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE))

## # A tibble: 42 x 6
## # Groups: tier, race_grouping, gender [9]
## tier race_grouping gender age_group_5 count median
## <chr> <chr> <chr> <fct> <int> <dbl>
## 1 Tier 1 white Male 50-54 5 180040
## 2 Tier 1 unknown Male 60-64 6 170790.
## 3 Tier 1 unknown Female 40-44 15 166999.
## 4 Tier 1 white Female 45-49 5 165000
## 5 Tier 1 person of color Female 55-59 6 162854.
## 6 Tier 1 person of color Female 50-54 9 160780
## 7 Tier 1 person of color Male 55-59 5 149030.
## 8 Tier 2 white Male 65+ 6 147473.
## 9 Tier 2 white Male 55-59 16 147161.
## 10 Tier 1 white Female 50-54 8 146280
```
## # A tibble: 49 x 6
## # Groups: tier, race_grouping, gender [14]
## #   tier race_grouping gender age_group_5 count median
## #  <chr> <chr> <chr> <fct> <int> <dbl>
## 1 Tier 1 white Male 50-54 5 180040  
## 2 Tier 1 white Male 60-64 6 170790.
## 3 Tier 1 white Male 40-44 15 166999.
## 4 Tier 1 white Female 45-49 5 165000  
## 5 Tier 1 white Female 55-59 6 162854.
## 6 Tier 1 white Male 55-59 9 160780  
## 7 Tier 2 white Female 55-59 5 149030.  
## 8 Tier 2 white Male 65+ 6 147473.  
## 9 Tier 2 white Male 55-59 16 147161.
##10 Tier 1 white Female 50-54 8 146280  
## # ... with 39 more rows

### Job profiles

```r
current_news_median_job_salaried <- news_salaried %>%
  group_by(job_profile_current) %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))
suppress_median(current_news_median_job_salaried)
```

```r
## # A tibble: 18 x 3
## #   job_profile_current count median
## #    <chr>       <int> <dbl>
## 1 300113 - Columnist 19  170497.
## 2 300313 - Columnist - Editorial 7  151896.
## 3 320113 - Critic 9  150962.
## 4 330113 - Editorial Writer 7  129236.
## 5 280212 - Staff Writer 306  124040
## 6 390510 - Graphics Editor 7  111071
## 7 360114 - Photographer 16  106015.
## 8 126902 - Topic Editor 6  103772.
## 9 390610 - Graphics Reporter 8  97280
##10 120602 - Operations Editor 7  90780
##11 280226 - Video Journalist 20  89240
##12 390310 - Video Graphics Editor 8  87280
##13 120202 - Assistant Editor 23  87000
##14 390110 - Multiplatform Editor 53  83147.
##15 280228 - Designer 29  76000
##16 126202 - Photo Editor 8  74962.
##17 390410 - Digital Video Editor 22  74500
##18 289711 - News Intern - 2 Year 5  65780
```

```r
current_news_median_job_hourly <- news_hourly %>%
  group_by(job_profile_current) %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))
suppress_median(current_news_median_job_hourly)
```

```r
## # A tibble: 7 x 3
## #   job_profile_current count median
## #    <chr>       <int> <dbl>
## 1 300113 - Columnist 19  170497.
## 2 300313 - Columnist - Editorial 7  151896.
## 3 320113 - Critic 9  150962.
## 4 330113 - Editorial Writer 7  129236.
## 5 280212 - Staff Writer 306  124040
## 6 390510 - Graphics Editor 7  111071
## 7 360114 - Photographer 16  106015.
```
## job_profile_current  count median
## <chr> <int> <dbl>
## 1 280225 - Producer 18 36.7
## 2 400151 - Administrative Aide 6 35.3
## 3 397110 - Multiplatform Editor (PT/PTOC) 23 34.7
## 4 380117 - Research Assistant 6 31.2
## 5 410251 - Editorial Aide 12 21.4
## 6 430117 - News Aide 8 17.1
## 7 440116 - Copy Aide 5 15.2

```r
current_news_median_job_gender_salaried <- news_salaried %>%
group_by(job_profile_current, gender)
current_news_median_job_gender_salaried <- current_news_median_job_gender_salaried %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_news_median_job_gender_salaried)
```

```r
# A tibble: 23 x 4
# Groups: job_profile_current [15]
## job_profile_current gender count median
## <chr> <chr> <int> <dbl>
## 1 300113 - Columnist Male 8 175984.
## 2 330113 - Editorial Writer Male 5 164900.
## 3 320113 - Critic Male 5 160780
## 4 300113 - Columnist Female 11 154780
## 5 300313 - Columnist - Editorial Male 5 151896.
## 6 280212 - Staff Writer Male 170 128440.
## 7 280212 - Staff Writer Female 136 113474.
## 8 390510 - Graphics Editor Male 5 111071
## 9 360114 - Photographer Male 11 109928.
## 10 280226 - Video Journalist Male 8 98555
## # ... with 13 more rows

current_news_median_job_gender_hourly <- news_hourly %>%
group_by(job_profile_current, gender)
current_news_median_job_gender_hourly <- current_news_median_job_gender_hourly %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_news_median_job_gender_hourly)
```

```r
# A tibble: 7 x 4
# Groups: job_profile_current [5]
## job_profile_current gender count median
## <chr> <chr> <int> <dbl>
## 1 280225 - Producer Male 6 36.7
## 2 397110 - Multiplatform Editor (PT/PTOC) Female 14 36.5
## 3 280225 - Producer Female 12 36.4
## 4 400151 - Administrative Aide Female 6 35.3
## 5 397110 - Multiplatform Editor (PT/PTOC) Male 9 33.4
## 6 380117 - Research Assistant Female 5 31.7
## 7 410251 - Editorial Aide Female 8 21.4

current_news_median_job_race_salaried <- news_salaried %>%
group_by(job_profile_current, race_ethnicity)
current_news_median_job_race_salaried <- current_news_median_job_race_salaried %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
```
suppress_median(current_news_median_job_race_salaried)

## # A tibble: 21 x 4
## # Groups: job_profile_current [14]
## job_profile_current race_ethnicity count median
## <chr> <chr> <int> <dbl>
## 1 300313 - Columnist - Editorial Writer White (United States of America) 6 1.91e5
## 2 300113 - Columnist White (United States of America) 13 1.77e5
## 3 300113 - Columnist Black or African American (United States of America) 5 1.53e5
## 4 320113 - Critic White (United States of America) 8 1.49e5
## 5 280212 - Staff Writer <NA> 21 1.40e5
## 6 330113 - Editorial Writer White (United States of America) 6 1.27e5
## 7 280212 - Staff Writer White (United States of America) 223 1.25e5
## 8 280212 - Staff Writer Black or African American (United States of America) 18 1.22e5
## 9 280212 - Staff Writer Asian (United States of America) 24 1.17e5
## 10 390510 - Graphics Editor White (United States of America) 5 1.11e5
## ... with 11 more rows

current_news_median_job_race_hourly <- news_hourly %>% group_by(job_profile_current, race_ethnicity)
current_news_median_job_race_hourly <- current_news_median_job_race_hourly %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_news_median_job_race_hourly)

## # A tibble: 6 x 4
## # Groups: job_profile_current [5]
## job_profile_current race_ethnicity gender count median
## <chr> <chr> <chr> <int> <dbl>
## 1 280225 - Producer Black or African American (United States of America) Female 5 37.6
## 2 280225 - Producer White (United States of America) Male 8 35.9
## 3 397110 - Multiplatform Editor White (United States of America) Male 18 34.8
## 4 380117 - Research Assistant White (United States of America) Male 5 31.7
## 5 410251 - Editorial Aide White (United States of America) Male 7 21.1
## 6 430117 - News Aide White (United States of America) Male 5 16.5

current_news_median_job_race_gender_salaried <- news_salaried %>% group_by(job_profile_current, race_ethnicity, gender)
current_news_median_job_race_gender_salaried <- current_news_median_job_race_gender_salaried %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_news_median_job_race_gender_salaried)

## # A tibble: 23 x 5
## # Groups: job_profile_current, race_ethnicity [13]
## job_profile_current race_ethnicity gender count median
## <chr> <chr> <chr> <int> <dbl>
## 1 300113 - Columnist White (United States of America) Female 7 2.24e5
## 2 300113 - Columnist White (United States of America) Male 6 1.76e5
## 3 320113 - Critic White (United States of America) Male 5 1.61e5
## 4 280212 - Staff Writer <NA> Male 11 1.40e5
## 5 280212 - Staff Writer <NA> Female 10 1.38e5
## 6 280212 - Staff Writer White (United States of America) Male 130 1.29e5
## 7 280212 - Staff Writer Black or African American (United States of America) Male 13 1.25e5

71
## 8 280212 - Staff Writer Asian (United States of America) Male 9 1.19e5
## 9 280212 - Staff Writer Asian (United States of America) Female 15 1.15e5
## 10 280212 - Staff Writer White (United States of America) Female 93 1.15e5
## ... with 13 more rows

current_news_median_job_race_gender_hourly <- news_hourly %>%
group_by(job_profile_current, race_ethnicity, gender) %>%
summarise(count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE))
suppress_median(current_news_median_job_race_gender_hourly)

## A tibble: 4 x 5
## Groups: job_profile_current, race_ethnicity [3]
## job_profile_current race_ethnicity gender count median
## <chr> <chr> <chr> <int> <dbl>
## 1 397110 - Multiplatform Editor white Female 10 39.9
## 2 280225 - Producer white Female 5 34.2
## 3 397110 - Multiplatform Editor white Male 8 33.4
## 4 410251 - Editorial Aide White (United States o Female 5 21.1

current_news_median_job_race_group_gender_salaried <- news_salaried %>%
group_by(job_profile_current, race_grouping, gender) %>%
summarise(count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE))
suppress_median(current_news_median_job_race_group_gender_salaried)

## A tibble: 27 x 5
## Groups: job_profile_current, race_grouping [16]
## job_profile_current race_grouping gender count median
## <chr> <chr> <chr> <int> <dbl>
## 1 300113 - Columnist white Female 7 224461.
## 2 300113 - Columnist white Male 6 175984.
## 3 320113 - Critic white Male 5 160780
## 4 280212 - Staff Writer unknown Male 14 137890
## 5 280212 - Staff Writer unknown Female 11 135000
## 6 280212 - Staff Writer white Male 130 129280
## 7 280212 - Staff Writer person of color Male 26 124540
## 8 280212 - Staff Writer white Female 93 115000
## 9 360114 - Photographer white Male 7 113757.
## 10 280226 - Video Journalist white Male 6 106500
## ... with 17 more rows

current_news_median_job_race_group_gender_hourly <- news_hourly %>%
group_by(job_profile_current, race_grouping) %>%
summarise(count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE))
suppress_median(current_news_median_job_race_group_gender_hourly)

## A tibble: 5 x 5
## Groups: job_profile_current, race_grouping [4]
## job_profile_current race_grouping gender count median
## <chr> <chr> <chr> <int> <dbl>
## 1 397110 - Multiplatform Editor (PT/PTO white Female 10 39.9
current_news_median_job_race_gender_age5_salaried <- news_salaried %>% group_by(job_profile_current, race_ethnicity, gender, age_group_5)
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
}
suppress_median(current_news_median_job_race_gender_age5_salaried)

# A tibble: 25 x 6
# Groups: job_profile_current, race_ethnicity, gender [7]
## job_profile_current race_ethnicity gender age_group_5 count median
## <chr> <chr> <chr> <fct> <int> <dbl>
## 1 280212 - Staff Writer White (United States) Male 65+ 5 1.59e5 3
## 2 280212 - Staff Writer White (United States) Male 55-59 17 1.54e5 3
## 3 280212 - Staff Writer White (United States) Female 55-59 7 1.54e5 3
## 4 280212 - Staff Writer White (United States) Female 45-49 10 1.45e5 3
## 5 280212 - Staff Writer White (United States) Male 60-64 11 1.35e5 3
## 6 280212 - Staff Writer White (United States) Male 40-44 20 1.33e5 3
## 7 280212 - Staff Writer White (United States) Male 50-54 14 1.32e5 3
## 8 280212 - Staff Writer White (United States) Male 45-49 9 1.31e5 3
## 9 280212 - Staff Writer White (United States) Female 60-64 6 1.28e5 3
## # ... with 15 more rows

current_news_median_job_race_gender_age5_hourly <- news_hourly %>% group_by(job_profile_current, race_ethnicity, gender, age_group_5)
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
}
suppress_median(current_news_median_job_race_gender_age5_hourly)

# A tibble: 0 x 6
# Groups: job_profile_current, race_ethnicity, gender [0]
# ... with 6 variables: job_profile_current <chr>, race_ethnicity <chr>,
# gender <chr>, age_group_5 <fct>, count <int>, median <dbl>

current_news_median_job_race_group_gender_age5_salaried <- news_salaried %>% group_by(job_profile_current, race_grouping, gender, age_group_5)
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
}
suppress_median(current_news_median_job_race_group_gender_age5_salaried)

# A tibble: 29 x 6
# Groups: job_profile_current, race_grouping, gender [9]
## job_profile_current race_grouping gender age_group_5 count median
## <chr> <chr> <chr> <fct> <int> <dbl>
## 1 280212 - Staff Writer white Male 65+ 5 159458.
## 2 280212 - Staff Writer white Male 55-59 17 153923.
## 3 280212 - Staff Writer white Female 55-59 7 153780
## 4 280212 - Staff Writer white Female 45-49 10 144560.
## 5 280212 - Staff Writer white Female 40-44 9 140000

73
## 6 280212 - Staff Writer white Male 60-64 11 134957.
## 7 280212 - Staff Writer white Male 40-44 20 132980.
## 8 280212 - Staff Writer white Male 50-54 14 132273.
## 9 280212 - Staff Writer white Male 45-49 9 130845
## 10 280212 - Staff Writer white Female 60-64 6 128441.
## # ... with 19 more rows

```r
current_news_median_job_race_group_gender_age5_hourly <- news_hourly %>%
group_by(job_profile_current, race_grouping, gender, age_group_5)
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)

suppress_median(current_news_median_job_race_group_gender_age5_hourly)
```

Performance evaluations

```r
current_news_median_job_race_group_gender_age5_hourly <- current_news_median_job_race_group_gender_age5_hourly
  %>%
  summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))
```

```r
# A tibble: 0 x 6
# Groups: job_profile_current, race_grouping, gender [0]
# ... with 6 variables: job_profile_current <chr>, race_grouping <chr>,
# gender <chr>, age_group_5 <fct>, count <int>, median <dbl>
```

```r
news_ratings <- filter(ratings_combined, dept == 'News')

news_ratings_gender <- news_ratings %>%
group_by(gender)
news_ratings_gender <- news_ratings_gender %>%
  summarise(count = length(performance_rating),
             median = median(performance_rating))

suppress_median(news_ratings_gender)
```

```r
# A tibble: 2 x 3
# gender count median
# <chr> <int> <dbl>
# 1 Female 1892 NA
# 2 Male 1772 NA
```

```r
news_ratings_race <- news_ratings %>%
group_by(race_ethnicity)
news_ratings_race <- news_ratings_race %>%
  summarise(count = length(performance_rating),
             median = median(performance_rating, na.rm = TRUE))

suppress_median(news_ratings_race)
```

```r
# A tibble: 9 x 3
# race_ethnicity count median
# <chr> <int> <dbl>
# 1 American Indian or Alaska Native (United States of America) 12 3.6
# 2 <NA> 88 3.6
# 3 White (United States of America) 2516 3.5
# 4 Asian (United States of America) 324 3.4
# 5 Prefer Not to Disclose (United States of America) 56 3.4
# 6 Black or African American (United States of America) 416 3.3
# 7 Hispanic or Latino (United States of America) 164 3.3
# 8 Native Hawaiian or Other Pacific Islander (United States of America) 8 3.3
# 9 Two or More Races (United States of America) 80 3.2
```
```r
news_ratings_race_gender <- news_ratings %>%
group_by(race_ethnicity, gender)

news_ratings_race_gender <- news_ratings_race_gender %>%
  summarise(
    count = length(performance_rating),
    median = median(performance_rating, na.rm = TRUE)
  )
suppress(news_ratings_race_gender)

## # A tibble: 16 x 4
## # Groups: race_ethnicity [9]
## race_ethnicity gender count median
## <chr>          <chr> <int> <dbl>
## 1 American Indian or Alaska Native (United States of A~ Female 8 3.7
## 2 Asian (United States of America) Female 232 3.4
## 3 Asian (United States of America) Male 92 3.4
## 4 Black or African American (United States of America) Female 224 3.25
## 5 Black or African American (United States of America) Male 192 3.3
## 6 Hispanic or Latino (United States of America) Female 80 3.3
## 7 Hispanic or Latino (United States of America) Male 84 3.3
## 8 Native Hawaiian or Other Pacific Islander (United S~ Male 8 3.3
## 9 Prefer Not to Disclose (United States of America) Female 24 3.5
## 10 Prefer Not to Disclose (United States of America) Male 32 3.3
## 11 Two or More Races (United States of America) Female 52 3.2
## 12 Two or More Races (United States of America) Male 28 3.2
## 13 White (United States of America) Female 1228 3.4
## 14 White (United States of America) Male 1288 3.5
## 15 <NA> Female 44 3.7
## 16 <NA> Male 44 3.55

news_ratings_race_gender_under3 <- filter(news_ratings, performance_rating < 3.1) %>%
group_by(race_grouping, gender)

news_ratings_race_gender_under3 <- news_ratings_race_gender_under3 %>%
  summarise(
    count = length(performance_rating),
    median = median(performance_rating, na.rm = TRUE)
  )
suppress(news_ratings_race_gender_under3)

## # A tibble: 4 x 4
## # Groups: race_grouping [2]
## race_grouping gender count median
## <chr>          <chr> <int> <dbl>
## 1 person of color Female 57 3
## 2 person of color Male 49 3
## 3 white Female 92 3
## 4 white Male 80 3

news_ratings_race_gender_over4 <- filter(news_ratings, performance_rating > 3.9) %>%
group_by(race_grouping)

news_ratings_race_gender_over4 <- news_ratings_race_gender_over4 %>%
  summarise(
    count = length(performance_rating),
    median = median(performance_rating, na.rm = TRUE)
  )
suppress(news_ratings_race_gender_over4)

## # A tibble: 6 x 4
## # Groups: race_grouping [3]
## race_grouping gender count median
## <chr>          <chr> <int> <dbl>
```
## 1 person of color Female 13 4.1
## 2 person of color Male 5 4.1
## 3 unknown Female 5 4.1
## 4 unknown Male 10 4.05
## 5 white Female 67 4.1
## 6 white Male 114 4.2

Pay changes

```r
news_change <- filter(reason_for_change_combined, dept == 'News')

news_change_gender <- news_change %>%
  group_by(business_process_reason, gender) %>%
  summarise(count = length(business_process_reason))

# A tibble: 37 x 3
# Groups: business_process_reason [19]
## business_process_reason gender count
## <chr> <chr> <int>
## 1 Data Change > Data Change > Change Job Details Female 282
## 2 Data Change > Data Change > Change Job Details Male 245
## 3 Hire Employee > New Hire > Conversion Female 1
## 4 Hire Employee > New Hire > Conversion Male 1
## 5 Hire Employee > New Hire > Convert Contingent Female 4
## 6 Hire Employee > New Hire > Convert Contingent Male 1
## 7 Hire Employee > New Hire > Fill Vacancy Female 70
## 8 Hire Employee > New Hire > Fill Vacancy Male 55
## 9 Hire Employee > New Hire > New Position Female 78
## 10 Hire Employee > New Hire > New Position Male 58
# ... with 27 more rows

news_change_race <- news_change %>%
  group_by(business_process_reason, race_ethnicity) %>%
  summarise(count = length(business_process_reason))

# A tibble: 70 x 3
# Groups: business_process_reason [14]
## business_process_reason race_ethnicity count
## <chr> <chr> <int>
## 1 <NA> White (United States of America) 7232
## 2 <NA> Black or African American (United States of America) 1167
## 3 Request Compensation Change > Adjusted Compensation for White (United States of America) 1164
## 4 <NA> Asian (United States of America) 918
## 5 Merit > Performance > Annual Performance White (United States of America) 889
## 6 <NA> Hispanic or Latino (United States of America) 484
## 7 Data Change > Data Change > Change Job Details White (United States of America) 345
## 8 <NA> Two or More Races (United States of America) 274
## 9 <NA> <NA> 207
## 10 Transfer > Transfer > Move to another location White (United States of America) 201
# ... with 60 more rows
```
news_change_race_gender <- news_change %>% group_by(business_process_reason, race_ethnicity, gender)
news_change_race_gender <- news_change_race_gender %>% summarise(
  count = length(business_process_reason)
)
suppress_count(news_change_race_gender)

## # A tibble: 107 x 4
## # Groups: business_process_reason, race_ethnicity [62]
## business_process_reason race_ethnicity gender count
## <chr> <chr> <chr> <int>
## 1 <NA> White (United States of A~ Male 3680
## 2 <NA> White (United States of A~ Female 3552
## 3 <NA> Asian (United States of A~ Female 702
## 4 <NA> Black or African American- Female 612
## 5 Request Compensation Change > A- White (United States of A~ Male 606
## 6 Request Compensation Change > A- White (United States of A~ Female 558
## 7 <NA> Black or African American- Male 555
## 8 Merit > Performance > Annual Pe- White (United States of A~ Male 476
## 9 Merit > Performance > Annual Pe- White (United States of A~ Female 413
## 10 <NA> Hispanic or Latino (Unite- Female 250
## # ... with 97 more rows

Performance evaluations x merit raises

reason_for_change_combined <- reason_for_change_combined %>% mutate(merit_raises = grepl('*Merit*', business_process_reason))
twenty14 = as.Date('2016-04-01')
twenty15 = as.Date('2017-04-01')
twenty16 = as.Date('2018-04-01')
twenty17 = as.Date('2019-04-01')
twenty18 = as.Date('2020-04-01')

reason_for_change_combined <- reason_for_change_combined %>%
  mutate(raise_after = case_when(
    effective_date < twenty14 ~ 'before 2015',
    effective_date < twenty15 ~ '2015',
    effective_date < twenty16 ~ '2016',
    effective_date < twenty17 ~ '2017',
    effective_date < twenty18 ~ '2018',
    TRUE ~ 'Other'))

merit_raises_news_gender_salaried <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'News', dept == 'News', pay_rate_type == 'Salaried')
merit_raises_news_gender_salaried <- merit_raises_news_gender_salaried %>% summarise(
  count = length(base_pay_change),
  median = median(base_pay_change, na.rm = TRUE)
)
suppress(merit_raises_news_gender_salaried)

## # A tibble: 2 x 3
## gender count median
## <chr> <int> <dbl>
## 1 Female 431 3000
## 2 Male 494 3000
merit_raises_news_gender_hourly <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'News', count = length(base_pay_change),
median = median(base_pay_change, na.rm = TRUE))
}
suppress(merit_raises_news_gender_hourly)

## # A tibble: 2 x 3
## gender count median
## <chr> <int> <dbl>
## 1 Female 78 1.27
## 2 Male 51 1.03

merit_raises_news_race_salaried <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'News', pay_rate_type == 'Salaried', count = length(base_pay_change),
median = median(base_pay_change, na.rm = TRUE))
)
suppress_median(merit_raises_news_race_salaried)

## # A tibble: 7 x 3
## race_ethnicity count median
## <chr> <int> <dbl>
## 1 American Indian or Alaska Native (United States of America) 5 3500
## 2 Two or More Races (United States of America) 7 3500
## 3 <NA> 14 3500
## 4 Asian (United States of America) 69 3000
## 5 Black or African American (United States of America) 82 3000
## 6 White (United States of America) 707 3000
## 7 Hispanic or Latino (United States of America) 36 2500

merit_raises_news_race_hourly <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'News', pay_rate_type == 'Hourly', count = length(base_pay_change),
median = median(base_pay_change, na.rm = TRUE))
)
suppress_median(merit_raises_news_race_hourly)

## # A tibble: 3 x 3
## race_ethnicity count median
## <chr> <int> <dbl>
## 1 White (United States of America) 91 1.28
## 2 Black or African American (United States of America) 16 1.25
## 3 Asian (United States of America) 18 1.03

merit_raises_news_race_group_salaried <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'News', pay_rate_type == 'Salaried', count = length(base_pay_change),
median = median(base_pay_change, na.rm = TRUE))
)
suppress_median(merit_raises_news_race_group_salaried)

## # A tibble: 3 x 3
## race_grouping count median
## <chr> <int> <dbl>
## 1 person of color 200 3000
merit_raises_news_race_group_hourly <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'News', pay_rate_type == 'Hourly')
merit_raises_news_race_group_hourly <- merit_raises_news_race_group_hourly %>% summarise(count = length(base_pay_change), median = median(base_pay_change, na.rm = TRUE))
suppress_median(merit_raises_news_race_group_hourly)

merit_raises_news_gender_race_group_salaried <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'News', pay_rate_type == 'Salaried')
merit_raises_news_gender_race_group_salaried <- merit_raises_news_gender_race_group_salaried %>% summarise(count = length(base_pay_change), median = median(base_pay_change, na.rm = TRUE))
suppress_median(merit_raises_news_gender_race_group_salaried)

merit_raises_news_gender_race_group_hourly <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'News', pay_rate_type == 'Hourly')
merit_raises_news_gender_race_group_hourly <- merit_raises_news_gender_race_group_hourly %>% summarise(count = length(base_pay_change), median = median(base_pay_change, na.rm = TRUE))
suppress_median(merit_raises_news_gender_race_group_hourly)

fifteen_raises_amount <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'News', pay_rate_type == 'Salaried', raise_after == 2015)
fifteen_raises_amount <- fifteen_raises_amount %>% summarise(count = length(base_pay_change), median_raise = median(base_pay_change, na.rm = TRUE))
suppress(fifteen_raises_amount)
## # A tibble: 4 x 4
## # Groups: race_grouping [2]
## race_grouping gender count median_raise
## <chr> <chr> <int> <dbl>
## 1 person of color Female 17 2888
## 2 person of color Male 10 2162.
## 3 white Female 44 2500
## 4 white Male 64 3000

fifteen_raises_score <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'News', pay_rate_type == 'Salaried', raise_after == 2015)

fifteen_raises_score <- fifteen_raises_score %>%
  group_by(race_grouping, gender)

fifteen_raises_score <- fifteen_raises_score %>%
  summarise(count = length(2015_annual_performance_rating),
             median = median(2015_annual_performance_rating, na.rm = TRUE))

suppress(fifteen_raises_score)

## # A tibble: 0 x 4
## # Groups: race_grouping [0]
## # ... with 4 variables: race_grouping <chr>, gender <chr>, count <int>,
## # median <chr>

sixteen_raises_amount <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'News', pay_rate_type == 'Salaried', raise_after == 2016)

sixteen_raises_amount <- sixteen_raises_amount %>%
  group_by(race_grouping, gender)

sixteen_raises_amount <- sixteen_raises_amount %>%
  summarise(count = length(base_pay_change),
             median_raise = median(base_pay_change, na.rm = TRUE))

suppress(sixteen_raises_amount)

## # A tibble: 4 x 4
## # Groups: race_grouping [2]
## race_grouping gender count median_raise
## <chr> <chr> <int> <dbl>
## 1 person of color Female 26 3000
## 2 person of color Male 17 3000
## 3 white Female 60 3000
## 4 white Male 81 3000

sixteen_raises_score <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'News', pay_rate_type == 'Salaried', raise_after == 2016)

sixteen_raises_score <- sixteen_raises_score %>%
  summarise(count = length(2016_annual_performance_rating),
             median = median(2016_annual_performance_rating, na.rm = TRUE))

suppress(sixteen_raises_score)

## # A tibble: 0 x 4
## # Groups: race_grouping [0]
## # ... with 4 variables: race_grouping <chr>, gender <chr>, count <int>,
## # median <chr>

seventeen_raises_amount <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'News', pay_rate_type == 'Salaried', raise_after == 2017)

seventeen_raises_amount <- seventeen_raises_amount %>%
  group_by(race_grouping, gender)

seventeen_raises_amount <- seventeen_raises_amount %>%
  summarise(count = length(base_pay_change),
             median_raise = median(base_pay_change, na.rm = TRUE))

suppress(seventeen_raises_amount)

## # A tibble: 4 x 4
## # Groups: race_grouping [2]
## race_grouping gender count median_raise
## <chr> <chr> <int> <dbl>
## 1 person of color Female 17 2888
## 2 person of color Male 10 2162.
## 3 white Female 44 2500
## 4 white Male 64 3000

suppress(seventeen_raises_amount)
median = median('2017_annual_performance_rating', na.rm = TRUE))
suppress(seventeen_raises_score)

median_raise = median(base_pay_change, na.rm = TRUE))
suppress(seventeen_raises_amount)

median = median('2018_annual_performance_rating', na.rm = TRUE))
suppress(eighteen_raises_score)

median_raise = median(base_pay_change, na.rm = TRUE))
suppress(eighteen_raises_amount)

merit.raises_15 <- filter(reason_for_change_combined, raise_after == '2015', merit.raises == TRUE)
merit.raises_16 <- filter(reason_for_change_combined, raise_after == '2016', merit.raises == TRUE)
merit.raises_17 <- filter(reason_for_change_combined, raise_after == '2017', merit.raises == TRUE)
merit.raises_18 <- filter(reason_for_change_combined, raise_after == '2018', merit.raises == TRUE)

names(merit_raises_15) <- c('base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping')
names(merit_raises_16) <- c('base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping')
names(merit_raises_17) <- c('base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping')
names(merit_raises_18) <- c('base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping')

merit_raises_combined <- rbind(merit_raises_15, merit_raises_16, merit_raises_17, merit_raises_18)

news_salaried_raises <- filter(merit_raises_combined, pay_rate_type == 'Salaried', dept == 'News') %>%
  group_by(race_grouping, gender) %>%
  summarise(count = length(base_pay_change),
             median = median(base_pay_change, na.rm = TRUE))

news_salaried_raises_scores <- filter(merit_raises_combined, pay_rate_type == 'Salaried', dept == 'News') %>%
  group_by(race_grouping, gender) %>%
  summarise(count = length(base_pay_change),
             median = median(base_pay_change, na.rm = TRUE))

news_hourly_raises <- filter(merit_raises_combined, pay_rate_type == 'Hourly', dept == 'News') %>%
  group_by(race_grouping, gender) %>%
  summarise(count = length(base_pay_change),
             median = median(base_pay_change, na.rm = TRUE))

news_hourlyraises_scores <- filter(merit_raises_combined, pay_rate_type == 'Hourly', dept == 'News') %>%
  group_by(race_grouping, gender) %>%
  summarise(count = length(base_pay_change),
             median = median(base_pay_change, na.rm = TRUE))

## # A tibble: 4 x 4
## # Groups: race_grouping [2]
## race_grouping gender count median
## <chr> <chr> <int> <dbl>
## 1 person of color Female 18 1.27
## 2 person of color Male 78 3.4
## 3 unknown Female 9 3.9
## 4 unknown Male 7 3.7

news_hourly_raises_scores <- filter(merit_raises_combined, pay_rate_type == 'Salaried', dept == 'News') %>%
  group_by(race_grouping, gender) %>%
  summarise(count = length(base_pay_change),
             median = median(base_pay_change, na.rm = TRUE))

## # A tibble: 6 x 4
## # Groups: race_grouping [3]
## race_grouping gender count median
## <chr> <chr> <int> <dbl>
## 1 person of color Female 96 3000
## 2 person of color Male 78 2659.
## 3 unknown Female 9 3000
## 4 unknown Male 7 2500
## 5 white Female 267 3000
## 6 white Male 354 3000

news_salaried_raises_scores <- filter(merit_raises_combined, pay_rate_type == 'Salaried', dept == 'News') %>%
  group_by(race_grouping, gender) %>%
  summarise(count = length(base_pay_change),
             median = median(base_pay_change, na.rm = TRUE))

## # A tibble: 6 x 4
## # Groups: race_grouping [3]
## race_grouping gender count median
## <chr> <chr> <int> <dbl>
## 1 person of color Female 96 3000
## 2 person of color Male 78 2659.
## 3 unknown Female 9 3000
## 4 unknown Male 7 2500
## 5 white Female 267 3000
## 6 white Male 354 3000
## 2 person of color Male 19 1.03
## 3 white Female 54 1.46
## 4 white Male 28 1.16

```r
define

```
```
suppress_median(bezos_race)

## # A tibble: 7 x 3
## race_ethnicity count median
## <chr> <int> <dbl>
## 1 <NA> 12 130000
## 2 Black or African American (United States of America) 26 94964.
## 3 White (United States of America) 224 94519.
## 4 Asian (United States of America) 31 87000
## 5 Prefer Not to Disclose (United States of America) 8 82140
## 6 Hispanic or Latino (United States of America) 22 81250.
## 7 Two or More Races (United States of America) 14 79860

graham_race <- graham %>% group_by(race_ethnicity)
graham_race <— graham_race %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(graham_race)

## # A tibble: 5 x 3
## race_ethnicity count median
## <chr> <int> <dbl>
## 1 <NA> 9 151171.
## 2 Hispanic or Latino (United States of America) 6 135272.
## 3 White (United States of America) 182 124500
## 4 Asian (United States of America) 15 111761.
## 5 Black or African American (United States of America) 22 104398.

bezos_race_group <- bezos %>% group_by(race_grouping)
bezos_race_group <— bezos_race_group %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(bezos_race_group)

## # A tibble: 3 x 3
## race_grouping count median
## <chr> <int> <dbl>
## 1 unknown 20 113890
## 2 white 224 94519.
## 3 person of color 93 86000

graham_race_group <- graham %>% group_by(race_grouping)
graham_race_group <— graham_race_group %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(graham_race_group)

## # A tibble: 3 x 3
## race_grouping count median
## <chr> <int> <dbl>
## 1 unknown 9 151171.
## 2 white 182 124500
## 3 person of color 46 110845.
bezios_gender_race_group <- bezos %>% group_by(race_grouping, gender)
bezios_gender_race_group <- bezios_gender_race_group %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(bezos_gender_race_group)

## # A tibble: 6 x 4
## # Groups: race_grouping 
## race_grouping gender count median
## <chr> <chr> <int> <dbl>
## 1 unknown Male 10 121390
## 2 unknown Female 10 109000
## 3 white Male 115 102780
## 4 person of color Male 32 94026.
## 5 white Female 109 88780
## 6 person of color Female 61 82000

graham_gender_race_group <- graham %>% group_by(race_grouping, gender)
graham_gender_race_group <- graham_gender_race_group %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(graham_gender_race_group)

## # A tibble: 5 x 4
## # Groups: race_grouping 
## race_grouping gender count median
## <chr> <chr> <int> <dbl>
## 1 unknown Male 6 150975.
## 2 white Male 103 128629.
## 3 person of color Male 24 117567.
## 4 white Female 79 112512.
## 5 person of color Female 22 108594.

bezios_gender_race_group_age5 <- bezos %>% group_by(race_grouping, gender, age_group_5)
bezios_gender_race_group_age5 <- bezios_gender_race_group_age5 %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(bezos_gender_race_group_age5)

## # A tibble: 20 x 5
## # Groups: race_grouping, gender 
## race_grouping gender age_group_5 count median
## <chr> <chr> <fct> <int> <dbl>
## 1 white Female 45-49 7 160780
## 2 white Male 55-59 8 156807.
## 3 white Female 40-44 6 143750
## 4 white Male 40-44 15 136468.
## 5 person of color Male 35-39 8 115530
## 6 white Female 50-54 8 114975.
## 7 white Male 35-39 24 107880
## 8 white Female 35-39 15 105000
## 9 white Male 45-49 9 102796.
<table>
<thead>
<tr>
<th>#</th>
<th>Race Grouping</th>
<th>Gender</th>
<th>Age Group</th>
<th>Count</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>person of color</td>
<td>Female</td>
<td>35-39</td>
<td>8</td>
<td>99619.</td>
</tr>
<tr>
<td>11</td>
<td>white</td>
<td>Male</td>
<td>30-34</td>
<td>29</td>
<td>94780</td>
</tr>
<tr>
<td>12</td>
<td>person of color</td>
<td>Male</td>
<td>25-29</td>
<td>8</td>
<td>88540</td>
</tr>
<tr>
<td>13</td>
<td>white</td>
<td>Female</td>
<td>30-34</td>
<td>24</td>
<td>87005</td>
</tr>
<tr>
<td>14</td>
<td>person of color</td>
<td>Female</td>
<td>30-34</td>
<td>19</td>
<td>87000</td>
</tr>
<tr>
<td>15</td>
<td>person of color</td>
<td>Male</td>
<td>30-34</td>
<td>5</td>
<td>87000</td>
</tr>
<tr>
<td>16</td>
<td>white</td>
<td>Female</td>
<td>25-29</td>
<td>37</td>
<td>81757.</td>
</tr>
<tr>
<td>17</td>
<td>person of color</td>
<td>Female</td>
<td>25-29</td>
<td>19</td>
<td>77000</td>
</tr>
<tr>
<td>18</td>
<td>white</td>
<td>Male</td>
<td>25-29</td>
<td>21</td>
<td>76780</td>
</tr>
<tr>
<td>19</td>
<td>person of color</td>
<td>Female</td>
<td>&lt;25</td>
<td>10</td>
<td>64390</td>
</tr>
<tr>
<td>20</td>
<td>white</td>
<td>Female</td>
<td>&lt;25</td>
<td>9</td>
<td>64280</td>
</tr>
</tbody>
</table>

```r
graham_gender_race_group_age5 <- graham %>% group_by(race_grouping, gender, age_group_5) %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(graham_gender_race_group_age5)
```

```r
# A tibble: 18 x 5
# Groups: race_grouping, gender [4]
# race_grouping gender age_group_5 count median
# <chr> <chr> <fct> <int> <dbl>
# 1 white Male 65+ 8 153937.
# 2 white Male 35-39 11 147300
# 3 white Male 55-59 19 146542.
# 4 white Female 55-59 16 138564.
# 5 white Male 50-54 21 134547.
# 6 white Male 60-64 14 123515.
# 7 white Female 40-44 5 120780
# 8 person of color Female 40-44 5 118512.
# 9 person of color Male 50-54 11 116349.
# 10 white Male 40-44 17 115237.
# 11 white Female 50-54 15 114803
# 12 white Female 60-64 7 112512.
# 13 white Male 45-49 8 111473.
# 14 white Female 45-49 12 100910.
# 15 white Female 30-34 8 100788.
# 16 person of color Female 50-54 5 96944.
# 17 white Female 35-39 11 88000
# 18 white Male 30-34 5 83650.
```

```r
bezos_gender_race_group_age5_tier <- bezos %>% group_by(race_grouping, gender, age_group_5, tier) %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(bezos_gender_race_group_age5_tier)
```

```r
# A tibble: 20 x 6
# Groups: race_grouping, gender, age_group_5 [10]
# race_grouping gender age_group_5 tier count median
# <chr> <chr> <fct> <chr> <int> <dbl>
# 1 white Male 40-44 Tier 1 8 191530
# 2 white Male 35-39 Tier 1 10 130018.
```
## 3 white Female 35-39 Tier 1 8 128330
## 4 white Male 30-34 Tier 1 12 125233.
## 5 white Male 45-49 Tier 2 5 120780
## 6 white Female 25-29 Tier 1 5 100000
## 7 white Male 30-34 Tier 2 5 100000
## 8 white Male 35-39 Tier 2 8 98890
## 9 white Male 30-34 Tier 2 6 93780
## 10 white Male 35-39 Tier 3 6 93030
## 11 white Male 25-29 Tier 2 6 91282.
## 12 white Female 25-29 Tier 2 6 91000
## 13 white Male 30-34 Tier 3 10 88240
## 14 person of color Female 30-34 Tier 2 7 88133.
## 15 white Female 30-34 Tier 3 11 86000
## 16 person of color Female 30-34 Tier 3 6 83890.
## 17 white Female 25-29 Tier 3 15 79140
## 18 person of color Female 25-29 Tier 3 12 77000
## 19 white Male 25-29 Tier 3 8 73890
## 20 white Female 25-29 Tier 4 8 69890

graham_gender_race_group_age5_tier <- graham %>%
group_by(race_grouping, gender, age_group_5, tier)
graham_gender_race_group_age5_tier <- graham_gender_race_group_age5_tier %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(graham_gender_race_group_age5_tier)

## # A tibble: 21 x 6
## # Groups: race_grouping, gender, age_group_5 [13]
## race_grouping gender age_group_5 tier count median
## <chr> <chr> <fct> <chr> <int> <dbl>
## 1 white Male 50-54 Tier 1 5 180040
## 2 white Male 35-39 Tier 1 5 173280
## 3 white Female 50-54 Tier 1 5 167780
## 4 white Male 55-59 Tier 1 5 167172.
## 5 white Male 60-64 Tier 1 5 166612.
## 6 white Female 55-59 Tier 1 5 162854.
## 7 white Female 55-59 Tier 2 5 149030.
## 8 white Male 65+ Tier 2 5 147473.
## 9 white Male 35-39 Tier 2 5 147300
## 10 white Male 55-59 Tier 2 5 143129.
## ... with 11 more rows

Overall disparity calculations

news_groups <- news_salaried %>%
group_by(age_group_5, tier)
news_groups <- news_groups %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(news_groups)

## # A tibble: 30 x 4
## # Groups: age_group_5 [10]
## age_group_5 tier count median
## <fct> <chr> <int> <dbl>

87
# 1 <25 Tier 2 10 65140
# 2 <25 Tier 3 8 66250
# 3 25-29 Tier 1 13 110000
# 4 25-29 Tier 2 23 90000
# 5 25-29 Tier 3 38 77000
# 6 25-29 Tier 4 17 75000
# 7 30-34 Tier 1 35 121280
# 8 30-34 Tier 2 27 94535
# 9 30-34 Tier 3 32 83140
## 10 30-34 Tier 4 9 77000
## # ... with 20 more rows
expected_medians <- merge(news_salaried, news_groups, by=c('age_group_5', 'tier'), all.x = TRUE)

below_expected_medians <- filter(expected_medians, current_base_pay < median) %>%
group_by(race_grouping, gender) %>%
summarise(count = length(current_base_pay))
suppress(below_expected_medians)

## # A tibble: 6 x 3
## # Groups: race_grouping 
## race_grouping gender count
## <chr> <chr> <int>
## 1 person of color Female 48
## 2 person of color Male 30
## 3 unknown Female 7
## 4 unknown Male 8
## 5 white Female 94
## 6 white Male 89

above_expected_medians <- filter(expected_medians, current_base_pay > median) %>%
group_by(race_grouping, gender) %>%
summarise(count = length(current_base_pay))
suppress(above_expected_medians)

## # A tibble: 5 x 3
## # Groups: race_grouping 
## race_grouping gender count
## <chr> <chr> <int>
## 1 person of color Female 30
## 2 person of color Male 21
## 3 unknown Male 8
## 4 white Female 90
## 5 white Male 121

expected_medians <- expected_medians %>%
mutate(disparity = current_base_pay - median,
       disparity_pct = (current_base_pay - median)/median)

disparity <- expected_medians %>%
group_by(race_grouping, gender)
disparity <- disparity %>%
summarise(count = length(disparity),
          median_disparity = median(disparity, na.rm = TRUE))
suppress(disparity)
## A tibble: 6 x 4
## Groups: race_grouping [3]
## race_grouping  gender count median_disparity
## <chr>         <chr> <int>     <dbl>
## 1 person of color Female   83  -1360
## 2 person of color Male     56  -407.
## 3 unknown                  Female  13  -1300
## 4 unknown                  Male    16   1537.
## 5 white                    Female  188  -14.2
## 6 white                    Male     218   2448.

disparity_pct_above <- filter(expected_medians, disparity_pct > .05) %>%
  group_by(race_grouping, gender) %>%
  summarise(count = length(disparity),
             median_disparity = median(disparity, na.rm = TRUE))
suppress(disparity_pct_above)

## A tibble: 5 x 4
## Groups: race_grouping [3]
## race_grouping  gender count median_disparity
## <chr>         <chr> <int>     <dbl>
## 1 person of color Female   20    9360
## 2 person of color Male     15   24700
## 3 unknown                  Male     7   29500
## 4 white                    Female  65  19211.
## 5 white                    Male    101  28780

disparity_pct_below <- filter(expected_medians, disparity_pct < -.05) %>%
  group_by(race_grouping, gender) %>%
  summarise(count = length(disparity),
             median_disparity = median(disparity, na.rm = TRUE))
suppress(disparity_pct_below)

## A tibble: 5 x 4
## Groups: race_grouping [3]
## race_grouping  gender count median_disparity
## <chr>         <chr> <int>     <dbl>
## 1 person of color Female   36  -10140
## 2 person of color Male     21  -15435
## 3 unknown                  Female   6  -14390
## 4 white                    Female  70  -14589.
## 5 white                    Male    68  -18102.

bezos_news_groups <- bezos %>%
  group_by(age_group_5, tier) %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))
suppress(bezos_news_groups)

## A tibble: 20 x 4
## Groups: age_group_5 [7]
## age_group_5 tier count median
## <fct>        <chr> <int>  <dbl>
## 1 <25 Tier 2 10 65140
## 2 <25 Tier 3 8 66250
## 3 25–29 Tier 1 12 110000
## 4 25–29 Tier 2 23 90000
## 5 25–29 Tier 3 38 77000
## 6 25–29 Tier 4 17 75000
## 7 30–34 Tier 1 28 120843. 
## 8 30–34 Tier 2 19 95656. 
## 9 30–34 Tier 3 30 84640
## 10 30–34 Tier 4 7 77000
## 11 35–39 Tier 1 26 122940
## 12 35–39 Tier 2 16 102801.
## 13 35–39 Tier 3 13 90780
## 14 35–39 Tier 4 7 77000
## 14 40–44 Tier 1 18 148572.
## 15 40–44 Tier 2 6 128713.
## 16 40–44 Tier 3 7 103000
## 17 45–49 Tier 1 13 129780
## 18 45–49 Tier 2 13 104560.
## 19 45–49 Tier 3 6 91234.
## 20 45–49 Tier 4 7 77000
## 21 50–54 Tier 1 13 165685.
## 22 50–54 Tier 2 30 117266.
## 23 50–54 Tier 3 10 100406.
## 24 50–54 Tier 4 5 92352.

bezos_expected_medians <- merge(bezos, bezos_news_groups, by=c('age_group_5', 'tier'), all.x = TRUE)

graham_news_groups <- graham %>% group_by(age_group_5, tier)
graham_news_groups <- graham_news_groups %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(graham_news_groups)

## # A tibble: 20 x 4
## # Groups: age_group_5 [8]
##   age_group_5 tier count median
##   <fct>    <chr> <int> <dbl>
## 1 30-34    Tier 1   7 121280
## 2 30-34    Tier 2   8  89092.
## 3 35-39    Tier 1   9 125000
## 4 35-39    Tier 2  14  95500
## 5 35-39    Tier 3   6  89000.
## 6 40-44    Tier 1  13 129780
## 7 40-44    Tier 2  13 104560.
## 8 45-49    Tier 1   8  95485.
## 9 45-49    Tier 2  12  94653.
## 10 50-54   Tier 1  13 165685.
## 11 50-54   Tier 2  30 117266.
## 12 50-54   Tier 3  10 100406.
## 13 55-59   Tier 1  17 170497.
## 14 55-59   Tier 2  18 143186.
## 15 55-59   Tier 3  6  92226.
## 16 60-64   Tier 1  10 158690.
## 17 60-64   Tier 2  9  912512.
## 18 60-64   Tier 3  5  107212.
## 19 65+     Tier 1  6  172067.
## 20 65+     Tier 2  8  147473.
```r
graham_expected_medians <- merge(graham, graham_news_groups, by=c('age_group_5', 'tier'), all.x = TRUE)

bezos_expected_medians <- bezos_expected_medians %>%
  mutate(disparity = current_base_pay - median,
         disparity_pct = (current_base_pay - median)/median)

graham_expected_medians <- graham_expected_medians %>%
  mutate(disparity = current_base_pay - median,
         disparity_pct = (current_base_pay - median)/median)

bezos_disparity_gender <- bezos_expected_medians %>%
  group_by(gender)
bezos_disparity_gender <- bezos_disparity_gender %>%
  summarise(
    count = length(disparity),
    median_disparity = median(disparity, na.rm = TRUE)
  )

## # A tibble: 2 x 3
##   gender count median_disparity
##   <chr>  <int>            <dbl>
## 1 Female 180            -352.1
## 2 Male   157             66.91

bezos_disparity_race_group <- bezos_expected_medians %>%
  group_by(race_grouping)
bezos_disparity_race_group <- bezos_disparity_race_group %>%
  summarise(
    count = length(disparity),
    median_disparity = median(disparity, na.rm = TRUE)
  )

## # A tibble: 3 x 3
## race_grouping count median_disparity
## <chr>         <int>            <dbl>
## 1 person of color   93              0
## 2 unknown           20            -4536.
## 3 white            224              0

bezos_disparity_gender_race_group <- bezos_expected_medians %>%
  group_by(race_grouping, gender)
bezos_disparity_gender_race_group <- bezos_disparity_gender_race_group %>%
  summarise(
    count = length(disparity),
    median_disparity = median(disparity, na.rm = TRUE)
  )

## # A tibble: 6 x 4
## # Groups: race_grouping [3]
## race_grouping gender count median_disparity
## <chr> <chr>  <int>            <dbl>
## 1 person of color Female  61            -590
## 2 person of color Male   32              0
## 3 unknown Female         10            -6070
## 4 unknown Male           10             7453.
## 5 white Female           109              0
## 6 white Male             115             454.

graham_disparity_gender <- graham_expected_medians %>%
  group_by(gender)
graham_disparity_gender <- graham_disparity_gender %>%
  summarise(
    count = length(disparity),
    median_disparity = median(disparity, na.rm = TRUE)
  )
```

91
count = length(disparity),
median_disparity = median(disparity, na.rm = TRUE)
)
suppress(graham_disparity_gender)

## # A tibble: 2 x 3
##   gender count median_disparity
##   <chr> <int>      <dbl>
## 1 Female 104       -905.
## 2 Male   133        475.

graham_disparity_race_group <- graham_expected_medians %>%
group_by(race_grouping)
graham_disparity_race_group <- graham_disparity_race_group %>%
summarise(
  count = length(disparity),
  median_disparity = median(disparity, na.rm = TRUE)
)
suppress(graham_disparity_race_group)

## # A tibble: 3 x 3
## race_grouping count median_disparity
## <chr>       <int>     <dbl>
## 1 person of color 46    -5439.
## 2 unknown      9     -3191.
## 3 white        182     2069.

graham_disparity_gender_race_group <- graham_expected_medians %>%
group_by(race_grouping, gender)
graham_disparity_gender_race_group <- graham_disparity_gender_race_group %>%
summarise(
  count = length(disparity),
  median_disparity = median(disparity, na.rm = TRUE)
)
suppress(graham_disparity_gender_race_group)

## # A tibble: 5 x 4
## Groups: race_grouping [3]
## race_grouping gender count median_disparity
## <chr>         <chr> <int>      <dbl>
## 1 person of color Female 22      -6599.
## 2 person of color Male 24       -1409.
## 3 unknown     Male 6           -2850.
## 4 white       Female 79        810.
## 5 white       Male 103        3355.

Regression

news_salaried_regression <- news_salaried[,c('department', 'gender', 'race_ethnicity', 'current_base_pay'),
news_salaried_regression <- fastDummies::dummy_cols(news_salaried_regression, select_columns = c('gender'),
names(news_salaried_regression) <- gsub('\<', 'under_', names(news_salaried_regression))
names(news_salaried_regression) <- gsub('>', 'over', names(news_salaried_regression))
linearMod1 <- lm(formula = current_base_pay ~ gender_Female + gender_Male, data=news_salaried_regression)
summary(linearMod1)

## # A tibble: 5 x 4
## Groups: race_grouping [3]
## race_grouping gender count median_disparity
## <chr>         <chr> <int>      <dbl>
## 1 person of color Female 22      -6599.
## 2 person of color Male 24       -1409.
## 3 unknown     Male 6           -2850.
## 4 white       Female 79        810.
## 5 white       Male 103        3355.
```r
## Call:
## lm(formula = current_base_pay ~ gender_Female + gender_Male, 
##     data = news_salaried_regression)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -66717  -30572  -10009   22943  207383
##
## Coefficients: (1 not defined because of singularities)
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      124717     2500   49.895 < 2e-16 ***
## gender_Female   -17250      3554    -4.854  1.56e-06 ***
## gender_Male       NA         NA        NA         NA
##
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 42570 on 572 degrees of freedom
## Multiple R-squared:  0.03957,  Adjusted R-squared:  0.03789
## F-statistic: 23.56 on 1 and 572 DF,  p-value: 1.561e-06

equation_2 <- lm(formula = current_base_pay ~ race_grouping_white + race_grouping_person_of_color, data=news_salaried_regression)

## Call:
## lm(formula = current_base_pay ~ race_grouping_white + race_grouping_person_of_color, 
##     data = news_salaried_regression)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -62782  -30287  -11247   24529  211317
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)      127084     7897   16.092  < 2e-16 ***
## race_grouping_white  -6302    8175   -0.771   0.44107
## race_grouping_person_of_color  -26614   8682   -3.065   0.00228 **
##
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 42530 on 571 degrees of freedom
## Multiple R-squared:  0.04295,  Adjusted R-squared:  0.0396
## F-statistic: 12.81 on 2 and 571 DF,  p-value: 3.602e-06

equation_3 <- lm(formula = current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color, data=news_salaried_regression)

## Call:
## lm(formula = current_base_pay ~ gender_Female + gender_Male + 
##     race_grouping_white + race_grouping_person_of_color, data = news_salaried_regression)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -69906  -30002  -9689    22094  204194
##
##```
Coefficients: (1 not defined because of singularities)

| Estimate | Std. Error | t value | Pr(>|t|) |
|----------|------------|---------|----------|
| (Intercept) | 133980 | 7934 | 16.888 | < 2e-16 *** |
| gender_Female | -15384 | 3519 | -4.371 | 1.47e-05 *** |
| gender_Male | NA | NA | NA | NA |
| race_grouping_white | -6075 | 8048 | -0.755 | 0.45069 |
| race_grouping_person_of_color | -24324 | 8564 | -2.840 | 0.00467 ** |

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 41870 on 570 degrees of freedom
Multiple R-squared: 0.074, Adjusted R-squared: 0.06912
F-statistic: 15.18 on 3 and 570 DF, p-value: 1.617e-09

linearMod4 <- lm(formula = current_base_pay ~ gender_Female + gender_Male + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = news_salaried_regression)
summary(linearMod4)

Call:
lm(formula = current_base_pay ~ gender_Female + gender_Male + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = news_salaried_regression)

Residuals:
Min 1Q Median 3Q Max
-86172 -22779 -7300 13780 181520

Coefficients: (2 not defined because of singularities)

| Estimate | Std. Error | t value | Pr(>|t|) |
|----------|------------|---------|----------|
| (Intercept) | 167625 | 9729 | 17.229 | < 2e-16 *** |
| gender_Female | -8165 | 3214 | -2.540 | 0.011345 * |
| gender_Male | NA | NA | NA | NA |
| age_group_5_under_25 | -94875 | 12415 | -7.642 | 9.25e-14 *** |
| age_group_5_25to29 | -75435 | 10489 | -7.192 | 2.05e-12 *** |
| age_group_5_30to34 | -59320 | 10374 | -5.718 | 1.75e-08 *** |
| age_group_5_35to39 | -48805 | 10485 | -4.655 | 4.05e-06 *** |
| age_group_5_40to44 | -30359 | 10760 | -2.821 | 0.004949 ** |
| age_group_5_45to49 | -38200 | 11239 | -3.399 | 0.000724 *** |
| age_group_5_50to54 | -35503 | 10659 | -3.331 | 0.000923 *** |
| age_group_5_55to59 | -19524 | 11005 | -1.774 | 0.076595 |
| age_group_5_60to64 | -25877 | 11987 | -2.159 | 0.031299 * |
| age_group_5_65_over | NA | NA | NA | NA |

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 37450 on 563 degrees of freedom
Multiple R-squared: 0.2682, Adjusted R-squared: 0.2552
F-statistic: 20.63 on 10 and 563 DF, p-value: < 2.2e-16

linearMod5 <- lm(formula = current_base_pay ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = news_salaried_regression)
summary(linearMod5)
Call:
lm(formula = current_base_pay ~ race_grouping_white + race_grouping_person_of_color +
    age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 +
    age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
    age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
    age_group_5_65_over, data = news_salaried_regression)

Residuals:
  Min     1Q    Median     3Q    Max
-83570  -24373     -6835    13690  175683

Coefficients: (1 not defined because of singularities)
             Estimate Std. Error t value Pr(>|t|)
(Intercept)     175496     11714     14.982  < 2e-16 ***
race_grouping_white  -10472     7206      -1.453  0.146732
race_grouping_person_of_color  -22748     7649      -2.974  0.003064 **
age_group_5_under_25  -93548     12337      -7.583  1.41e-13 ***
age_group_5_25to29  -75151     10408      -7.221  1.69e-12 ***
age_group_5_30to34  -58814     10318      -5.700  1.93e-08 ***
age_group_5_35to39  -46772     10444      -4.478  9.11e-06 ***
age_group_5_40to44  -28517     10714      -2.662  0.007999 **
age_group_5_45to49  -37927     11169      -3.396  0.000733 ***
age_group_5_50to54  -33076     10623      -3.114  0.001942 **
age_group_5_55to59  -19411     10936      -1.775  0.076454 .
age_group_5_60to64  -26272     11912      -2.205  0.027827 *
age_group_5_65_over NA         NA         NA         NA

Residual standard error: 37220 on 562 degrees of freedom
Multiple R-squared: 0.2784, Adjusted R-squared: 0.2643
F-statistic: 19.71 on 11 and 562 DF,  p-value: < 2.2e-16

linearMod6 <- lm(formula = current_base_pay ~ gender_Female + gender_Male +
                  race_grouping_white + race_grouping_person_of_color +
                  age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 +
                  age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
                  age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
                  age_group_5_65_over, data = news_salaried_regression)

Coefficients: (2 not defined because of singularities)
             Estimate Std. Error t value Pr(>|t|)
(Intercept)     177610     11711     15.166  < 2e-16 ***
gender_Female  -7131       3197     -2.230  0.026121 *
gender_Male NA         NA         NA         NA
## race_grouping_white -10244 7182 -1.426 0.154303
## race_grouping_person_of_color -21779 7634 -2.853 0.004492 **
## age_group_5_under_25 -90575 12366 -7.325 8.37e-13 ***
## age_group_5_25to29 -72999 10416 -7.008 6.94e-12 ***
## age_group_5_30to34 -57392 10302 -5.571 3.93e-08 ***
## age_group_5_35to39 -46156 10411 -4.433 1.12e-05 ***
## age_group_5_40to44 -28528 10677 -2.672 0.007759 **
## age_group_5_45to49 -37051 11136 -3.327 0.000935 ***
## age_group_5_50to54 -32670 10587 -3.086 0.002130 **
## age_group_5_55to59 -18756 10902 -1.720 0.085904 .
## age_group_5_60to64 -25603 11874 -2.156 0.031490 *
## age_group_5_65_over NA NA NA NA
## ---
## Signif. codes: 0 *** 0.001 ** 0.01 * 0.05 . 0.1 1
##
## Residual standard error: 37090 on 561 degrees of freedom
## Multiple R-squared: 0.2848, Adjusted R-squared: 0.2695
## F-statistic: 18.61 on 12 and 561 DF, p-value: < 2.2e-16

linearMod7 <- lm(formula = current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over + tier_Tier_1 + tier_Tier_2 + tier_Tier_3 + tier_Tier_4, data = news_salaried_regression)

summary(linearMod7)

## Call:
## lm(formula = current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over + tier_Tier_1 + tier_Tier_2 + tier_Tier_3 + tier_Tier_4, data = news_salaried_regression)
##
## Residuals:
## Min 1Q Median 3Q Max
## -73755 -19471 -4221 11237 181914
##
## Coefficients: (3 not defined because of singularities)
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 120967.5 11990.4 10.089 < 2e-16 ***
## gender_Female -4876.0 2760.7 -1.766 0.077907 .
## gender_Male NA NA NA NA
## race_grouping_white 9406.8 6384.3 1.473 0.141200
## race_grouping_person_of_color 651.1 6797.4 0.096 0.923721
## age_group_5_under_25 -70967.7 10751.7 -6.601 9.54e-11 ***
## age_group_5_25to29 -51967.8 9147.7 -5.681 2.16e-08 ***
## age_group_5_30to34 -45835.5 8943.2 -5.125 4.10e-07 ***
## age_group_5_35to39 -40948.5 8989.4 -4.555 6.43e-06 ***
## age_group_5_40to44 -25228.7 9217.5 -2.737 0.006397 **
## age_group_5_45to49 -28012.2 9613.5 -2.914 0.003713 **
## age_group_5_50to54 -22011.8 945.4 -2.407 0.016413 *
## age_group_5_55to59 -13805.7 9398.9 -1.469 0.142435
## age_group_5_60to64 -20565.1 10235.5 -2.009 0.044997 *
## age_group_5_65_over NA NA NA NA
## tier_Tier_1 53348.3 6214.7 8.584 < 2e-16 ***
## tier_Tier_2 23380.1 6049.1 3.865 0.000124 ***
## tier_Tier_3 2870.8 6037.4 0.475 0.634619
### tier_Tier_4

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>NA</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
</tbody>
</table>

---

**Signif. codes:*** 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1**

**Residual standard error: 31930 on 558 degrees of freedom**

**Multiple R-squared: 0.4729, Adjusted R-squared: 0.4587**

**F-statistic: 33.38 on 15 and 558 DF, p-value: < 2.2e-16**

```r
linearMod8 <- lm(formula = current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over + tier_Tier_1 + tier_Tier_2 + tier_Tier_3 + tier_Tier_4 + years_of_service_grouped_0 + years_of_service_grouped_1to2 + years_of_service_grouped_3to5 + years_of_service_grouped_6to10 + years_of_service_grouped_11to15 + years_of_service_grouped_16to20 + years_of_service_grouped_21to25 + years_of_service_grouped_25_over, data = news_salaried_regression)
```

```r
summary(linearMod8)
```

---

**Residuals:**

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-78346</td>
<td>-19056</td>
<td>3790</td>
<td>11052</td>
</tr>
</tbody>
</table>

---

**Coefficients: (4 not defined because of singularities)**

|                         | Estimate | Std. Error | t value | Pr(>|t|) |
|-------------------------|----------|------------|---------|---------|
| (Intercept)             | 125096   | 13133      | 9.525   | < 2e-16 *** |
| gender_Female           | -4750    | 2766       | -1.717  | 0.08653 . |
| race_grouping_white     | 9871     | 6401       | 1.542   | 0.12364 |
| race_grouping_person_of_color | 1117   | 6822       | 0.164   | 0.86996 |
| age_group_5_under_25    | -78083   | 11460      | -6.814  | 2.50e-11 *** |
| age_group_5_25to29      | -57223   | 9842       | -5.814  | 1.03e-08 *** |
| age_group_5_30to34      | -49792   | 9556       | -5.211  | 2.66e-07 *** |
| age_group_5_35to39      | -44320   | 9550       | -4.641  | 4.35e-06 *** |
| age_group_5_40to44      | -27729   | 9612       | -2.885  | 0.00407 **  |
| age_group_5_45to49      | -29546   | 9874       | -2.992  | 0.00289 **  |
| age_group_5_50to54      | -22921   | 9304       | -2.464  | 0.01406 *   |
| age_group_5_55to59      | -14698   | 9472       | -1.552  | 0.12129 |
| age_group_5_60to64      | -23419   | 10417      | -2.248  | 0.02495 *   |
| age_group_5_65_over     | NA       | NA         | NA      | NA       |
| tier_Tier_1             | 54494    | 6295       | 8.657   | < 2e-16 *** |
| tier_Tier_2             | 24832    | 6191       | 4.011   | 6.88e-05 *** |
| tier_Tier_3             | 3350     | 6125       | 0.547   | 0.58466 |
| tier_Tier_4             | NA       | NA         | NA      | NA       |
| years_of_service_grouped_0 | 1437    | 8374       | 0.172   | 0.86380 |
| years_of_service_grouped_1to2 | 2300   | 7767       | 0.296   | 0.76728 |
| years_of_service_grouped_3to5 | -2442  | 7680      | -0.319  | 0.75001 |
| years_of_service_grouped_6to10 | -7719 | 8036      | -0.961  | 0.33722 |
| years_of_service_grouped_11to15 | -6384 | 8152      | -0.783  | 0.43394 |
| years_of_service_grouped_16to20 | -6308 | 7463      | -0.845  | 0.39831 |

---

97
## years_of_service_grouped_21to25  -12596  8953  -1.407  0.16003
## years_of_service_grouped_25_over   NA   NA   NA   NA   NA
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 31910 on 551 degrees of freedom
## Multiple R-squared: 0.4802, Adjusted R-squared: 0.4594
## F-statistic: 23.14 on 22 and 551 DF, p-value: < 2.2e-16

merit_raises_combined_salaried_regression <- filter(merit_raises_combined, dept == 'News', pay_rate_type == 'Salaried')

merit_raises_combined_salaried_regression <- fastDummies::dummy_cols(merit_raises_combined_salaried_regression)

names(merit_raises_combined_salaried_regression) <- gsub('_', '', names(merit_raises_combined_salaried_regression))

names(merit_raises_combined_salaried_regression) <- gsub('\+', '_over', names(merit_raises_combined_salaried_regression))

names(merit_raises_combined_salaried_regression) <- gsub('<', 'under', names(merit_raises_combined_salaried_regression))


linearMod9 <- lm(formula = base_pay_change ~ gender_Female + gender_Male, data = merit_raises_combined_salaried_regression)

## Call:
## lm(formula = base_pay_change ~ gender_Female + gender_Male, data = merit_raises_combined_salaried_regression)
##
## Residuals:
##     Min   1Q Median   3Q   Max
## -2775.8 -1074.6 -275.8  724.2 16724.0
##
## Coefficients: (1 not defined because of singularities)
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3275.85   75.31   43.49  <2e-16 ***
## gender_Female -201.24  111.20  -1.81   0.0707 .
## gender_Male NA    NA    NA    NA    NA
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1578 on 809 degrees of freedom
## Multiple R-squared: 0.004032, Adjusted R-squared: -0.0028
## F-statistic: 3.275 on 1 and 809 DF, p-value: 0.07072

linearMod10 <- lm(formula = base_pay_change ~ race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_salaried_regression)

## Call:
## lm(formula = base_pay_change ~ race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_salaried_regression)
##
## Residuals:
##     Min   1Q Median   3Q   Max
## -2747.6 -997.6 -247.6 752.4 16752.3
##
## Coefficients:  Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3426.8   394.1  8.694  <2e-16 ***
## race_grouping_white  -179.2  399.2  -0.449  0.654
## race_grouping_person_of_color  -494.1  411.9  -1.200  0.231
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1577 on 808 degrees of freedom
## Multiple R-squared:  0.00714,  Adjusted R-squared:  0.004682
## F-statistic: 2.905 on 2 and 808 DF,  p-value: 0.05531

linearMod11 <- lm(formula = base_pay_change ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_salaried_regression)
summary(linearMod11)

##
## Call:
## lm(formula = base_pay_change ~ gender_Female + gender_Male +
##     race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_salaried_regression)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -2824.7 -1031.6  -324.7   675.3 16675.2
##
## Coefficients: (1 not defined because of singularities)
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)        3527.6     398.7   8.847  <2e-16 ***
## gender_Female      -179.4     111.6  -1.607  0.108
## gender_Male         NA         NA      NA      NA
## race_grouping_white -203.0     399.1  -0.509  0.611
## race_grouping_person_of_color -496.0     411.5  -1.205  0.228
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1575 on 807 degrees of freedom
## Multiple R-squared:  0.01031,  Adjusted R-squared:  0.006628
## F-statistic: 2.802 on 3 and 807 DF,  p-value: 0.03901

linearMod12 <- lm(formula = base_pay_change ~ gender_Female + gender_Male + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 +
                   age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
                   age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
                   age_group_5_65_over, data = merit_raises_combined_salaried_regression)
summary(linearMod12)

##
## Call:
## lm(formula = base_pay_change ~ gender_Female + gender_Male +
##    age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 +
##    age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
##    age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
##    age_group_5_65_over, data = merit_raises_combined_salaried_regression)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -2811.4 -919.2 -269.6  580.8 16528.8
##
## Coefficients: (2 not defined because of singularities)
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)        2650.4     254.7   10.408  < 2e-16 ***
## gender_Female      -225.4     110.8  -2.034   0.042296 *
## gender_Male         NA         NA      NA      NA

99
## age_group_5_under_25 -312.6 684.1 -0.457 0.647863
## age_group_5_25to29 661.0 315.0 2.098 0.036175 *
## age_group_5_30to34 820.6 285.2 2.877 0.004117 **
## age_group_5_35to39 994.2 289.1 3.439 0.000614 ***
## age_group_5_40to44 942.4 297.5 3.168 0.001596 **
## age_group_5_45to49 768.5 309.3 2.484 0.013191 *
## age_group_5_50to54 113.3 288.2 0.393 0.694409
## age_group_5_55to59 561.6 299.4 1.875 0.061104 .
## age_group_5_60to64 476.2 330.1 1.442 0.149582
## age_group_5_65_over NA NA NA NA
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error: 1552 on 800 degrees of freedom
## Multiple R-squared: 0.04691, Adjusted R-squared: 0.03499
## F-statistic: 3.937 on 10 and 800 DF, p-value: 2.953e-05

linearMod13 <- lm(formula = base_pay_change ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_salaried_regression)

## Call:
## lm(formula = base_pay_change ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_salaried_regression)
## Residuals:
## Min 1Q Median 3Q Max
## -2764.1 -940.3 -289.7 602.3 16548.0
## Coefficients: (1 not defined because of singularities)
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2607.1 469.0 5.559 3.69e-08 ***
## race_grouping_white -33.8 395.9 -0.085 0.931998
## race_grouping_person_of_color -425.7 407.7 -1.044 0.296638
## age_group_5_under_25 -423.3 680.6 -0.622 0.534129
## age_group_5_25to29 690.8 314.9 2.194 0.028528 *
## age_group_5_30to34 878.6 286.2 3.070 0.002210 **
## age_group_5_35to39 1066.6 290.4 3.673 0.000256 ***
## age_group_5_40to44 1053.7 301.1 3.500 0.000492 ***
## age_group_5_45to49 790.0 290.5 2.710 0.006885 *
## age_group_5_50to54 216.4 299.1 0.724 0.469046
## age_group_5_55to59 583.6 299.1 1.951 0.051391 .
## age_group_5_60to64 498.6 330.1 1.510 0.131317
## age_group_5_65_over NA NA NA NA
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error: 1549 on 799 degrees of freedom
## Multiple R-squared: 0.0519, Adjusted R-squared: 0.03885
## F-statistic: 3.976 on 11 and 799 DF, p-value: 1.165e-05

100
linearMod14 <- lm(formula = base_pay_change ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_salaried_regression)

summary(linearMod14)

## Call:
## lm(formula = base_pay_change ~ gender_Female + gender_Male +
##     race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 +
##     age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 +
##     age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 +
##     age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over,
##     data = merit_raises_combined_salaried_regression)

## Residuals:
##    Min     1Q Median     3Q    Max
## -2877.3 -956.5  -288.3  589.4 16449.3

## Coefficients: (2 not defined because of singularities)
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    2704.77    471.57   5.736  1.38e-08 ***
## gender_Female -196.82     111.10  -1.772   0.076853 .
## gender_Male NA NA NA NA
## race_grouping_white -64.13     395.78  -0.162  0.871313
## race_grouping_person_of_color -431.55     407.13  -1.060  0.289475
## age_group_5_under_25 -326.62     681.85  -0.479  0.632053
## age_group_5_25to29  736.69     315.51   2.335  0.019794 *
## age_group_5_30to34  909.99     286.33   3.178  0.001539 **
## age_group_5_35to39 1086.86     290.23   3.745  0.000194 ***
## age_group_5_40to44 1048.77     300.69   3.488  0.000513 ***
## age_group_5_45to49  808.51     308.75   2.619  0.008996 **
## age_group_5_50to64  221.43     290.13   0.763  0.445568
## age_group_5_55to59  594.80     298.76   1.991  0.046834 *
## age_group_5_60to64  512.72     329.73   1.555  0.120345
## age_group_5_65_over NA NA NA NA

## Signif. codes:  0 '***'  0.001 '**'  0.01 '*'  0.05 '.'  0.1 ' ' 1

## Residual standard error: 1547 on 798 degrees of freedom
## Multiple R-squared:  0.05562,  Adjusted R-squared:  0.04142
## F-statistic:  3.916 on 12 and 798 DF,  p-value: 7.08e-06

linearMod15 <- lm(formula = performance_rating ~ gender_Female + gender_Male, data = merit_raises_combined_salaried_regression)

summary(linearMod15)

## Call:
## lm(formula = performance_rating ~ gender_Female + gender_Male,
##     data = merit_raises_combined_salaried_regression)

## Residuals:
##    Min     1Q Median     3Q    Max
## -0.8339 -0.2063  0.0339  0.1937  1.0661

## Coefficients: (1 not defined because of singularities)
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    101

101
(Intercept)  3.60631  0.01616  223.118  < 2e-16 ***  
gender_Female -0.07241  0.02383  -3.038  0.00246 **  
gender_Male NA NA NA NA

---

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3281 on 761 degrees of freedom  
(48 observations deleted due to missingness)  
Multiple R-squared:  0.01199,  Adjusted R-squared:  0.01069  
F-statistic: 9.232 on 1 and 761 DF,  p-value: 0.00246

linearMod16 <- lm(formula = performance_rating ~ race_grouping_white + race_grouping_person_of_color,   
data = merit_raises_combined_salaried_regression)

Call:  
lm(formula = performance_rating ~ race_grouping_white + race_grouping_person_of_color,   
data = merit_raises_combined_salaried_regression)

Residuals:

Min 1Q Median 3Q Max
-0.90017 -0.20017 -0.00017 0.19983 0.99983

Coefficients:

(Intercept) 3.72500  0.08115  45.900  < 2e-16 ***
race_grouping_white -0.12483  0.08226  -1.517  0.12957
race_grouping_person_of_color -0.26258  0.08500  -3.089  0.00208 **

---

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3246 on 760 degrees of freedom  
(48 observations deleted due to missingness)  
Multiple R-squared:  0.03398,  Adjusted R-squared:  0.03143  
F-statistic: 13.37 on 2 and 760 DF,  p-value: 1.974e-06

linearMod17 <- lm(formula = performance_rating ~ gender_Female + gender_Male + 
race_grouping_white + race_grouping_person_of_color,   
data = merit_raises_combined_salaried_regression)

Call:  
lm(formula = performance_rating ~ gender_Female + gender_Male + 
race_grouping_white + race_grouping_person_of_color,   
data = merit_raises_combined_salaried_regression)

Residuals:

Min 1Q Median 3Q Max
-0.86450 -0.22704 -0.02704 0.17296 1.03550

Coefficients: (1 not defined because of singularities)

(Intercept)  3.76018  0.08192  45.900  < 2e-16 ***
gender_Female -0.06254  0.02363  -2.647  0.00830 **
gender_Male NA NA NA NA
race_grouping_white -0.13314  0.08200  -1.624  0.10486
race_grouping_person_of_color -0.26288  0.08466  -3.105  0.00197 **
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Residual standard error: 0.3233 on 759 degrees of freedom

## Multiple R-squared: 0.04281, Adjusted R-squared: 0.03903

## F-statistic: 11.32 on 3 and 759 DF,  p-value: 2.878e-07

linearMod18 <- lm(formula = performance_rating ~ gender_Female + gender_Male + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_salaried_regression)

summary(linearMod18)

## Call:
## lm(formula = performance_rating ~ gender_Female + gender_Male + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_salaried_regression)

## Residuals:
## Min 1Q Median 3Q Max
## -0.79665 -0.22275 -0.04557 0.20335 1.04477

## Coefficients: (2 not defined because of singularities)
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)        3.71995   0.05474  67.951  < 2e-16 ***
## gender_Female     -0.05524   0.02383  -2.318  0.020734 *
## gender_Male         NA         NA         NA        NA
## age_group_5_under_25 -0.42391   0.14348  -2.954  0.003230 **
## age_group_5_25to29  -0.23365   0.06766  -3.453  0.000585 ***
## age_group_5_30to34  -0.16806   0.06119  -2.746  0.006170 **
## age_group_5_35to39  -0.11914   0.06202  -1.921  0.055133 .
## age_group_5_40to44  -0.07345   0.06377  -1.152  0.249801
## age_group_5_45to49  -0.14779   0.06609  -2.236  0.025631 *
## age_group_5_50to54  -0.10948   0.06225  -1.759  0.079048 .
## age_group_5_55to59  -0.05711   0.06463  -0.884  0.377127
## age_group_5_60to64  -0.09720   0.07062  -1.376  0.169122
## age_group_5_65_over    NA       NA       NA        NA

## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Residual standard error: 0.3244 on 752 degrees of freedom

## Multiple R-squared: 0.04554, Adjusted R-squared: 0.03284

## F-statistic: 3.588 on 10 and 752 DF,  p-value: 0.0001139

linearMod19 <- lm(formula = performance_rating ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_salaried_regression)

summary(linearMod19)

## Call:
## lm(formula = performance_rating ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_salaried_regression)

## Residuals:
## Min 1Q Median 3Q Max
## -0.9677 -0.22275 -0.04557 0.20335 1.04477

## Coefficients: (1 not defined because of singularities)
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)        3.71995   0.05474  67.951  < 2e-16 ***
## race_grouping_white NA         NA         NA        NA
## race_grouping_person_of_color     NA         NA       NA        NA

## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Residual standard error: 0.3244 on 752 degrees of freedom

## Multiple R-squared: 0.04554, Adjusted R-squared: 0.03284

## F-statistic: 3.588 on 10 and 752 DF,  p-value: 0.0001139

linearMod19 <- lm(formula = performance_rating ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_salaried_regression)

summary(linearMod19)

## Call:
## lm(formula = performance_rating ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_salaried_regression)
## Residuals:
<table>
<thead>
<tr>
<th></th>
<th>Min</th>
<th>1Q</th>
<th>Median</th>
<th>3Q</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>-0.85831</td>
<td>-0.21122</td>
<td>-0.02461</td>
<td>0.20166</td>
<td>0.99313</td>
</tr>
</tbody>
</table>

## Coefficients: (1 not defined because of singularities)

|                      | Estimate  | Std. Error | t value | Pr(>|t|) |
|----------------------|-----------|------------|---------|----------|
| (Intercept)          | 3.81831   | 0.09786    | 39.020  | < 2e-16  *** |
| race_grouping_white  | -0.11831  | 0.08200    | -1.443  | 0.14947  |
| race_grouping_person_of_color | -0.25310 | 0.08450    | -2.995  | 0.00283  ** |
| age_group_5_under_25 | -0.45000  | 0.14130    | -3.185  | 0.00151  ** |
| age_group_5_25to29   | -0.21815  | 0.06703    | -3.255  | 0.00119  ** |
| age_group_5_30to34   | -0.14169  | 0.06088    | -2.327  | 0.02022  * |
| age_group_5_35to39   | -0.09313  | 0.06165    | -1.511  | 0.13129  |
| age_group_5_40to44   | -0.04061  | 0.06390    | -0.635  | 0.52532  |
| age_group_5_45to49   | -0.13706  | 0.06537    | -2.097  | 0.03637  * |
| age_group_5_50to54   | -0.07541  | 0.06208    | -1.215  | 0.22483  |
| age_group_5_55to59   | -0.04649  | 0.06392    | -0.727  | 0.46726  |
| age_group_5_60to64   | -0.08878  | 0.06993    | -1.270  | 0.20460  |
| age_group_5_65_over  | NA        | NA         | NA      | NA       |

---

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3204 on 751 degrees of freedom
(48 observations deleted due to missingness)
Multiple R-squared:  0.06981,  Adjusted R-squared:  0.05618
F-statistic: 5.124 on 11 and 751 DF,  p-value: 8.908e-08

linearMod20 <- lm(formula = performance_rating ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_salaried_regression)

Summary: (2 not defined because of singularities)

|                      | Estimate  | Std. Error | t value | Pr(>|t|) |
|----------------------|-----------|------------|---------|----------|
| (Intercept)          | 3.84222   | 0.09844    | 39.031  | < 2e-16  *** |
| gender_Female        | -0.04611  | 0.02364    | -1.950  | 0.05149  |
| gender_Male          | NA        | NA         | NA      | NA       |
| race_grouping_white  | -0.12557  | 0.08193    | -1.533  | 0.12578  |
| race_grouping_person_of_color | -0.25452 | 0.08435    | -3.017  | 0.00264  ** |
| age_group_5_under_25 | -0.42822  | 0.06970    | -6.148  | 0.000256  ** |
| age_group_5_25to29   | -0.20809  | 0.06624    | -3.150  | 0.00200  ** |
| age_group_5_30to34   | -0.13661  | 0.06082    | -2.246  | 0.02499  * |
## age_group_5_35to39  -0.08886  0.06157  -1.443  0.14937
## age_group_5_40to44  -0.04247  0.06379  -0.666  0.50578
## age_group_5_45to49  -0.13432  0.06527  -2.058  0.03994 *
## age_group_5_50to54  -0.07309  0.06197  -1.179  0.23861
## age_group_5_55to59  -0.04519  0.06381  -0.708  0.47902
## age_group_5_60to64  -0.08742  0.06980  -1.252  0.21081
## age_group_5_65_over NA NA NA NA

## Signif. codes:  0 ***  0.001 **  0.01 *  0.05 .  0.1  1
##
## Residual standard error: 0.3198 on 750 degrees of freedom
## (48 observations deleted due to missingness)
## Multiple R-squared: 0.0745, Adjusted R-squared: 0.0597
## F-statistic: 5.031 on 12 and 750 DF, p-value: 4.315e-08

```r
news_hourly_regression <-
```

```r
names(news_hourly_regression) <-
  gsub('_', '-', names(news_hourly_regression))
names(news_hourly_regression) <-
  gsub('+', '_over', names(news_hourly_regression))
names(news_hourly_regression) <-
  gsub('<', 'under_', names(news_hourly_regression))
```

```r
linearMod21 <- lm(formula = current_base_pay ~ gender_Female + gender_Male, data = news_hourly_regression)
summary(linearMod21)
```

```r
linearMod22 <- lm(formula = current_base_pay ~ race_grouping_white + race_grouping_person_of_color, data = news_hourly_regression)
summary(linearMod22)
```
## Min 1Q Median 3Q Max
## -20.799 -6.485 -0.920 5.844 33.511

## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 39.230 8.131 4.825 5.47e-06 ***
## race_grouping_white -3.681 8.257 -0.446 0.657
## race_grouping_person_of_color -9.099 8.397 -1.084 0.281
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Residual standard error: 11.5 on 93 degrees of freedom
## Multiple R-squared: 0.05071, Adjusted R-squared: 0.0303
## F-statistic: 2.484 on 2 and 93 DF, p-value: 0.08892

linearMod23 <- lm(formula = current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color, data = news_hourly_regression)

## Call:
## lm(formula = current_base_pay ~ gender_Female + gender_Male +
## race_grouping_white + race_grouping_person_of_color, data = news_hourly_regression)
## Residuals:
## Min 1Q Median 3Q Max
## -21.839 -5.883 -0.707 5.165 32.471

## Coefficients: (1 not defined because of singularities)
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 36.336 8.486 4.282 4.54e-05 ***
## gender_Female 2.894 2.481 1.166 0.246
## gender_Male NA NA NA NA
## race_grouping_white -2.641 8.289 -0.319 0.751
## race_grouping_person_of_color -8.134 8.422 -0.966 0.337
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Residual standard error: 11.48 on 92 degrees of freedom
## Multiple R-squared: 0.06455, Adjusted R-squared: 0.03404
## F-statistic: 2.116 on 3 and 92 DF, p-value: 0.1036

linearMod24 <- lm(formula = current_base_pay ~ gender_Female + gender_Male + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 +
## age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
## age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
## age_group_5_65_over, data = news_hourly_regression)

## Call:
## lm(formula = current_base_pay ~ gender_Female + gender_Male +
## age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 +
## age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
## age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
## age_group_5_65_over, data = news_hourly_regression)
## Residuals:
## Min 1Q Median 3Q Max
## -21.9861 -6.8240 0.1867 6.5690 21.9307
## Coefficients: (2 not defined because of singularities)

| Estimate | Std. Error | t value | Pr(>|t|) |
|----------|------------|---------|----------|
| (Intercept) | 43.2529 | 4.9315 | 8.771 | 1.57e-13 *** |
| gender_Female | 3.8764 | 2.3643 | 1.640 | 0.104789 |
| gender_Male | NA | NA | NA |
| age_group_5_under_25 | -20.4162 | 5.3074 | -3.847 | 0.000231 *** |
| age_group_5_25to29 | -17.3859 | 5.0272 | -3.458 | 0.000851 *** |
| age_group_5_30to34 | -12.0079 | 5.5888 | -2.149 | 0.034514 * |
| age_group_5_35to39 | -13.9194 | 5.8204 | -2.391 | 0.018988 * |
| age_group_5_40to44 | -7.8709 | 5.7096 | -1.379 | 0.171659 |
| age_group_5_45to49 | -0.8132 | 6.1672 | -0.132 | 0.895405 |
| age_group_5_50to54 | -6.2127 | 5.8069 | -1.070 | 0.287704 |
| age_group_5_55to59 | -9.4947 | 6.0858 | -1.560 | 0.122442 |
| age_group_5_60to64 | -3.5887 | 6.4582 | -0.556 | 0.579887 |
| age_group_5_65_over | NA | NA | NA |

---

## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Residual standard error: 10.18 on 85 degrees of freedom
## Multiple R-squared: 0.3194, Adjusted R-squared: 0.2394
## F-statistic: 3.99 on 10 and 85 DF, p-value: 0.000172

```r
linearMod25 <- lm(formula = current_base_pay ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = news_hourly_regression)
summary(linearMod25)
```

## Call:
```
lm(formula = current_base_pay ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = news_hourly_regression)
```

## Residuals:
```
Min 1Q Median 3Q Max
-21.7944 -8.2916 0.2457 6.6045 23.3800
```

## Coefficients: (1 not defined because of singularities)

| Estimate | Std. Error | t value | Pr(>|t|) |
|----------|------------|---------|----------|
| (Intercept) | 57.365 | 8.810 | 6.511 | 5.17e-09 *** |
| race_grouping_white | -9.920 | 7.561 | -1.312 | 0.193086 |
| race_grouping_person_of_color | -12.646 | 7.728 | -1.636 | 0.105485 |
| age_group_5_under_25 | -20.825 | 5.346 | -3.895 | 0.000196 *** |
| age_group_5_25to29 | -17.290 | 5.045 | -3.427 | 0.000948 *** |
| age_group_5_30to34 | -15.444 | 5.590 | -2.763 | 0.007043 ** |
| age_group_5_35to39 | -14.666 | 5.820 | -2.520 | 0.013624 * |
| age_group_5_40to44 | -9.303 | 5.710 | -1.629 | 0.106989 |
| age_group_5_45to49 | -1.320 | 6.207 | -0.213 | 0.832088 |
| age_group_5_50to54 | -6.815 | 5.831 | -1.169 | 0.245745 |
| age_group_5_55to59 | -12.189 | 6.009 | -2.028 | 0.045688 * |
| age_group_5_60to64 | -5.454 | 6.528 | -0.836 | 0.405793 |
| age_group_5_65_over | NA | NA | NA |

---

## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

107
## Residual standard error: 10.21 on 84 degrees of freedom  
## Multiple R-squared: 0.3243, Adjusted R-squared: 0.2358  
## F-statistic: 3.665 on 11 and 84 DF, p-value: 0.0002868

```r
linearMod26 <- lm(formula = current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = news_hourly_regression)
```

```r
summary(linearMod26)
```

```r
## Call:  
## lm(formula = current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = news_hourly_regression)  
## Residuals:  
##     Min      1Q  Median      3Q     Max  
## -22.3029 -7.4082 -0.4587  6.2195 22.0580  
## Coefficients: (2 not defined because of singularities)  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 53.290 9.243 5.766 1.35e-07 ***  
## gender_Female 3.305 2.385 1.386 0.169551  
## gender_Male NA NA NA NA  
## race_grouping_white -8.590 7.581 -1.133 0.260405  
## race_grouping_person_of_color -11.063 7.770 -1.424 0.158260  
## age_group_5_under_25 -20.927 5.318 -3.935 0.000172 ***  
## age_group_5_25to29 -17.120 5.020 -3.411 0.001003 **  
## age_group_5_30to34 -13.803 5.685 -2.428 0.017343 *  
## age_group_5_35to39 -14.081 5.803 -2.426 0.017414 *  
## age_group_5_40to44 -8.450 5.712 -1.479 0.142828  
## age_group_5_45to49 -1.371 6.173 -0.222 0.824773  
## age_group_5_50to54 -6.612 5.801 -1.140 0.257613  
## age_group_5_55to59 -10.424 6.111 -1.706 0.091769 .  
## age_group_5_60to64 -4.692 6.516 -0.720 0.473490  
## age_group_5_65_over NA NA NA NA  
## --  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  

## Residual standard error: 10.15 on 83 degrees of freedom  
## Multiple R-squared: 0.3396, Adjusted R-squared: 0.2441  
## F-statistic: 3.557 on 12 and 83 DF, p-value: 0.0002782
```

```r
linearMod27 <- lm(formula = current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over + tier_Tier_1 + tier_Tier_2 + tier_Tier_3 + tier_Tier_4, data = news_hourly_regression)
```

```r
summary(linearMod27)
```

```r
## Call:  
## lm(formula = current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over + tier_Tier_1 + tier_Tier_2 + tier_Tier_3 + tier_Tier_4, data = news_hourly_regression)  
## Residual standard error: 10.15 on 83 degrees of freedom  
## Multiple R-squared: 0.3396, Adjusted R-squared: 0.2441  
## F-statistic: 3.557 on 12 and 83 DF, p-value: 0.0002782
```
## Residuals:
##     Min      1Q  Median       3Q      Max
## -24.2238 -6.6641 -0.0323  5.5695  21.6317
##
## Coefficients: (3 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 53.6848 14.0087 3.832 0.000252 ***
## gender_Female 3.3510 2.3687 1.415 0.161044
## gender_Male NA NA NA NA
## race_grouping_white -6.0059 7.6086 -0.789 0.432234
## race_grouping_person_of_color -8.6620 7.7760 -1.114 0.268644
## age_group_5_under_25 -21.7546 5.2967 -4.107 9.61e-05 ***
## age_group_5_25to29 -19.2538 5.0763 -3.793 0.000288 ***
## age_group_5_30to34 -14.5598 5.6815 -2.563 0.012263 *
## age_group_5_35to39 -14.7994 5.7933 -2.563 0.012528 *
## age_group_5_40to44 -9.3640 6.1190 -1.518 0.133215
## age_group_5_45to49 -6.7158 5.7787 -1.162 0.248626
## age_group_5_50to54 -6.6075 6.5166 -1.014 0.313667
## age_group_5_55to59 -6.6075 6.5166 -1.014 0.313667
## age_group_5_60to64 -6.0755 6.5166 -1.014 0.313667
## age_group_5_65_over NA NA NA NA
## tier_Tier_1 -0.6841 10.6737 -0.064 0.949058
## tier_Tier_2 -4.7945 10.5300 -0.455 0.650114
## tier_Tier_3 0.3514 10.4667 0.034 0.973149
## tier_Tier_4 NA NA NA NA
##
## Residual standard error: 10.04 on 80 degrees of freedom
## Multiple R-squared: 0.3771, Adjusted R-squared: 0.2603
## F-statistic: 3.229 on 15 and 80 DF,  p-value: 0.0003464

linearMod28 <- lm(formula = current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over + tier_Tier_1 + tier_Tier_2 + tier_Tier_3 + tier_Tier_4 + years_of_service_grouped_0 + years_of_service_grouped_1to2 + years_of_service_grouped_3to5 + years_of_service_grouped_6to10 + years_of_service_grouped_11to15 + years_of_service_grouped_16to20 + years_of_service_grouped_21to25 + years_of_service_grouped_25_over, data = news_hourly_regression)

## Residuals:
##     Min      1Q  Median      3Q      Max
## -25.3075 -6.0851 -0.2956  4.8862  21.3493
##
## Coefficients: (4 not defined because of singularities)
## Estimate Std. Error t value Pr(>|t|)
(Intercept)  53.931   16.201   3.329  0.001369 **
gender_Female  3.898    2.602   1.498  0.138366
gender_Male NA NA NA NA
race_grouping_white -6.516    8.151  -0.799  0.426690
race_grouping_person_of_color -8.812    8.264  -1.066  0.289779
age_group_5_under_25 -26.059    7.469  -3.489  0.000825 ***
age_group_5_25to29 -23.476    7.134  -3.291  0.001541 **
age_group_5_30to34 -18.204    6.875  -2.648  0.009919 **
age_group_5_35to39 -16.802    6.260  -2.684  0.008991 **
age_group_5_40to44 -10.518    6.094  -1.726  0.088617 .
age_group_5_45to49 -3.107    6.913  -0.449  0.654408
age_group_5_50to54 -6.589    6.271  -1.051  0.296836
age_group_5_55to59 -10.051    6.443  -1.560  0.123107
age_group_5_60to64 -4.929    7.126  -0.692  0.491289
age_group_5_65_over NA NA NA NA
tier_Tier_1 -2.363   11.910  -0.198  0.843253
tier_Tier_2 -6.444   11.796  -0.546  0.586533
tier_Tier_3 -1.568   11.796  -0.133  0.894608
tier_Tier_4 NA NA NA NA
years_of_service_grouped_0  4.690    8.300   0.565  0.573758
years_of_service_grouped_1to2  6.449    7.941   0.812  0.419322
years_of_service_grouped_3to5  5.510    8.094   0.681  0.498186
years_of_service_grouped_6to10  3.580    6.740   0.531  0.596937
years_of_service_grouped_11to15  2.551    7.223   0.351  0.724986
years_of_service_grouped_16to20 -1.724    6.531  -0.264  0.792586
years_of_service_grouped_21to25  3.795    8.071   0.470  0.639637
years_of_service_grouped_25_over NA NA NA NA
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 10.33 on 73 degrees of freedom
Multiple R-squared: 0.3984, Adjusted R-squared: 0.2171
F-statistic: 2.197 on 22 and 73 DF,  p-value: 0.006537

merit_raises_combined_hourly_regression <- 
filter(merit_raises_combined, dept == 'News', pay_rate_type == 'Hourly')

merit_raises_combined_hourly_regression <- 
fastDummies::dummy_cols(merit_raises_combined_hourly_regression, select_columns = c(gender, race_grouping, age_group_5), names(merit_raises_combined_hourly_regression) <- gsub('_', '-', names(merit_raises_combined_hourly_regression))
names(merit_raises_combined_hourly_regression) <- gsub('<', 'under_', names(merit_raises_combined_hourly_regression))
linearMod29 <- lm(formula = base_pay_change ~ gender_Female + gender_Male, data=merit_raises_combined_hourly_regression)

summary(linearMod29)

# Call:
# lm(formula = base_pay_change ~ gender_Female + gender_Male, data = merit_raises_combined_hourly_regression)
# # Residuals:
# Min 1Q Median 3Q Max
# -1.4296 -0.8772 -0.3572 0.1916 11.2704
# # Coefficients: (1 not defined because of singularities)
```
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.3872  0.2213  6.270 6.27e-09 ***
## gender_Female 0.3023  0.2845  1.063 0.29
## gender_Male   NA      NA      NA      NA
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.517 on 117 degrees of freedom
## Multiple R-squared: 0.009564, Adjusted R-squared: 0.001098
## F-statistic: 1.13 on 1 and 117 DF, p-value: 0.29
```

```
linearMod30 <- lm(formula = base_pay_change ~ race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_hourly_regression)
```

```
##
## Call:
## lm(formula = base_pay_change ~ race_grouping_white + race_grouping_person_of_color, 
##     data = merit_raises_combined_hourly_regression)
##
## Residuals:
##    Min     1Q  Median     3Q    Max
## -1.4851 -0.7451 -0.2051  0.2399 11.2149
##
## Coefficients: (1 not defined because of singularities)
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.1824  0.2468  4.790 4.92e-06 ***
## race_grouping_white 0.5627  0.2973  1.892 0.0609 .
## race_grouping_person_of_color NA   NA      NA      NA
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.501 on 117 degrees of freedom
## Multiple R-squared: 0.0297, Adjusted R-squared: 0.0214
## F-statistic: 3.581 on 1 and 117 DF, p-value: 0.06092
```

```
linearMod31 <- lm(formula = base_pay_change ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_hourly_regression)
```

```
##
## Call:
## lm(formula = base_pay_change ~ gender_Female + gender_Male + 
##     race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_hourly_regression)
##
## Residuals:
##    Min     1Q  Median     3Q    Max
## -1.5607 -0.7907 -0.2761  0.2293 11.1393
##
## Coefficients: (2 not defined because of singularities)
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.0747  0.2837  3.788 0.000242 ***
## gender_Female 0.2214  0.2859  0.775 0.440140
## gender_Male   NA      NA      NA      NA
## race_grouping_white 0.5246  0.3019  1.738 0.084920 .
## race_grouping_person_of_color NA   NA      NA      NA
## ---
```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error: 1.504 on 116 degrees of freedom
## Multiple R-squared:  0.03469,  Adjusted R-squared:  0.01805
## F-statistic:  2.084 on 2 and 116 DF,  p-value: 0.129

```
linearMod32 <- lm(formula = base_pay_change ~ gender_Female + gender_Male + age_group_5_under_25 + age_group_5_25to29 + age_group_5_25to34 + age_group_5_35to39 + age_group_5_30to34 + age_group_5_40to44 + age_group_5_40to49 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_hourly_regression)
```

```
## Call:
## lm(formula = base_pay_change ~ gender_Female + gender_Male + age_group_5_under_25 + age_group_5_25to29 + age_group_5_25to34 + age_group_5_35to39 + age_group_5_30to34 + age_group_5_40to44 + age_group_5_40to49 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_hourly_regression)
## Residuals:
## Min 1Q Median 3Q Max
## -2.1687 -0.7028 -0.2051 0.1201 11.2348
## Coefficients: (2 not defined because of singularities)
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.37422 0.72535 1.895 0.0608 .
## gender_Female 0.25473 0.31068 0.820 0.4141
## gender_Male NA NA NA NA
## age_group_5_under_25 -0.94913 1.12219 -0.846 0.3995
## age_group_5_25to29 0.09621 0.74907 0.128 0.8980
## age_group_5_25to34 0.08219 0.78051 0.105 0.9163
## age_group_5_35to39 -0.16741 0.81641 -0.205 0.8379
## age_group_5_40to44 0.20675 0.78051 0.253 0.8006
## age_group_5_45to49 0.13063 0.77953 0.166 0.8686
## age_group_5_50to54 -0.24684 0.79405 -0.311 0.7565
## age_group_5_55to59 -0.20909 0.82883 -0.252 0.8013
## age_group_5_60to64 1.81974 1.02228 1.780 0.0779 .
## age_group_5_65_over NA NA NA NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error: 1.524 on 108 degrees of freedom
## Multiple R-squared:  0.0775,  Adjusted R-squared: -0.007921
## F-statistic:  0.9073 on 10 and 108 DF,  p-value: 0.5293

linearMod33 <- lm(formula = base_pay_change ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_25to34 + age_group_5_35to39 + age_group_5_30to34 + age_group_5_40to44 + age_group_5_40to49 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_hourly_regression)
```

```
## Call:
## lm(formula = base_pay_change ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_25to34 + age_group_5_35to39 + age_group_5_30to34 + age_group_5_40to44 + age_group_5_40to49 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_hourly_regression)
## Residuals:
## Min 1Q Median 3Q Max
## -2.1050 -0.7879 -0.2464 0.3048 10.9892
##
## Coefficients: (2 not defined because of singularities)
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.33904 0.68508 1.955 0.0532 .
## race_grouping_white 0.59741 0.32158 1.858 0.0659 .
## race_grouping_person_of_color NA NA NA NA
## age_group_5_under_25 -1.42644 1.11564 -1.279 0.2038
## age_group_5_25to29 0.03434 0.74037 0.046 0.9631
## age_group_5_30to34 -0.33672 0.77282 -0.436 0.6639
## age_group_5_35to39 -0.30357 0.80155 -0.379 0.7056
## age_group_5_40to44 -0.12854 0.81293 -0.158 0.8747
## age_group_5_45to49 -0.06264 0.78171 -0.080 0.9363
## age_group_5_50to54 -0.46271 0.79365 -0.583 0.5611
## age_group_5_55to59 -0.43830 0.81509 -0.538 0.5919
## age_group_5_60to64 1.44856 1.02761 1.410 0.1615
## age_group_5_65_over NA NA NA NA
##
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.505 on 108 degrees of freedom
## Multiple R-squared: 0.1005, Adjusted R-squared: 0.01721
## F-statistic: 1.207 on 10 and 108 DF, p-value: 0.2949

linearMod34 <- lm(formula = base_pay_change ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_hourly_regression)

## Call:
## lm(formula = base_pay_change ~ gender_Female + gender_Male +
## race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 +
## age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 +
## age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 +
## age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over,
## data = merit_raises_combined_hourly_regression)
##
## Residuals:
## ## Min 1Q Median 3Q Max
## ## -2.1318 -0.7714 -0.2091 0.2843 10.9822
##
## Coefficients: (3 not defined because of singularities)
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.26541 0.72209 1.752 0.0826 .
## gender_Female 0.10738 0.32017 0.335 0.7380
## gender_Male NA NA NA NA
## race_grouping_white 0.56673 0.33561 1.689 0.0942 .
## race_grouping_person_of_color NA NA NA NA
## age_group_5_under_25 -1.35793 1.13873 -1.192 0.2357
## age_group_5_25to29 0.03434 0.74037 0.046 0.9631
## age_group_5_30to34 -0.33672 0.77282 -0.436 0.6639
## age_group_5_35to39 -0.30357 0.80155 -0.379 0.7056
## age_group_5_40to44 -0.12854 0.81293 -0.158 0.8747
## age_group_5_45to49 -0.06264 0.78171 -0.080 0.9363
## age_group_5_50to54 -0.46271 0.79365 -0.583 0.5611
## age_group_5_55to59 -0.43830 0.81509 -0.538 0.5919
## age_group_5_60to64 1.44856 1.02761 1.410 0.1615
## age_group_5_65_over NA NA NA NA
##
## 113
## age_group_5_55to59
-0.39395 0.82907 -0.475 0.6356
## age_group_5_60to64
1.47233 1.03429 1.424 0.1575
## age_group_5_65_over
NA NA NA NA
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.511 on 107 degrees of freedom
Multiple R-squared: 0.1014, Adjusted R-squared: 0.009067
F-statistic: 1.098 on 11 and 107 DF, p-value: 0.3699

linearMod35 <- lm(formula = performance_rating ~ gender_Female + gender_Male, data=merit_raises_combined_hourly_regression)
summary(linearMod35)

##
## Call:
## lm(formula = performance_rating ~ gender_Female + gender_Male, data = merit_raises_combined_hourly_regression)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -0.59767 -0.26943 -0.09767  0.25882  0.90233
##
## Coefficients: (1 not defined because of singularities)
##                          Estimate Std. Error    t value  Pr(>|t|)
## (Intercept)                3.49767    0.05346   65.428  < 2e-16 ***
## gender_Female              0.04350    0.06830    0.637    0.526
## gender_Male                NA         NA         NA         NA
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3505 on 109 degrees of freedom
## (8 observations deleted due to missingness)
## Multiple R-squared: 0.003708, Adjusted R-squared: -0.005432
## F-statistic: 0.4057 on 1 and 109 DF, p-value: 0.5255

linearMod36 <- lm(formula = performance_rating ~ race_grouping_white + race_grouping_person_of_color, data=merit_raises_combined_hourly_regression)
summary(linearMod36)

##
## Call:
## lm(formula = performance_rating ~ race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_hourly_regression)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -0.67467 -0.27467 -0.01944  0.22533  0.82533
##
## Coefficients: (1 not defined because of singularities)
##                     Estimate Std. Error    t value  Pr(>|t|)
## (Intercept)            3.41944    0.05724   59.735  < 2e-16 ***
## race_grouping_white    0.15522    0.06964    2.229    0.0279 *
## race_grouping_person_of_color NA         NA         NA         NA
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## 114
## Residual standard error: 0.3435 on 109 degrees of freedom
## (8 observations deleted due to missingness)
## Multiple R-squared: 0.04359, Adjusted R-squared: 0.03482
## F-statistic: 4.968 on 1 and 109 DF,  p-value: 0.02787

linearMod37 <- lm(formula = performance_rating ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_hourly_regression)

##
## Call:
## lm(formula = performance_rating ~ gender_Female + gender_Male +
## race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_hourly_regression)
##
## Residuals:
##     Min     1Q   Median     3Q    Max
## -0.66495 -0.27924 -0.02699  0.23505  0.83505
##
## Coefficients: (2 not defined because of singularities)
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)        3.41270   0.06599  51.713  <2e-16 ***
## gender_Female      0.01429   0.06860   0.208  0.8354
## gender_Male         NA        NA        NA        NA
## race_grouping_white 0.15225   0.07138   2.133  0.0352 *
## race_grouping_person_of_color NA        NA        NA        NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.345 on 108 degrees of freedom
## (8 observations deleted due to missingness)
## Multiple R-squared: 0.04398,   Adjusted R-squared: 0.02627
## F-statistic: 2.484 on 2 and 108 DF,  p-value: 0.08817

linearMod38 <- lm(formula = performance_rating ~ gender_Female + gender_Male + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_hourly_regression)

##
## Call:
## lm(formula = performance_rating ~ gender_Female + gender_Male +
## age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 +
## age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
## age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
## age_group_5_65_over, data = merit_raises_combined_hourly_regression)
##
## Residuals:
##     Min     1Q   Median     3Q    Max
## -0.77659 -0.26798 -0.07659  0.22153  0.74303
##
## Coefficients: (2 not defined because of singularities)
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)        3.52255   0.16374  21.512  <2e-16 ***
## gender_Female      0.04682   0.07343   0.638  0.525
## gender_Male         NA        NA        NA        NA
## age_group_5_under_25 -0.24596   0.28679  -0.858  0.393
## age_group_5_25to29  -0.08878   0.16801  -0.528  0.598
## age_group_5_30to34  -0.04220   0.17780  -0.237  0.813
## age_group_5_35to39  -0.21655  0.18607  -1.164  0.247
## age_group_5_40to44  0.25404  0.18325   1.386  0.169
## age_group_5_45to49  -0.01376  0.17676  -0.078  0.938
## age_group_5_50to54  0.01885  0.18008   0.105  0.917
## age_group_5_55to59  0.03442  0.19241   0.179  0.858
## age_group_5_60to64  -0.22042  0.24978  -0.882  0.380
## age_group_5_65_over NA NA NA NA
### ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
###
## Residual standard error: 0.3418 on 100 degrees of freedom
### (8 observations deleted due to missingness)
## Multiple R-squared: 0.1312, Adjusted R-squared: 0.04432
## F-statistic: 1.51 on 10 and 100 DF, p-value: 0.1468
linearMod39 <- lm(formula = performance_rating ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_hourly_regression)

summary(linearMod39)

##
## Call:
## lm(formula = performance_rating ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_hourly_regression)
##
## Residuals:
##          Min        1Q     Median        3Q        Max
## -0.68218 -0.24318  -0.04153   0.19243   0.69655
##
## Coefficients: (2 not defined because of singularities)
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)      3.50345    0.15323 22.865  <2e-16 ***
## race_grouping_white  0.14139    0.07338  1.927    0.057 .
## race_grouping_person_of_color NA NA NA NA
## age_group_5_under_25 -0.34483    0.28477 -1.211    0.229
## age_group_5_25to29  -0.10331    0.16547 -0.624    0.534
## age_group_5_30to34  -0.13408    0.17470 -0.767    0.444
## age_group_5_35to39  -0.24044    0.18142 -1.325    0.188
## age_group_5_40to44   0.17873    0.18008  0.983    0.327
## age_group_5_45to49  -0.05770    0.17475 -0.330    0.741
## age_group_5_50to54  -0.02759    0.17902 -0.154    0.878
## age_group_5_55to59  -0.01993    0.18858 -0.106    0.916
## age_group_5_60to64 -0.31150    0.24978 -1.249    0.215
## age_group_5_65_over NA NA NA NA
##
## Residual standard error: 0.3363 on 100 degrees of freedom
## (8 observations deleted due to missingness)
## Multiple R-squared: 0.1589, Adjusted R-squared: 0.07478
## F-statistic: 1.889 on 10 and 100 DF, p-value: 0.05533

linearMod40 <- lm(formula = performance_rating ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_hourly_regression)

summary(linearMod40)
## Call:
## lm(formula = performance_rating ~ gender_Female + gender_Male +
## race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 +
## age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 +
## age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 +
## age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over,
## data = merit_raises_combined_hourly_regression)
##
## Residuals:
## Min 1Q Median 3Q Max
## -0.68129 -0.24287 -0.04171 0.19235 0.69532
##
## Coefficients: (3 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    3.500910   0.162366 21.562 <2e-16 ***
## gender_Female  0.003766   0.076429   0.049 0.9608
## gender_Male NA NA NA NA
## race_grouping_white  0.140193   0.077622  1.806 0.0739 .
## race_grouping_person_of_color NA NA NA NA
## age_group_5_under_25 -0.342986   0.288645 -1.188 0.2376
## age_group_5_25to29 -0.103155   0.166335 -0.620 0.5366
## age_group_5_30to34 -0.131600   0.182655 -0.720 0.4729
## age_group_5_35to39 -0.239073   0.184425 -1.296 0.1979
## age_group_5_40to44  0.180380   0.185744  0.971 0.3339
## age_group_5_45to49 -0.056883   0.176417 -0.322 0.7478
## age_group_5_50to54 -0.027612   0.179923 -0.153 0.8783
## age_group_5_55to59 -0.018268   0.192492 -0.095 0.9246
## age_group_5_60to64 -0.310280   0.251966 -1.231 0.2211
## age_group_5_65_over NA NA NA NA
##
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.338 on 99 degrees of freedom
## (8 observations deleted due to missingness)
## Multiple R-squared: 0.1589, Adjusted R-squared: 0.06545
## F-statistic: 1.7 on 11 and 99 DF, p-value: 0.08412

### Commercial

#### Gender

current_commercial_gender_salaried <- commercial_salaried %>%
  group_by(gender)
current_commercial_gender_salaried <- current_commercial_gender_salaried %>%
  summarise(
    count = length(current_base_pay)
  )
suppress(current_commercial_gender_salaried)

---

# A tibble: 2 x 2
#  gender count
#  <chr> <int>
# 1 Female  86
# 2 Male    47

117
current_commercial_gender_hourly <- commercial_hourly %>% group_by(gender)
current_commercial_gender_hourly <- current_commercial_gender_hourly %>% summarise(
  count = length(current_base_pay)
)
suppress(current_commercial_gender_hourly)

## # A tibble: 2 x 2
##   gender  count
##     <chr> <int>
## 1  Female    74
## 2   Male     73

current_commercial_gender_salaried_median <- commercial_salaried %>% group_by(gender)
current_commercial_gender_salaried_median <- current_commercial_gender_salaried_median %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_commercial_gender_salaried_median)

## # A tibble: 2 x 3
##   gender  count median
##     <chr> <int>   <dbl>
## 1  Female    86 85977.
## 2   Male     47 86880

current_commercial_gender_hourly_median <- commercial_hourly %>% group_by(gender)
current_commercial_gender_hourly_median <- current_commercial_gender_hourly_median %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_commercial_gender_hourly_median)

## # A tibble: 2 x 3
##   gender  count median
##     <chr> <int>   <dbl>
## 1  Female    74  28.9
## 2   Male     73  23.4

current_commercial_gender_age_salaried <- commercial_salaried %>% group_by(gender)
current_commercial_gender_age_salaried <- current_commercial_gender_salaried_median %>% summarise(
  median_age = median(age)
)

## # A tibble: 2 x 2
##   gender median_age
##     <chr>     <dbl>
## 1  Female       32
## 2   Male       39

current_commercial_gender_age_hourly <- commercial_hourly %>% group_by(gender)
current_commercial_gender_age_hourly <- current_commercial_gender_salaried_median %>% summarise(
  median_age = median(age)
)

## # A tibble: 2 x 2
##   gender median_age
##     <chr>     <dbl>
## 1  Female       32
## 2   Male       39
## 1 Female 43.5
## 2 Male 47

current_commercial_gender_age_5_salary <- commercial_salaried %>% group_by(age_group_5, gender)
current_commercial_gender_age_5_salary <- current_commercial_gender_age_5_salary %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_commercial_gender_age_5_salary)

## # A tibble: 14 x 4
## # Groups: age_group_5 [9]
## age_group_5 gender count median
## <fct> <chr> <int> <dbl>
## 1 <25 Female 8 63500
## 2 25-29 Female 29 75000
## 3 25-29 Male 6 79140
## 4 30-34 Female 9 100000
## 5 30-34 Male 7 97696.
## 6 35-39 Female 9 149101
## 7 35-39 Male 9 77627.
## 8 40-44 Female 8 124288.
## 9 45-49 Female 7 90585
## 10 45-49 Male 6 85090.
## 11 50-54 Female 7 90669.
## 12 55-59 Female 5 96780
## 13 55-59 Male 5 97135.
## 14 60-64 Male 6 95754.

current_commercial_gender_age_5_hourly <- commercial_hourly %>% group_by(age_group_5, gender)
current_commercial_gender_age_5_hourly <- current_commercial_gender_age_5_hourly %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_commercial_gender_age_5_hourly)

## # A tibble: 18 x 4
## # Groups: age_group_5 [10]
## age_group_5 gender count median
## <fct> <chr> <int> <dbl>
## 1 <25 Male 7 23.1
## 2 25-29 Female 14 31.8
## 3 25-29 Male 8 26.2
## 4 30-34 Female 6 30.3
## 5 35-39 Female 5 30.8
## 6 35-39 Male 8 30.6
## 7 40-44 Female 12 29.5
## 8 40-44 Male 5 21.5
## 9 45-49 Female 7 31.3
## 10 45-49 Male 10 22.4
## 11 50-54 Female 6 23.3
## 12 50-54 Male 12 24.1
## 13 55-59 Female 9 26.4
## 14 55-59 Male 7 23.4
## 15 60-64 Female 6 24.5

119
## 16 60-64 Male 7 24.3
## 17 65+ Female 5 27.7
## 18 65+ Male 6 22.7

current_commercial_gender_age_10_salary <- commercial_salaried %>%
group_by(age_group_10, gender)
current_commercial_gender_age_10_salary <- current_commercial_gender_age_10_salary %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_commercial_gender_age_10_salary)

## # A tibble: 9 x 4
## # Groups: age_group_10 
## # age_group_10 gender count median
## <fct>      <chr> <int> <dbl>
## 1 <25       Female 8 63500
## 2 25-34    Female 38 80212
## 3 25-34    Male 13 86880
## 4 35-44    Female 17 143576.
## 5 35-44    Male 10 84029.
## 6 45-54    Female 14 90627.
## 7 45-54    Male 9 85000
## 8 55-64    Female 9 96780
## 9 55-64    Male 11 97135.

current_commercial_gender_age_10_hourly <- commercial_hourly %>%
group_by(age_group_10, gender)
current_commercial_gender_age_10_hourly <- current_commercial_gender_age_10_hourly %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_commercial_gender_age_10_hourly)

## # A tibble: 11 x 4
## # Groups: age_group_10 
## # age_group_10 gender count median
## <fct>      <chr> <int> <dbl>
## 1 <25       Male 7 23.1
## 2 25-34    Female 20 31.0
## 3 25-34    Male 11 26.0
## 4 35-44    Female 17 29.7
## 5 35-44    Male 13 27.2
## 6 45-54    Female 13 26.1
## 7 45-54    Male 22 23.5
## 8 55-64    Female 15 25.4
## 9 55-64    Male 14 23.9
## 10 65+     Female 5 27.7
## 11 65+     Male 6 22.7

current_commercial_gender_salaried_under_40 <- filter(commercial_salaried, age < 40) %>%
group_by(gender)
current_commercial_gender_salaried_under_40 <- current_commercial_gender_salaried_under_40 %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_commercial_gender_salaried_under_40)

## # A tibble: 2 x 3
# gender count median
# <chr>  <int>  <dbl>
# 1 Female  55 80424
# 2 Male   24 83140  
current_commercial_gender_salaried_over_40 <- filter(commercial_salaried, age > 39)
%>% group_by(gender)
%>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_commercial_gender_salaried_over_40)

## # A tibble: 2 x 3
## #  gender count median
## #  <chr> <int> <dbl>
## 1 Female  31  96780
## 2 Male   23  90000  
current_commercial_gender_hourly_under_40 <- filter(commercial_hourly, age < 40)
%>% group_by(gender)
%>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_commercial_gender_hourly_under_40)

## # A tibble: 2 x 3
## #  gender count median
## #  <chr> <int> <dbl>
## 1 Female  29  30.4
## 2 Male   26  26.5  
current_commercial_gender_hourly_over_40 <- filter(commercial_hourly, age > 39)
%>% group_by(gender)
%>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_commercial_gender_hourly_over_40)

## # A tibble: 2 x 3
## #  gender count median
## #  <chr> <int> <dbl>
## 1 Female  45  27.7
## 2 Male   47  23.2  

Race and ethnicity

current_commercial_race_salaried <- commercial_salaried
%>% group_by(race_ethnicity)
%>% summarise(
  count = length(current_base_pay)
)
suppress_count(current_commercial_race_salaried)

## # A tibble: 4 x 2
## #  race_ethnicity  count
## #  <chr>          <int>
## 1 White (United States of America) 99
## 2 Black or African American (United States of America) 14
## 3 Asian (United States of America) 13
## 4 Hispanic or Latino (United States of America) 5

```r
current_commercial_race_hourly <- commercial_hourly %>%
group_by(race_ethnicity) %>%
summarise(count = length(current_base_pay))

suppress_count(current_commercial_race_hourly)
```

## # A tibble: 4 x 2
## race_ethnicity     count
## <chr>              <int>
## 1 Black or African American (United States of America) 82
## 2 White (United States of America) 43
## 3 Hispanic or Latino (United States of America) 9
## 4 Asian (United States of America) 7

```r
current_commercial_race_group_salaried <- commercial_salaried %>%
group_by(race_grouping) %>%
summarise(count = length(current_base_pay))

suppress_count(current_commercial_race_group_salaried)
```

## # A tibble: 2 x 2
## race_grouping     count
## <chr>              <int>
## 1 white             99
## 2 person of color   32

```r
current_commercial_race_group_hourly <- commercial_hourly %>%
group_by(race_grouping) %>%
summarise(count = length(current_base_pay))

suppress_count(current_commercial_race_group_hourly)
```

## # A tibble: 2 x 2
## race_grouping     count
## <chr>              <int>
## 1 person of color 101
## 2 white            43

```r
current_commercial_race_salaried_median <- commercial_salaried %>%
group_by(race_ethnicity) %>%
summarise(count = length(current_base_pay),
          median = median(current_base_pay, na.rm = FALSE))

suppress_median(current_commercial_race_salaried_median)
```

## # A tibble: 4 x 3
## race_ethnicity     count median
## <chr>              <int>  <dbl>
## 1 White (United States of America) 99 88000
## 2 Black or African American (United States of America) 14 84640
## 3 Asian (United States of America) 13 80000
## 4 Hispanic or Latino (United States of America) 5 80000
```
current_commercial_race_hourly_median <- commercial_hourly %>% group_by(race_ethnicity) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))

suppress_median(current_commercial_race_hourly_median)

## A tibble: 4 x 3
## race_ethnicity  count median
## <chr>           <int>  <dbl>
## 1 White (United States of America) 43 30.4
## 2 Asian (United States of America) 7 26.0
## 3 Black or African American (United States of America) 82 24.9
## 4 Hispanic or Latino (United States of America) 9 23.1

current_commercial_race_group_salaried_median <- commercial_salaried %>% group_by(race_grouping) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))

suppress_median(current_commercial_race_group_salaried_median)

## A tibble: 2 x 3
## race_grouping  count median
## <chr>           <int> <dbl>
## 1 white         99 88000
## 2 person of color 32 83445.

current_commercial_race_group_hourly_median <- commercial_hourly %>% group_by(race_grouping) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))

suppress_median(current_commercial_race_group_hourly_median)

## A tibble: 2 x 3
## race_grouping  count median
## <chr>           <int> <dbl>
## 1 white         43 30.4
## 2 person of color 101 25.2

current_commercial_race_age_salaried_median <- commercial_salaried %>% group_by(race_ethnicity) %>% summarise(median_age = median(age))

## A tibble: 5 x 2
## race_ethnicity median_age
## <chr>            <dbl>
## 1 Asian (United States of America) 32
## 2 Black or African American (United States of America) 48
## 3 Hispanic or Latino (United States of America) 41
## 4 Prefer Not to Disclose (United States of America) 35.5
## 5 White (United States of America) 35
```r
current_commercial_race_age_hourly <- commercial_hourly %>%
group_by(race_ethnicity)
  median_age = median(age)
)
## # A tibble: 7 x 2
## # Groups: race_ethnicity [7]
## race_ethnicity median_age
## <chr>         <dbl>
## 1 American Indian or Alaska Native (United States of America) 38
## 2 Asian (United States of America) 28
## 3 Black or African American (United States of America) 48.5
## 4 Hispanic or Latino (United States of America) 30
## 5 Prefer Not to Disclose (United States of America) 35
## 6 Two or More Races (United States of America) 31
## 7 White (United States of America) 39

current_commercial_race_age_5_salary <- commercial_salaried %>%
group_by(age_group_5, race_ethnicity)
## # A tibble: 9 x 4
## # Groups: age_group_5 [9]
## age_group_5 race_ethnicity count median
## <fct>             <chr>     <int>  <dbl>
## 1 <25              White (United States of America) 9 63000
## 2 25-29            White (United States of America) 28 78692.
## 3 30-34            White (United States of America) 12 98848.
## 4 35-39            White (United States of America) 13 149101
## 5 40-44            White (United States of America) 6 126865.
## 6 45-49            White (United States of America) 7 90000
## 7 50-54            White (United States of America) 9 87392.
## 8 55-59            White (United States of America) 8 96957.
## 9 60-64            White (United States of America) 6 97651.

current_commercial_race_age_5_hourly <- commercial_hourly %>%
group_by(age_group_5, race_ethnicity)
## # A tibble: 11 x 4
## # Groups: age_group_5 [9]
## age_group_5 race_ethnicity count median
## <fct>             <chr>     <int>  <dbl>
## 1 <25              Black or African American (United States of America) 5 22.4
## 2 25-29            White (United States of America) 11 31.8
## 3 35-39            White (United States of America) 6 30.8
## 4 40-44            Black or African American (United States of America) 13 28.9
## 5 45-49            Black or African American (United States of America) 14 23.1
## 6 50-54            Black or African American (United States of America) 12 23.3
## 7 50-54            White (United States of America) 5 24.4
```
```
# A tibble: 6 x 4
# Groups: age_group_10 [5]
# age_group_10 race_ethnicity count median
## <fct> <chr> <int> <dbl>
## 1 <25 White (United States of America) 9 63000
## 2 25-34 Asian (United States of America) 6 82418.
## 3 25-34 White (United States of America) 40 82000
## 4 35-44 White (United States of America) 19 148730.
## 5 45-54 White (United States of America) 16 88696.
## 6 55-64 White (United States of America) 14 97325.
# A tibble: 11 x 4
# Groups: age_group_10 [6]
# age_group_10 race_ethnicity count median
## <fct> <chr> <int> <dbl>
## 1 <25 Black or African American (United States of A~ 5 22.4
## 2 25-34 Black or African American (United States of A~ 7 26.7
## 3 25-34 Hispanic or Latino (United States of America) 6 25.0
## 4 25-34 White (United States of America) 12 31.8
## 5 35-44 Black or African American (United States of A~ 17 29.2
## 6 35-44 White (United States of America) 8 30.6
## 7 45-54 Black or African American (United States of A~ 26 23.3
## 8 45-54 White (United States of America) 8 30.8
## 9 55-64 Black or African American (United States of A~ 22 24.5
## 10 55-64 White (United States of America) 7 26.4
## 11 65+ Black or African American (United States of A~ 5 23.4
```

```
```
```r
## # A tibble: 14 x 4
## # Groups: age_group_5 
## age_group_5 race_grouping count median
## <fct> <chr> <int> <dbl>
## 1 <25 person of color 7 25.6
## 2 25-29 person of color 10 26.3
## 3 25-29 white 11 31.8
## 4 30-34 person of color 8 28.8
## 5 35-39 person of color 6 30.8
## 6 35-39 white 6 30.8
## 7 40-44 person of color 14 28.5
## 8 45-49 person of color 14 23.1
## 9 50-54 person of color 13 23.2
## 10 50-54 white 5 24.4
## 11 55-59 person of color 11 27.0
## 12 55-59 white 5 25.4
## 13 60-64 person of color 11 24.3
## 14 65+ person of color 7 23.4

current_commercial_race_group_age_5_hourly <- commercial_hourly %>%
group_by(age_group_5, race_grouping) %>%
summarise(count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE))
suppress(current_commercial_race_group_age_5_hourly)
```

```r
## # A tibble: 9 x 4
## # Groups: age_group_10 
## age_group_10 race_grouping count median
## <fct> <chr> <int> <dbl>
## 1 <25 white 9 63000
## 2 25-34 person of color 10 74918.
## 3 25-34 white 40 82000
## 4 35-44 person of color 7 90431.

current_commercial_race_group_age_10_salary <- commercial_salaried %>%
group_by(age_group_10, race_grouping) %>%
summarise(count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE))
suppress(current_commercial_race_group_age_10_salary)
```

126
current_commercial_race_group_age_10_hourly <- commercial_hourly %>% group_by(age_group_10, race_grouping) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))

## # A tibble: 10 x 4
## # Groups: age_group_10 [6]
## # groups: age_group_10 [6]
## age_group_10 race_grouping count median
## <fct> <chr> <int> <dbl>
## 1 <25 person of color 7 25.6
## 2 25-34 person of color 18 26.5
## 3 25-34 white 12 31.8
## 4 35-44 person of color 20 29.1
## 5 35-44 white 8 30.6
## 6 45-54 person of color 27 23.2
## 7 45-54 white 8 30.8
## 8 55-64 person of color 22 24.5
## 9 55-64 white 7 26.4
## 10 65+ person of color 7 23.4

current_commercial_race_salaried_under_40 <- filter(commercial_salaried, age < 40) %>% group_by(race_ethnicity) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))

## # A tibble: 2 x 3
## race_ethnicity count median
## <chr> <int> <dbl>
## 1 White (United States of America) 62 82000
## 2 Asian (United States of America) 10 77418.

current_commercial_race_salaried_over_40 <- filter(commercial_salaried, age > 39) %>% group_by(race_ethnicity) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))

## # A tibble: 2 x 3
## race_ethnicity count median
## <chr> <int> <dbl>
## 1 White (United States of America) 37 97135.
## 2 Black or African American (United States of America) 10 84849.

current_commercial_race_hourly_under_40 <- filter(commercial_hourly, age < 40) %>% group_by(race_ethnicity) %>% summarise(count = length(current_base_pay),

## # A tibble: 2 x 3
## race_ethnicity count median
## <chr> <int> <dbl>
## 1 White (United States of America) 37 97135.
## 2 Black or African American (United States of America) 10 84849.
median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_commercial_race_hourly_under_40)

## # A tibble: 3 x 3
## race_ethnicity        count median
## <chr>           <int> <dbl>
## 1 White (United States of America)   22 31.5
## 2 Black or African American (United States of America) 16 26.5
## 3 Hispanic or Latino (United States of America)    8 25.6

current_commercial_race_hourly_over_40 <- filter(commercial_hourly, age > 39) %>% group_by(race_ethnicity)
current_commercial_race_hourly_over_40 <- current_commercial_race_hourly_over_40 %>% summarise( count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE) )
suppress_median(current_commercial_race_hourly_over_40)

## # A tibble: 2 x 3
## race_ethnicity  count median
## <chr>          <int> <dbl>
## 1 White (United States of America)  21 29.2
## 2 Black or African American (United States of America) 66 24.4

Gender x race/ethnicity

current_commercial_race_gender_salaried <- commercial_salaried %>% group_by(race_ethnicity, gender)
current_commercial_race_gender_salaried <- current_commercial_race_gender_salaried %>% summarise( count = length(current_base_pay) )
suppress(current_commercial_race_gender_salaried)

## # A tibble: 6 x 3
## # Groups: race_ethnicity [3]
## race_ethnicity gender count
## <chr>     <chr>  <int>
## 1 Asian (United States of America) Female     8
## 2 Asian (United States of America)  Male      5
## 3 Black or African American (United States of America) Female  7
## 4 Black or African American (United States of America)  Male  7
## 5 White (United States of America) Female  67
## 6 White (United States of America)  Male  32

current_commercial_race_gender_hourly <- commercial_hourly %>% group_by(race_ethnicity, gender)
current_commercial_race_gender_hourly <- current_commercial_race_gender_hourly %>% summarise( count = length(current_base_pay) )
suppress(current_commercial_race_gender_hourly)

## # A tibble: 5 x 3
## # Groups: race_ethnicity [3]
## race_ethnicity gender count
## <chr>     <chr>  <int>
## 1 Black or African American (United States of America) Female  41
## 2 Black or African American (United States of America)  Male  41

128
current_commercial_race_gender_median_salaried <- commercial_salaried %>%
group_by(race_ethnicity, gender)
current_commercial_race_gender_median_salaried <- current_commercial_race_gender_median_salaried %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))
suppress(current_commercial_race_gender_median_salaried)

current_commercial_race_gender_hourly_median <- commercial_hourly %>%
group_by(race_ethnicity, gender)
current_commercial_race_gender_hourly_median <- current_commercial_race_gender_hourly_median %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))
suppress(current_commercial_race_gender_hourly_median)

current_commercial_race_gender_salaried_under_40 <- filter(commercial_salaried, age < 40) %>%
group_by(race_ethnicity, gender)
current_commercial_race_gender_salaried_under_40 <- current_commercial_race_gender_salaried_under_40 %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))
suppress(current_commercial_race_gender_salaried_under_40)

current_commercial_race_gender_salaried_over_40 <- filter(commercial_salaried, age > 39) %>%
group_by(race_ethnicity, gender)
current_commercial_race_gender_salaried_over_40 <- current_commercial_race_gender_salaried_over_40 %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))
suppress(current_commercial_race_gender_salaried_over_40)
count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_commercial_race_gender_salaried_over_40)

## # A tibble: 3 x 4
## # Groups: race_ethnicity [2]
## race_ethnicity                      gender count median
## <chr>                               <chr> <int> <dbl>
## 1 Black or African American (United States of America) Female  6 94950.
## 2 White (United States of America) Female 21 97546
## 3 White (United States of America) Male 16 95564.

current_commercial_race_gender_hourly_under_40 <- filter(commercial_hourly, age < 40) %>% group_by(race_ethnicity, gender) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))
suppress(current_commercial_race_gender_hourly_under_40)

## # A tibble: 5 x 4
## # Groups: race_ethnicity [3]
## race_ethnicity                      gender count median
## <chr>                               <chr> <int> <dbl>
## 1 Black or African American (United States of America) Female  8 26.5
## 2 Black or African American (United States of America) Male  8 26.3
## 3 Hispanic or Latino (United States of America) Female  6 28.5
## 4 White (United States of America) Female 12 33.3
## 5 White (United States of America) Male 10 30.6

current_commercial_race_gender_hourly_over_40 <- filter(commercial_hourly, age > 39) %>% group_by(race_ethnicity, gender) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))
suppress(current_commercial_race_gender_hourly_over_40)

## # A tibble: 4 x 4
## # Groups: race_ethnicity [2]
## race_ethnicity                      gender count median
## <chr>                               <chr> <int> <dbl>
## 1 Black or African American (United States of America) Female 33 26.1
## 2 Black or African American (United States of America) Male 33 23.1
## 3 White (United States of America) Female 10 31.0
## 4 White (United States of America) Male 11 23.8

Years of service

current_commercial_yos_salaried <- commercial_salaried %>% group_by(years_of_service_grouped) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))
suppress(current_commercial_yos_salaried)
current_commercial_yos_hourly <- commercial_hourly %>% group_by(years_of_service_grouped) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))

current_commercial_yos_gender_salaried <- commercial_salaried %>% group_by(years_of_service_grouped, gender) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))

current_commercial_yos_gender_hourly <- commercial_hourly %>% group_by(years_of_service_grouped, gender) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))
suppress(current_commercial_yos_gender_hourly)

## A tibble: 13 x 4
## Groups: years_of_service_grouped [8]
## years_of_service_grouped gender count median
## <fct> <chr> <int> <dbl>
## 1 0 Female 10 29.5
## 2 0 Male 16 22.0
## 3 1-2 Female 18 30.3
## 4 1-2 Male 15 24.4
## 5 3-5 Female 5 30.8
## 6 3-5 Male 9 22.1
## 7 6-10 Female 5 26.3
## 8 6-10 Male 14 23.6
## 9 11-15 Male 10 29.0
## 10 16-20 Female 10 24.2
## 11 16-20 Male 7 27.3
## 12 21-25 Female 8 27.9
## 13 25+ Female 14 26.6

current_commercial_yos_race_salaried <- commercial_salaried %>% group_by(years_of_service_grouped, race_ethnicity)
current_commercial_yos_race_salaried <- current_commercial_yos_race_salaried %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_commercial_yos_race_salaried)

## A tibble: 6 x 4
## Groups: years_of_service_grouped [6]
## years_of_service_grouped race_ethnicity count median
## <fct> <chr> <int> <dbl>
## 1 0 White (United States of America) 23 82000
## 2 1-2 White (United States of America) 30 80212
## 3 3-5 White (United States of America) 19 108780
## 4 6-10 White (United States of America) 11 102500
## 5 16-20 White (United States of America) 5 87392.
## 6 21-25 White (United States of America) 6 97651.

current_commercial_yos_race_hourly <- commercial_hourly %>% group_by(years_of_service_grouped, race_ethnicity)
current_commercial_yos_race_hourly <- current_commercial_yos_race_hourly %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_commercial_yos_race_hourly)

## A tibble: 13 x 4
## Groups: years_of_service_grouped [8]
## years_of_service_grouped race_ethnicity count median
## <fct> <chr> <int> <dbl>
## 1 0 Black or African American (United St~ 11 25.6
## 2 0 White (United States of America) 6 29.5
## 3 1-2 Black or African American (United St~ 14 23.6
## 4 1-2 White (United States of America) 13 34.7
## 5 3-5 Black or African American (United St~ 6 21.8
## 6 3-5 White (United States of America) 5 23.2
## 7 6-10 Black or African American (United States of America) 12 23.6
## 8 6-10 White (United States of America) 6 29.9
## 9 11-15 Black or African American (United States of America) 7 30.4
## 10 11-15 White (United States of America) 6 26.0
## 11 16-20 Black or African American (United States of America) 12 24.1
## 12 21-25 Black or African American (United States of America) 9 29.7
## 13 25+ Black or African American (United States of America) 11 24.7

```r
current_commercial_yos_race_gender_salaried <- commercial_salaried %>%
group_by(years_of_service_grouped, race_ethnicity, gender) %>%
summarise(count = length(current_base_pay),
          median = median(current_base_pay, na.rm = FALSE))
```

## # A tibble: 7 x 5
## # Groups: years_of_service_grouped, race_ethnicity [4]
##   years_of_service_grouped race_ethnicity gender count median
##              <fct>               <chr> <chr>  <int> <dbl>
## 1 0 White (United States of America) Female 15 7.43e4
## 2 0 White (United States of America) Male 8 9.25e4
## 3 1-2 White (United States of America) Female 21 7.74e4
## 4 1-2 White (United States of America) Male 9 8.33e4
## 5 3-5 White (United States of America) Female 14 9.41e4
## 6 3-5 White (United States of America) Male 5 1.26e5
## 7 6-10 White (United States of America) Female 10 1.01e5

```r
current_commercial_yos_race_gender_hourly <- commercial_hourly %>%
group_by(years_of_service_grouped, race_ethnicity, gender) %>%
summarise(count = length(current_base_pay),
          median = median(current_base_pay, na.rm = FALSE))
```

## # A tibble: 11 x 5
## # Groups: years_of_service_grouped, race_ethnicity [10]
##   years_of_service_grouped race_ethnicity gender count median
##                   <fct>                   <chr> <chr>  <int> <dbl>
## 1 0 Black or African American (United States of America) Male 7 20.6
## 2 1-2 Black or African American (United States of America) Male 6 25.8
## 3 1-2 Black or African American (United States of America) Male 8 21.9
## 4 1-2 White (United States of America) Female 9 35.0
## 5 3-5 Black or African American (United States of America) Male 5 21.5
## 6 6-10 Black or African American (United States of America) Male 10 23.4
## 7 11-15 Black or African American (United States of America) Male 5 29.9
## 8 11-15 White (United States of America) Male 5 26.8
## 9 16-20 Black or African American (United States of America) Female 8 23.7
## 10 21-25 Black or African American (United States of America) Female 8 27.9
## 11 25+ Black or African American (United States of America) Female 10 25.5

**Age**

```r
current_median_commercial_age_5_salaried <- commercial_salaried %>%
group_by(age_group_5) %>%
summarise(count = length(current_base_pay),
          median = median(current_base_pay, na.rm = FALSE))
```

## # A tibble: 133
median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_median_commercial_age_5_salaried)

## # A tibble: 9 x 3
## #  age_group_5 count median
##  <fct> <int> <dbl>
## 1 <25 10 64000
## 2 25-29 35 75000
## 3 30-34 16 98848.
## 4 35-39 18 101092.
## 5 40-44 9 143576.
## 6 45-49 13 86105.
## 7 50-54 10 87002.
## 8 55-59 10 96957.
## 9 60-64 10 95754.

current_median_commercial_age_5_hourly <- commercial_hourly %>%
  group_by(age_group_5) %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )
suppress(current_median_commercial_age_5_hourly)

## # A tibble: 10 x 3
## #  age_group_5 count median
##  <fct> <int> <dbl>
## 1 <25 11 25.6
## 2 25-29 22 29.8
## 3 30-34 9 29.5
## 4 35-39 13 30.8
## 5 40-44 17 28.9
## 6 45-49 17 24.0
## 7 50-54 18 23.6
## 8 55-59 16 26.2
## 9 60-64 13 24.3
##10 65+ 11 23.4

current_median_commercial_age_10_salaried <- commercial_salaried %>%
  group_by(age_group_10) %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )
suppress(current_median_commercial_age_10_salaried)

## # A tibble: 5 x 3
## #  age_group_10 count median
##  <fct> <int> <dbl>
## 1 <25 10 64000
## 2 25-34 51 82000
## 3 35-44 27 105000
## 4 45-54 23 86613
## 5 55-64 20 96957.

current_median_commercial_age_10_hourly <- commercial_hourly %>%
  group_by(age_group_10)
current_median_commercial_age_10_hourly <- current_median_commercial_age_10_hourly %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_median_commercial_age_10_hourly)

## # A tibble: 6 x 3
## #   age_group_10 count median
## <fct>      <int> <dbl>
## 1 <25       11    25.6
## 2 25-34     31    29.5
## 3 35-44     30    29.2
## 4 45-54     35    23.8
## 5 55-64     29    24.7
## 6 65+       11    23.4

current_commercial_age_5_yos_salary <- commercial_salaried %>% group_by(age_group_5, years_of_service_grouped) %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_commercial_age_5_yos_salary)

## # A tibble: 9 x 4
## # Groups: age_group_5 [6]
##   age_group_5 years_of_service_grouped count median
## <fct>                  <fct>      <int> <dbl>
## 1 <25                   0           6 62500
## 2 25-29                 0           14 75000
## 3 25-29                 1-2         17 76000
## 4 30-34                 0           6 100000
## 5 30-34                 1-2          7 96980
## 6 35-39                 3-5         17 149101
## 7 35-39                 6-10        6 101092.
## 8 40-44                 3-5          5 167000
## 9 60-64                 21-25       5 97514.

current_commercial_age_5_yos_hourly <- commercial_hourly %>% group_by(age_group_5, years_of_service_grouped) %>% summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_commercial_age_5_yos_hourly)

## # A tibble: 10 x 4
## # Groups: age_group_5 [8]
##   age_group_5 years_of_service_grouped count median
## <fct>                  <fct>      <int> <dbl>
## 1 <25                   0           5 23.1
## 2 <25                   1-2         6 27.9
## 3 25-29                 0           6 33.3
## 4 25-29                 1-2         15 26.7
## 5 30-34                 0           5 22.0
## 6 35-39                 11-15       5 30.4
## 7 40-44                 3-5         5 29.2
### 8 55-59 25+ 6 27.9
### 9 60-64 16-20 5 24.3
### 10 65+ 25+ 5 26.8

```r
current_commercial_age_10_yos_salary <- commercial_salaried %>%
group_by(age_group_10, years_of_service_grouped)
current_commercial_age_10_yos_salary <- current_commercial_age_10_yos_salary %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )
suppress(current_commercial_age_10_yos_salary)
```

**# A tibble: 8 x 4**
**## Groups: age_group_10 [5]**
**## age_group_10 years_of_service_grouped count median**
**## <fct> <fct> <int> <dbl>**
**## 1 <25 0 6 62500**
**## 2 25-34 0 20 82000**
**## 3 25-34 1-2 24 80810.**
**## 4 25-34 3-5 5 85850**
**## 5 35-44 3-5 12 158050.**
**## 6 35-44 6-10 6 101092.**
**## 7 45-54 3-5 5 86613**
**## 8 55-64 21-25 5 97514.**

```r
current_commercial_age_10_yos_hourly <- commercial_hourly %>%
group_by(age_group_10, years_of_service_grouped)
current_commercial_age_10_yos_hourly <- current_commercial_age_10_yos_hourly %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )
suppress(current_commercial_age_10_yos_hourly)
```

**# A tibble: 15 x 4**
**## Groups: age_group_10 [6]**
**## age_group_10 years_of_service_grouped count median**
**## <fct> <fct> <int> <dbl>**
**## 1 <25 0 5 23.1**
**## 2 <25 1-2 6 27.9**
**## 3 25-34 0 11 30.3**
**## 4 25-34 1-2 15 26.7**
**## 5 35-44 0 5 29.2**
**## 6 35-44 3-5 6 26.2**
**## 7 35-44 11-15 7 30.4**
**## 8 45-54 0 5 20.5**
**## 9 45-54 1-2 5 22.4**
**## 10 45-54 6-10 7 23.8**
**## 11 45-54 16-20 6 28.3**
**## 12 55-64 6-10 6 23.4**
**## 13 55-64 16-20 6 24.3**
**## 14 55-64 25+ 8 27.9**
**## 15 65+ 25+ 5 26.8**

```r
current_median_commercial_age_5_gender_salaried <- commercial_salaried %>%
group_by(age_group_5, gender)
current_median_commercial_age_5_gender_salaried <- current_median_commercial_age_5_gender_salaried %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )
```

136
```r
## A tibble: 14 x 4
## Groups: age_group_5 [9]
## #  age_group_5 gender count median
## <fct> <chr> <int> <dbl>
## 1 <25 Female   8  63500
## 2 25-29 Female  29  75000
## 3 25-29 Male     6  79140
## 4 30-34 Female  9 100000
## 5 30-34 Male     7  97696.
## 6 35-39 Female  9 149101
## 7 35-39 Male     9  77627.
## 8 40-44 Female  8 124288.
## 9 45-49 Female  7  90585
## 10 45-49 Male    7  97627.
## 11 50-54 Female  7  90669.
## 12 55-59 Female  5  96780
## 13 55-59 Male    5  97135.
## 14 60-64 Male    6  95754.
```

```r
current_median_commercial_age_5_gender_salaried <- commercial_salaried %>%
group_by(age_group_10, gender) %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
suppress(current_median_commercial_age_5_gender_salaried)
```

## A tibble: 18 x 4
## Groups: age_group_5 [10]
## #  age_group_5 gender count median
## <fct> <chr> <int> <dbl>
## 1 <25 Male     7  23.1
## 2 25-29 Female 14  31.8
## 3 25-29 Male   8  26.2
## 4 30-34 Female  6  30.3
## 5 35-39 Female  5  30.8
## 6 35-39 Male   8  30.6
## 7 40-44 Female 12  29.5
## 8 40-44 Male   5  21.5
## 9 45-49 Female  7  31.3
## 10 45-49 Male  10  22.4
## 11 50-54 Female  6  23.3
## 12 50-54 Male  12  24.1
## 13 55-59 Female  9  26.4
## 14 55-59 Male   7  23.4
## 15 60-64 Female  6  24.5
## 16 60-64 Male   7  24.3
## 17 65+ Female   5  27.7
## 18 65+ Male     6  22.7
```

```r
current_median_commercial_age_10_gender_salaried <- commercial_salaried %>%
group_by(age_group_10, gender) %>%
summarise(
  count = length(current_base_pay),
  median = median(current_base_pay, na.rm = FALSE)
)
```
## # A tibble: 9 x 4
## # Groups: age_group_10 [5]
## # age_group_10 gender count median
## <fct> <chr> <int> <dbl>
## 1 <25 Female 8 63500
## 2 25-34 Female 38 80212
## 3 25-34 Male 13 86880
## 4 35-44 Female 17 143576.
## 5 35-44 Male 10 84029.
## 6 45-54 Female 14 90627.
## 7 45-54 Male 9 85000
## 8 55-64 Female 9 96780
## 9 55-64 Male 11 97135.

current_median_commercial_age_10_gender_hourly <- commercial_hourly %>% group_by(age_group_10, gender)
current_median_commercial_age_10_gender_hourly <- current_median_commercial_age_10_gender_hourly %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))
suppress(current_median_commercial_age_10_gender_hourly)

## # A tibble: 11 x 4
## # Groups: age_group_5 [6]
## # age_group_5 gender count median
## <fct> <chr> <int> <dbl>
## 1 <25 Male 7 23.1
## 2 25-29 Female 20 31.0
## 3 25-34 Male 11 26.0
## 4 35-44 Female 17 29.7
## 5 35-44 Male 13 27.2
## 6 45-54 Female 13 26.1
## 7 45-54 Male 22 23.5
## 8 55-64 Female 15 25.4
## 9 55-64 Male 14 23.9
## 10 65+ Female 5 27.7
## 11 65+ Male 6 22.7

current_median_commercial_age_5_race_salaried <- commercial_salaried %>% group_by(age_group_5, race_ethnicity)
current_median_commercial_age_5_race_salaried <- current_median_commercial_age_5_race_salaried %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))
suppress(current_median_commercial_age_5_race_salaried)

## # A tibble: 9 x 4
## # Groups: age_group_5 [9]
## # age_group_5 race_ethnicity count median
## <fct> <chr> <int> <dbl>
## 1 <25 White (United States of America) 9 63000
## 2 25-29 White (United States of America) 28 78692.
## 3 30-34 White (United States of America) 12 98848.
## 4 35-39 White (United States of America) 13 149101

138
current_median_commercial_age_5_race_hourly <- commercial_hourly %>%
group_by(age_group_5, race_ethnicity) %>%
summarise(count = length(current_base_pay),
          median = median(current_base_pay, na.rm = FALSE))
suppress(current_median_commercial_age_5_race_hourly)

current_median_commercial_age_10_race_salaried <- commercial_salaried %>%
group_by(age_group_10, race_ethnicity) %>%
summarise(count = length(current_base_pay),
           median = median(current_base_pay, na.rm = FALSE))
suppress(current_median_commercial_age_10_race_salaried)

current_median_commercial_age_10_race_hourly <- commercial_hourly %>%
group_by(age_group_10, race_ethnicity) %>%
summarise(count = length(current_base_pay),
           median = median(current_base_pay, na.rm = FALSE))
suppress(current_median_commercial_age_10_race_hourly)
## age_group race_ethnicity count median
## <fct>       <chr>        <int>   <dbl>
## 1 <25        Black or African American (United States of A-  5  22.4
## 2 25-34      Black or African American (United States of A-  7  26.7
## 3 25-34      Hispanic or Latino (United States of America)  6  25.0
## 4 25-34      White (United States of America)                12  31.8
## 5 35-44      Black or African American (United States of A- 17  29.2
## 6 35-44      White (United States of America)                 8  30.6
## 7 45-54      Black or African American (United States of A- 26  23.3
## 8 45-54      White (United States of America)                 8  30.8
## 9 55-64      Black or African American (United States of A- 22  24.5
## 10 55-64     White (United States of America)                 7  26.4
## 11 65+       Black or African American (United States of A-  5  23.4

current_median_commercial_age_5_race_group_salaried <- commercial_salaried %>% group_by(age_group_5, race_grouping) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))

suppress(current_median_commercial_age_5_race_group_salaried)

## # A tibble: 12 x 4
## # Groups: age_group_5 [9]
## age_group_5 race_grouping count median
## <fct>       <chr>        <int>   <dbl>
## 1 <25        white         9 63000
## 2 25-29      person of color 7 72000
## 3 25-29      white         28 78692.
## 4 30-34      white         12 96848.
## 5 35-39      person of color 5 73522.
## 6 35-39      white         13 149101
## 7 40-44      white         6 126865.
## 8 45-49      person of color 6 85450.
## 9 45-49      white         7 90000
## 10 50-54     white         9 87392.
## 11 55-59     white         8 96957.
## 12 60-64     white         6 97651.

current_median_commercial_age_5_race_group_hourly <- commercial_hourly %>% group_by(age_group_5, race_grouping) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))

suppress(current_median_commercial_age_5_race_group_hourly)

## # A tibble: 14 x 4
## # Groups: age_group_5 [10]
## age_group_5 race_grouping count median
## <fct>       <chr>        <int>   <dbl>
## 1 <25        person of color 7 25.6
## 2 25-29      person of color 10 26.3
## 3 25-29      white         11 31.8
## 4 30-34      person of color 8 28.8
## 5 35-39      person of color 6 30.8
## 6 35-39      white         6 30.8

140
## 7 40-44 person of color 14 28.5
## 8 45-49 person of color 14 23.1
## 9 50-54 person of color 13 23.2
## 10 50-54 white 5 24.4
## 11 55-59 person of color 11 27.0
## 12 55-59 white 5 25.4
## 13 60-64 person of color 11 24.3
## 14 65+ person of color 7 23.4

```r
current_median_commercial_age_10_race_group_salaried <- commercial_salaried %>%
group_by(age_group_10, race_grouping) %>%
summarise(count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE))

suppress(current_median_commercial_age_10_race_group_salaried)
```

## # A tibble: 9 x 4
## # Groups: age_group_10 [5]
## #   age_group_10 race_grouping count median
## # <fct>          <chr>     <int> <dbl>
## 1 <25           white      9  63000
## 2 25-34         person of color 10 74918
## 3 25-34         white      40 82000
## 4 35-44         person of color 7 90431
## 5 35-44         white      19 148730
## 6 45-54         person of color 7 85000
## 7 45-54         white      16 88696
## 8 55-64         person of color 6 82709
## 9 55-64         white      14 97325

```r
current_median_commercial_age_10_race_group_hourly <- commercial_hourly %>%
group_by(age_group_10, race_grouping) %>%
summarise(count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE))

suppress(current_median_commercial_age_10_race_group_hourly)
```

## # A tibble: 10 x 4
## # Groups: age_group_10 [6]
## #   age_group_10 race_grouping count median
## # <fct>          <chr>     <int> <dbl>
## 1 <25           person of color 7  25.6
## 2 25-34         person of color 18  26.5
## 3 25-34         white      12  31.8
## 4 35-44         person of color 20  29.1
## 5 35-44         white      8  30.6
## 6 45-54         person of color 27  23.2
## 7 45-54         white      8  30.8
## 8 55-64         person of color 22  24.5
## 9 55-64         white      7  26.4
##10 65+           person of color 7  23.4

```r
current_median_commercial_age_5_race_gender_salaried <- commercial_salaried %>%
group_by(age_group_5, race_ethnicity, gender) %>%
summarise(count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE))
```

```r
current_median_commercial_age_5_race_gender_salaried <- current_median_commercial_age_5_race_gender_salaried %>%
group_by(age_group_5, race_ethnicity, gender) %>%
summarise(count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE))
```
suppress(current_median_commercial_age_5_race_gender_salaried)

## A tibble: 8 x 5
## Groups: age_group_5, race_ethnicity [7]

## # A tibble: 8 x 5
## # Groups: age_group_5, race_ethnicity [7]
## age_group_5 race_ethnicity gender count median
## <fct> <chr> <chr> <int> <dbl>
## 1 <25 White (United States of America) Female 7 62000
## 2 25-29 White (United States of America) Female 25 76000
## 3 30-34 White (United States of America) Female 5 131097.
## 4 30-34 White (United States of America) Male 7 97696.
## 5 35-39 White (United States of America) Female 9 149101
## 6 40-44 White (United States of America) Female 6 128865.
## 7 50-54 White (United States of America) Female 6 98281.
## 8 55-59 White (United States of America) Male 5 97135.

current_median_commercial_age_5_race_gender_hourly <- commercial_hourly %>% group_by(age_group_5, race_ethnicity, gender)
current_median_commercial_age_5_race_gender_hourly <- current_median_commercial_age_5_race_gender_hourly
  %>% summarise(count = length(current_base_pay),
                median = median(current_base_pay, na.rm = FALSE))

## A tibble: 10 x 5
## Groups: age_group_5, race_ethnicity [8]

## # A tibble: 10 x 5
## # Groups: age_group_5, race_ethnicity [8]
## age_group_5 race_ethnicity gender count median
## <fct> <chr> <chr> <int> <dbl>
## 1 <25 Black or African American (United State~ Male 5 22.4
## 2 25-29 White (United States of America) Female 7 35.0
## 3 40-44 Black or African American (United State~ Female 9 29.7
## 4 45-49 Black or African American (United State~ Male 10 22.4
## 5 50-54 Black or African American (United State~ Female 6 23.3
## 6 50-54 Black or African American (United State~ Male 6 23.0
## 7 50-54 White (United States of America) Male 5 24.4
## 8 55-59 Black or African American (United State~ Female 7 28.6
## 9 60-64 Black or African American (United State~ Female 5 24.3
## 10 60-64 Black or African American (United State~ Male 6 23.8

current_median_commercial_age_10_race_gender_salaried <- commercial_salaried %>% group_by(age_group_10, race_ethnicity, gender)
current_median_commercial_age_10_race_gender_salaried <- current_median_commercial_age_10_race_gender_salaried
  %>% summarise(count = length(current_base_pay),
                median = median(current_base_pay, na.rm = FALSE))

## A tibble: 9 x 5
## Groups: age_group_10, race_ethnicity [6]

## # A tibble: 9 x 5
## # Groups: age_group_10, race_ethnicity [6]
## age_group_10 race_ethnicity gender count median
## <fct> <chr> <chr> <int> <dbl>
## 1 <25 White (United States of America) Female 7 62000
## 2 25-34 Asian (United States of America) Female 5 90000
## 3 25-34 White (United States of America) Female 30 78692.
## 4 25-34 White (United States of America) Male 10 96348.
## 5 35-44 White (United States of America) Female 15 148730.
## 6 45-54 White (United States of America) Female 10 98281.
current_median_commercial_age_10_race_gender_hourly <- commercial_hourly %>% group_by(age_group_10, race_ethnicity, gender)  
  count = length(current_base_pay),  
  median = median(current_base_pay, na.rm = FALSE)  
)  
suppress(current_median_commercial_age_10_race_gender_hourly)

current_median_commercial_age_5_race_group_gender_salaried <- commercial_salaried %>% group_by(age_group_5, race_grouping, gender)  
  count = length(current_base_pay),  
  median = median(current_base_pay, na.rm = FALSE)  
)  
suppress(current_median_commercial_age_5_race_group_gender_salaried)

current_median_commercial_age_5_race_group_gender_hourly <- commercial_hourly %>% group_by(age_group_5, race_grouping, gender)  
  count = length(current_base_pay),  
  median = median(current_base_pay, na.rm = FALSE)  
)  
suppress(current_median_commercial_age_5_race_group_gender_hourly)
## # Groups: age_group_5, race_grouping [10]
## age_group_5 race_grouping gender count median
## <fct> <chr> <chr> <int> <dbl>
## 1 <25 person of color Male 5 22.4
## 2 25-29 person of color Female 7 26.3
## 3 25-29 white Female 7 35.0
## 4 30-34 person of color Female 5 30.4
## 5 40-44 person of color Female 10 29.5
## 6 45-49 person of color Male 10 22.4
## 7 50-54 person of color Female 6 23.3
## 8 50-54 person of color Male 7 21.1
## 9 50-54 white Male 5 24.4
## 10 55-59 person of color Female 7 28.6
## 11 60-64 person of color Female 5 24.3
## 12 60-64 person of color Male 6 23.8

current_median_commercial_age_10_race_group_gender_salaried <- commercial_salaried %>%
group_by(age_group_10, race_grouping, gender)
current_median_commercial_age_10_race_group_gender_salaried <- current_median_commercial_age_10_race_group_gender_salaried
  %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))
suppress(current_median_commercial_age_10_race_group_gender_salaried)

## A tibble: 10 x 5
## Groups: age_group_10, race_grouping [7]
## age_group_10 race_grouping gender count median
## <fct> <chr> <chr> <int> <dbl>
## 1 <25 white Female 7 62000
## 2 25-34 person of color Female 7 85000
## 3 25-34 white Female 30 78692.
## 4 25-34 white Male 10 96348.
## 5 35-44 person of color Male 6 81977.
## 6 35-44 white Female 15 148730.
## 7 45-54 white Female 10 98281.
## 8 45-54 white Male 6 86196.
## 9 55-64 white Female 5 96780
## 10 55-64 white Male 9 97514.

current_median_commercial_age_10_race_group_gender_hourly <- commercial_hourly %>%
group_by(age_group_10, race_grouping)
current_median_commercial_age_10_race_group_gender_hourly <- current_median_commercial_age_10_race_group_gender_hourly
  %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))
suppress(current_median_commercial_age_10_race_group_gender_hourly)

## A tibble: 11 x 5
## Groups: age_group_10, race_grouping [7]
## age_group_10 race_grouping gender count median
## <fct> <chr> <chr> <int> <dbl>
## 1 <25 person of color Male 5 22.4
## 2 25-34 person of color Female 12 27.4
## 3 25-34 person of color Male 6 26.2
## 4 25-34 white Female 8 33.4
## 5 35-44 person of color Female 13 29.7
## 6 35-44 person of color Male 7 23.1
<table>
<thead>
<tr>
<th>#</th>
<th>Age</th>
<th>Gender</th>
<th>Count</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>45-54</td>
<td>person of color Female</td>
<td>10</td>
<td>23.7</td>
</tr>
<tr>
<td>8</td>
<td>45-54</td>
<td>person of color Male</td>
<td>17</td>
<td>22.3</td>
</tr>
<tr>
<td>9</td>
<td>45-54</td>
<td>white Male</td>
<td>5</td>
<td>24.4</td>
</tr>
<tr>
<td>10</td>
<td>55-64</td>
<td>person of color Female</td>
<td>12</td>
<td>25.0</td>
</tr>
<tr>
<td>11</td>
<td>55-64</td>
<td>person of color Male</td>
<td>10</td>
<td>23.9</td>
</tr>
</tbody>
</table>

**Departments**

```r
current_commercial_median_department_salaried <- commercial_salaried converged group_by(department)
current_commercial_median_department_salaried <- current_commercial_median_department_salaried converged summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))
suppress_median(current_commercial_median_department_salaried)
```

```r
current_commercial_median_department_hourly <- commercial_hourly converged group_by(department)
current_commercial_median_department_hourly <- current_commercial_median_department_hourly converged summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))
suppress_median(current_commercial_median_department_hourly)
```

```r
current_commercial_median_department_gender_salaried <- commercial_salaried converged group_by(department, gender)
current_commercial_median_department_gender_salaried <- current_commercial_median_department_gender_salaried converged summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))
suppress_median(current_commercial_median_department_gender_salaried)
```

```r
current_commercial_median_department_gender_salaried <- commercial_salaried converged group_by(department, gender)
current_commercial_median_department_gender_salaried <- current_commercial_median_department_gender_salaried converged summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))
suppress_median(current_commercial_median_department_gender_salaried)
```
current_commercial_median_department_gender_hourly <- commercial_hourly %>%
group_by(department, gender) %>%
count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_commercial_median_department_gender_hourly)

## # A tibble: 7 x 4
## # Groups: department [4]
##    department gender count median
##   <chr>      <chr> <int> <dbl>
## 1 Public Relations Female  5   35
## 2 Client Solutions Male  24  30.1
## 3 Finance Female  17  29.2
## 4 Finance Male 6  28.8
## 5 Client Solutions Female 38  28.8
## 6 Circulation Female 9  23.2
## 7 Circulation Male 40  22.4

current_commercial_median_department_race_salaried <- commercial_salaried %>%
group_by(department, race_ethnicity) %>%
count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_commercial_median_department_race_salaried)

## # A tibble: 5 x 4
## # Groups: department [3]
##    department race_ethnicity count median
##   <chr>          <chr>     <int> <dbl>
## 1 Client Solutions White (United States of America) 79 90000
## 2 WP News Media Serv- White (United States of America) 8 88302.
## 3 Client Solutions Black or African American (United States of America) 10 83805.
## 4 Marketing White (United States of America) 5 83280
## 5 Client Solutions Asian (United States of America) 9 76139.

current_commercial_median_department_race_hourly <- commercial_hourly %>%
group_by(department, race_ethnicity) %>%
count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_commercial_median_department_race_hourly)

## # A tibble: 8 x 4
## # Groups: department [3]
##    department race_ethnicity count median
##   <chr>          <chr>     <int> <dbl>
## 1 Client Solutions White (United States of America) 24 31.0
## 2 Finance White (United States of America) 5 29.5
## 3 Finance Black or African American (United States of America) 16 29.1
## 4 Client Solutions Hispanic or Latino (United States of America) 6 28.5
## 5 Client Solutions Black or African American (United States of America) 25 27.0
## 6 Client Solutions Asian (United States of America) 5 26.3
## 7 Circulation White (United States of America) 8 22.8
## 8 Circulation Black or African American (United States of America) 35 22.4
```r
current_commercial_median_department_race_gender_salaried <-
  commercial_salaried %>%
group_by(
  department, race_ethnicity, gender)

  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )

  suppress_median(current_commercial_median_department_race_gender_salaried)

## # A tibble: 4 x 5
## # Groups: department, race_ethnicity [3]
##   department                    race_ethnicity               gender count median
##          <chr>                  <chr>                      <chr> <int> <dbl>
## 1 Client Solutions               White (United States of America) Male 22 98894.8
## 2 Client Solutions               Black or African American (United States) Female 6 921579
## 3 Client Solutions               White (United States of America) Female 57 86613
## 4 Client Solutions               Asian (United States of America) Female 5 80000

current_commercial_median_department_race_gender_hourly <-
  commercial_hourly %>%
group_by(
  department, race_ethnicity, gender)

  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )

  suppress_median(current_commercial_median_department_race_gender_hourly)

## # A tibble: 9 x 5
## # Groups: department, race_ethnicity [6]
##   department                    race_ethnicity               gender count median
##          <chr>                  <chr>                      <chr> <int> <dbl>
## 1 Client Solutions               White (United States of America) Female 13 31.7
## 2 Client Solutions               White (United States of America) Male 11 30.8
## 3 Finance                        Black or African American (United States of America) Female 12 29.1
## 4 Client Solutions               Hispanic or Latino (United States of America) Female 6 28.5
## 5 Client Solutions               Black or African American (United States of America) Male 9 28.2
## 6 Client Solutions               Black or African American (United States of America) Female 16 26.0
## 7 Circulation                    Black or African American (United States of America) Female 9 23.2
## 8 Circulation                    White (United States of America) Male 8 22.8
## 9 Circulation                    Black or African American (United States of America) Male 26 22.4

current_commercial_median_department_race_group_gender_salaried <-
  commercial_salaried %>%
group_by(
  department, race_grouping, gender)

  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )

  suppress_median(current_commercial_median_department_race_group_gender_salaried)

## # A tibble: 4 x 5
## # Groups: department, race_grouping [2]
##   department          race_grouping gender count median
##          <chr>           <chr>     <chr> <int> <dbl>
## 1 Client Solutions white Male 22 98894.8
## 2 Client Solutions white Female 57 86613
## 3 Client Solutions person of color Female 13 80000
## 4 Client Solutions person of color Male 9 76139.

current_commercial_median_department_race_group_gender_hourly <-
  commercial_hourly %>%
group_by(
  department, race_grouping, gender)

  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
  )

  suppress_median(current_commercial_median_department_race_group_gender_hourly)
```

147
count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_commercial_median_department_race_group_gender_hourly)

## # A tibble: 8 x 5
## # Groups: department, race_grouping [5]
## department race_grouping gender count median
## <chr> <chr> <chr> <int> <dbl>
## 1 Client Solutions white Female 13 31.7
## 2 Client Solutions white Male 11 30.8
## 3 Finance person of color Female 13 28.9
## 4 Client Solutions person of color Male 13 27.0
## 5 Client Solutions person of color Female 25 26.3
## 6 Circulation person of color Female 9 23.2
## 7 Circulation white Male 8 22.8
## 8 Circulation person of color Male 30 22.4

current_commercial_median_department_race_gender_age5_salaried <- commercial_salaried %>%
group_by(department, race_ethnicity, gender, age_group_5)
count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_commercial_median_department_race_gender_age5_salaried)

## # A tibble: 6 x 6
## # Groups: department, race_ethnicity, gender [2]
## department race_ethnicity gender age_group_5 count median
## <chr> <chr> <chr> <fct> <int> <dbl>
## 1 Client Solutions White (United States of America) Female 35-39 9 1.49e5
## 2 Client Solutions White (United States of America) Female 40-44 6 1.27e5
## 3 Client Solutions White (United States of America) Female 50-54 5 1.06e5
## 4 Client Solutions White (United States of America) Male 30-34 5 1.00e5
## 5 Client Solutions White (United States of America) Female 25-29 23 7.50e4
## 6 Client Solutions White (United States of America) Female <25 6 6.10e4

current_commercial_median_department_race_gender_age5_hourly <- commercial_hourly %>%
group_by(department, race_ethnicity, gender, age_group_5)
count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_commercial_median_department_race_gender_age5_hourly)

## # A tibble: 3 x 6
## # Groups: department, race_ethnicity, gender [2]
## department race_ethnicity gender age_group_5 count median
## <chr> <chr> <chr> <fct> <int> <dbl>
## 1 Client Solutions White (United States of America) Female 25-29 5 31.8
## 2 Circulation Black or African American ~ Male 60-64 6 23.8
## 3 Circulation Black or African American ~ Male 45-49 7 21.5

current_commercial_median_department_race_group_gender_age5_salaried <- commercial_salaried %>%
group_by(department, race_grouping, gender, age_group_5)
count = length(current_base_pay),
median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_commercial_median_department_race_group_gender_age5_salaried)

## # A tibble: 6 x 6
## # Groups: department, race_grouping, gender [2]
## #   department race_grouping gender age_group_5 count median
## # <chr>       <chr>      <chr>   <fct>  <int> <dbl>
## 1 Client Solutions white Female 35-39     9  149101
## 2 Client Solutions white Female 40-44     6  126865.
## 3 Client Solutions white Female 50-54     5  105893
## 4 Client Solutions white Male  30-34     5   100000
## 5 Client Solutions white Female 25-29   23   75000
## 6 Client Solutions white Female <25      6   61000

current_commercial_median_department_race_group_gender_age5_hourly <- commercial_hourly %>%
group_by(department, race_grouping, gender, age_group_5)
current_commercial_median_department_race_group_gender_age5_hourly <- current_commercial_median_department_race_group_gender_age5_hourly %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))

## # A tibble: 5 x 6
## # Groups: department, race_grouping, gender [3]
## #   department race_grouping gender age_group_5 count median
## # <chr>       <chr>      <chr>   <fct>  <int> <dbl>
## 1 Client Solutions white Female 25-29     5  31.8
## 2 Client Solutions person of color Female 40-44  5  25.0
## 3 Circulation person of color Male  60-64     6  23.8
## 4 Circulation person of color Male  45-49     7  21.5
## 5 Circulation person of color Male  50-54     5  20.8

Job profiles

current_commercial_median_job_salaried <- commercial_salaried %>%
  group_by(job_profile_current)
current_commercial_median_job_salaried <- current_commercial_median_job_salaried %>%
  summarise(count = length(current_base_pay),
             median = median(current_base_pay, na.rm = FALSE))

## # A tibble: 10 x 3
## job_profile_current count median
## <chr>               <int> <dbl>
## 1 450220 - Sales Representative 25 153987.
## 2 350227 - Custom Content Writer  7 100000
## 3 551104 - Senior Financial Accountant  5  90566
## 4 450120 - Account Manager     26  88645.
## 5 390110 - Multiplatform Editor  9  86105.
## 6 280228 - Artist            7  85000
## 7 340227 - Designer           5  75035
## 8 481205 - Digital Analyst    5  75000
## 9 660127 - Make-Up Person     5  71665.
##10 231303 - Client Service Manager 15  67096.
current_commercial_median_job_hourly <- commercial_hourly %>% group_by(job_profile_current) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))

suppress_median(current_commercial_median_job_hourly)

## # A tibble: 5 x 3
## # Groups: job_profile_current [4]
## job_profile_current  count median
## <chr>       <int> <dbl>
## 1 341027 - Desktop Publisher       6 30.8
## 2 574504 - Senior Accounting Specialist 11 30.4
## 3 565005 - Accounting Specialist   12 26.6
## 4 470121 - Account Executive      16 25.2
## 5 600318 - Circulation Driver (Class A) 35 22.4

current_commercial_median_job_gender_salaried <- commercial_salaried %>% group_by(job_profile_current, gender) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))

suppress_median(current_commercial_median_job_gender_salaried)

## # A tibble: 6 x 4
## # Groups: job_profile_current [4]
## job_profile_current gender count median
## <chr>     <chr>  <int> <dbl>
## 1 450220 - Sales Representative Male    6 162339.
## 2 450220 - Sales Representative Female 19 150780
## 3 450120 - Account Manager Female      17 90110
## 4 390110 - Multiplatform Editor Male   5  85900.
## 5 450120 - Account Manager Male        9  85418.
## 6 231303 - Client Service Manager Female 13  68000

current_commercial_median_job_gender_hourly <- commercial_hourly %>% group_by(job_profile_current, gender) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))

suppress_median(current_commercial_median_job_gender_hourly)

## # A tibble: 5 x 4
## # Groups: job_profile_current [4]
## job_profile_current gender count median
## <chr>     <chr>  <int> <dbl>
## 1 574504 - Senior Accounting Specialist Female 10 30.1
## 2 565005 - Accounting Specialist Male      5 27.2
## 3 565005 - Accounting Specialist Female   7 26.0
## 4 470121 - Account Executive Female       15 25.0
## 5 600318 - Circulation Driver (Class A) Male 34 22.5

current_commercial_median_job_race_salaried <- commercial_salaried %>% group_by(job_profile_current, race_ethnicity) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))

suppress_median(current_commercial_median_job_race_salaried)
suppress_median(current_commercial_median_job_race_salaried)

## # A tibble: 6 x 4
## # Groups: job_profile_current [5]
## job_profile_current race_ethnicity count median
## <chr> <chr> <int> <dbl>
## 1 450220 - Sales Representative White (United States of America) 23 1.51e5
## 2 350227 - Custom Content Manager White (United States of America) 6 1.00e5
## 3 450120 - Account Manager White (United States of America) 15 9.07e4
## 4 390110 - Multiplatform Engineer White (United States of America) 8 8.83e4
## 5 450120 - Account Manager Black or African American (United States of America) 7 8.54e4
## 6 231303 - Client Service Representative White (United States of America) 14 6.55e4

current_commercial_median_job_race_hourly <- commercial_hourly %>%
group_by(job_profile_current, race_ethnicity)
current_commercial_median_job_race_hourly <- current_commercial_median_job_race_hourly %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_commercial_median_job_race_hourly)

## # A tibble: 6 x 4
## # Groups: job_profile_current [4]
## job_profile_current race_ethnicity count median
## <chr> <chr> <int> <dbl>
## 1 574504 - Senior Accounting Manager Black or African American (United States of America) 8 30.1
## 2 565005 - Accounting Specialist Black or African American (United States of America) 7 26.0
## 3 470121 - Account Executive White (United States of America) 5 25.4
## 4 470121 - Account Executive Black or African American (United States of America) 9 24.7
## 5 600318 - Circulation Drive Manager White (United States of America) 7 23.0
## 6 600318 - Circulation Drive Manager Black or African American (United States of America) 23 22.4

current_commercial_median_job_race_gender_salaried <- commercial_salaried %>%
group_by(job_profile_current, race_ethnicity, gender)
current_commercial_median_job_race_gender_salaried <- current_commercial_median_job_race_gender_salaried %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_commercial_median_job_race_gender_salaried)

## # A tibble: 4 x 5
## # Groups: job_profile_current, race_ethnicity [3]
## job_profile_current race_ethnicity gender count median
## <chr> <chr> <chr> <int> <dbl>
## 1 450220 - Sales Representative White (United States of America) Male 5 1.55e5
## 2 450220 - Sales Representative White (United States of America) Female 18 1.50e5
## 3 450120 - Account Manager White (United States of America) Female 11 9.01e4
## 4 231303 - Client Service Manager White (United States of America) Female 12 6.60e4

current_commercial_median_job_race_gender_hourly <- commercial_hourly %>%
group_by(job_profile_current, race_ethnicity, gender)
current_commercial_median_job_race_gender_hourly <- current_commercial_median_job_race_gender_hourly %>%
  summarise(
    count = length(current_base_pay),
    median = median(current_base_pay, na.rm = FALSE)
)
suppress_median(current_commercial_median_job_race_gender_hourly)

## # A tibble: 5 x 5

151
```r
# Groups: job_profile_current, race_ethnicity [5]
# job_profile_current  race_ethnicity  gender count  median
# <chr> <chr> <chr> <int> <dbl>
# 1 574504 - Senior Accounting Specialist  Black or African American  Female 7 29.7
# 2 565005 - Accounting Specialist  Black or African American  Female 5 26.0
# 3 470121 - Account Executive  Black or African American  Female 9 24.7
# 4 600318 - Circulation Driver (Class A)  White (United States of America)  Male 7 23.0
# 5 600318 - Circulation Driver (Class A)  Black or African American  Male 22 22.4

current_commercial_median_job_race_group_gender_salaried <- commercial_salaried %>%
group_by(job_profile_current, race_grouping, gender)
summarise(count = length(current_base_pay),
          median = median(current_base_pay, na.rm = FALSE))
suppress_median(current_commercial_median_job_race_group_gender_salaried)
```
```r
# 2 231303 - Client Serv~ White (United Stat~ Female 25-29 8 6.62e4

current_commercial_median_job_race_gender_age5_hourly <- commercial_hourly %>% group_by(job_profile_current, race_ethnicity, gender, age_group_5) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))
suppress_median(current_commercial_median_job_race_gender_age5_hourly)
```

```r
## # A tibble: 2 x 6
## # Groups: job_profile_current, race_ethnicity, gender [1]
## #  job_profile_current race_ethnicity gender age_group_5 count median
## # <chr> <chr> <chr> <fct> <int> <dbl>
## 1 600318 - Circulatio~ Black or African Am~ Male 60-64 6 23.8
## 2 600318 - Circulatio~ Black or African Am~ Male 45-49 7 21.5
```

```r
current_commercial_median_job_race_group_gender_age5_salaried <- commercial_salaried %>% group_by(job_profile_current, race_grouping, gender, age_group_5) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))
suppress_median(current_commercial_median_job_race_group_gender_age5_salaried)
```

```r
## # A tibble: 2 x 6
## # Groups: job_profile_current, race_grouping, gender [2]
## #  job_profile_current race_grouping gender age_group_5 count median
## # <chr> <chr> <chr> <fct> <int> <dbl>
## 1 450220 - Sales Representat~ white Female 35-39 8 1.50e5
## 2 231303 - Client Service Ma~ white Female 25-29 8 6.62e4
```

```r
current_commercial_median_job_race_group_gender_age5_hourly <- commercial_hourly %>% group_by(job_profile_current, race_grouping, gender, age_group_5) %>% summarise(count = length(current_base_pay), median = median(current_base_pay, na.rm = FALSE))
suppress_median(current_commercial_median_job_race_group_gender_age5_hourly)
```

```r
## # A tibble: 2 x 6
## # Groups: job_profile_current, race_grouping, gender [1]
## #  job_profile_current race_grouping gender age_group_5 count median
## # <chr> <chr> <chr> <fct> <int> <dbl>
## 1 600318 - Circulation Drive~ person of co~ Male 60-64 6 23.8
## 2 600318 - Circulation Drive~ person of co~ Male 45-49 7 21.5
```

Performance evaluations

```r
commercial_ratings <- filter(ratings_combined, dept == 'Commercial')
commercial_ratings_gender <- commercial_ratings %>% group_by(gender) %>% summarise(count = length(performance_rating), median = median(performance_rating, na.rm = TRUE))
suppress_median(commercial_ratings_gender)
```

```r
## # A tibble: 2 x 3
```
## gender count median
## <chr> <int> <dbl>
## 1 Female 1308 3.3
## 2 Male 984 3.2

```r
commercial_ratings_race <- commercial_ratings %>% group_by(race_ethnicity) %>% summarise(
  count = length(performance_rating),
  median = median(performance_rating, na.rm = TRUE)
)
suppress_median(commercial_ratings_race)
```

## A tibble: 6 x 3
## race_ethnicity count median
## <chr> <int> <dbl>
## 1 Asian (United States of America) 168 3.3
## 2 Two or More Races (United States of America) 36 3.3
## 3 White (United States of America) 1096 3.3
## 4 Black or African American (United States of America) 860 3.2
## 5 Hispanic or Latino (United States of America) 96 3.15
## 6 Prefer Not to Disclose (United States of America) 28 3

```r
commercial_ratings_race_gender <- commercial_ratings %>% group_by(race_ethnicity, gender) %>% summarise(
  count = length(performance_rating),
  median = median(performance_rating, na.rm = TRUE)
)
suppress(commercial_ratings_race_gender)
```

## A tibble: 12 x 4
## # Groups: race_ethnicity [6]
## race_ethnicity gender count median
## <chr> <chr> <int> <dbl>
## 1 Asian (United States of America) Female 116 3.3
## 2 Asian (United States of America) Male 52 3.1
## 3 Black or African American (United States of America) Female 408 3.2
## 4 Black or African American (United States of America) Male 452 3.05
## 5 Hispanic or Latino (United States of America) Female 56 3.15
## 6 Hispanic or Latino (United States of America) Male 40 3.1
## 7 Prefer Not to Disclose (United States of America) Female 16 3
## 8 Prefer Not to Disclose (United States of America) Male 12 NA
## 9 Two or More Races (United States of America) Female 20 3.3
## 10 Two or More Races (United States of America) Male 16 3.35
## 11 White (United States of America) Female 684 3.3
## 12 White (United States of America) Male 412 3.3

```r
commercial_ratings_race_gender_under3 <- filter(commercial_ratings, performance_rating < 3.1) %>% group_by(race_grouping, gender) %>% summarise(
  count = length(performance_rating),
  median = median(performance_rating, na.rm = TRUE)
)
suppress(commercial_ratings_race_gender_under3)
```

## A tibble: 4 x 4
## # Groups: race_grouping [2]
## race_grouping gender count median
## <chr> <chr> <int> <dbl>
commercial_ratings_race_gender_over4 <- filter(commercial_ratings, performance_rating > 3.9) %>% group_by(race_grouping, gender) %>% summarise(count = length(performance_rating), median = median(performance_rating, na.rm = TRUE))

Pay changes

commercial_change <- filter(reason_for_change_combined, dept == 'Commercial')

commercial_change_gender <- commercial_change %>% group_by(business_process_reason, gender) %>% summarise(count = length(business_process_reason))

commercial_change_race <- commercial_change %>% group_by(business_process_reason, race_ethnicity) %>% summarise(count = length(business_process_reason))
## 1 <NA> White (United States of America) 2995
## 2 <NA> Black or African American (United States of America) 2340
## 3 <NA> Asian (United States of America) 448
## 4 Request Compensation Change > Adjustment White (United States of America) 392
## 5 Request Compensation Change > Adjustment Black or African American (United States of America) 339
## 6 <NA> Hispanic or Latino (United States of America) 272
## 7 Merit > Performance > Annual Performance Black or African American (United States of America) 239
## 8 Merit > Performance > Annual Performance White (United States of America) 220
## 9 Request Compensation Change > Adjustment White (United States of America) 179
## 10 Transfer > Transfer > Move to another Black or African American (United States of America) 116
## ... with 41 more rows

```
commercial_change_race_gender <- commercial_change %>%
group_by(business_process_reason, race_ethnicity, gender)
commercial_change_race_gender <- commercial_change_race_gender %>%
summarise(count = length(business_process_reason))
suppress_count(commercial_change_race_gender)
```

## # A tibble: 77 x 4
## # Groups: business_process_reason, race_ethnicity [45]
## # business_process_reason race_ethnicity gender  count
## # <chr> <chr> <chr> <int>
## 1 <NA> White (United States of America) Female 1839
## 2 <NA> Black or African American (United States of America) Male 1272
## 3 <NA> White (United States of America) Male 1156
## 4 <NA> Black or African American (United States of America) Female 1068
## 5 <NA> Asian (United States of America) Female 320
## 6 Request Compensation Change > Adjustment White (United States of America) Female 236
## 7 Request Compensation Change > Adjustment Black or African American (United States of America) Female 179
## 8 <NA> Hispanic or Latino (United States of America) Female 164
## 9 Request Compensation Change > Adjustment Black or African American (United States of America) Male 160
## 10 Request Compensation Change > Adjustment White (United States of America) Male 166
## ... with 67 more rows

Performance evaluations x merit raises

```
reason_for_change_combined <- reason_for_change_combined %>%
mutate(merit_raises = grepl("*Merit*", business_process_reason))
twenty14 = as.Date('2016-04-01')
twenty15 = as.Date('2017-04-01')
twenty16 = as.Date('2018-04-01')
twenty17 = as.Date('2019-04-01')
twenty18 = as.Date('2020-04-01')

reason_for_change_combined <- reason_for_change_combined %>%
mutate(raise_after=case_when(
effective_date < twenty14 ~ 'before 2015',
effective_date < twenty15 ~ '2015',
effective_date < twenty16 ~ '2016',
effective_date < twenty17 ~ '2017',
effective_date < twenty18 ~ '2018',
TRUE ~ 'Other'))
merit Raises Commercial Gender Salaried <- filter(reason_for_change_combined, merit_raises == TRUE, dept = 'Commercial', pay_rate_type = 'Salaried')
merit_raises_commercial_gender_salaried <- merit_raises_commercial_gender_salaried %>%
summarise(
156
```
```r
count = length(base_pay_change),
median = median(base_pay_change, na.rm = TRUE)
)
suppress(merit_raises_commercial_gender_salaried)

## # A tibble: 2 x 3
##   gender count median
##   <chr> <int>  <dbl>
## 1 Female  97   1317.
## 2 Male    74   1205.

merit_raises_commercial_gender_hourly <- filter(reason_for_change_combined, merit_raises == TRUE, dept == Commercial)
merit_raises_commercial_gender_hourly <- merit_raises_commercial_gender_hourly %>% summarise(
  count = length(base_pay_change),
  median = median(base_pay_change, na.rm = TRUE)
)
suppress(merit_raises_commercial_gender_hourly)

## # A tibble: 2 x 3
##   gender count median
##   <chr> <int>  <dbl>
## 1 Female 170   0.425
## 2 Male   138   0.33

merit_raises_commercial_race_salaried <- filter(reason_for_change_combined, merit_raises == TRUE, dept == Commercial)
merit_raises_commercial_race_salaried <- merit_raises_commercial_race_salaried %>% summarise(
  count = length(base_pay_change),
  median = median(base_pay_change, na.rm = TRUE)
)
suppress_median(merit_raises_commercial_race_salaried)

## # A tibble: 4 x 3
## race_ethnicity count median
## <chr>          <int>  <dbl>
## 1 Asian (United States of America)  23 1375
## 2 Hispanic or Latino (United States of America)  6 1322.
## 3 White (United States of America)       110 1287.
## 4 Black or African American (United States of America) 30 1117.

merit_raises_commercial_race_hourly <- filter(reason_for_change_combined, merit_raises == TRUE, dept == Commercial)
merit_raises_commercial_race_hourly <- merit_raises_commercial_race_hourly %>% summarise(
  count = length(base_pay_change),
  median = median(base_pay_change, na.rm = TRUE)
)
suppress_median(merit_raises_commercial_race_hourly)

## # A tibble: 4 x 3
## race_ethnicity count median
## <chr>          <int>  <dbl>
## 1 Asian (United States of America)  11 0.45
## 2 White (United States of America)   85 0.42
## 3 Hispanic or Latino (United States of America) 11 0.37
## 4 Black or African American (United States of America) 197 0.35

merit_raises_commercial_race_group_salaried <- filter(reason_for_change_combined, merit_raises == TRUE, dept == Commercial)
merit_raises_commercial_race_group_salaried <- merit_raises_commercial_race_group_salaried %>% summarise(
  count = length(base_pay_change),
  median = median(base_pay_change, na.rm = TRUE)
)
suppress_median(merit_raises_commercial_race_group_salaried)
```

157
count = length(base_pay_change),
median = median(base_pay_change, na.rm = TRUE)
)
suppress_median(merit_raises_commercial_race_group_salaried)

## # A tibble: 2 x 3
## race_grouping  count  median
## <chr>       <int>  <dbl>
## 1 white        110  1287.
## 2 person of color  60  1225

merit_raises_commercial_race_group_hourly <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'Commercial', pay_rate_type == 'Hourly')
merit_raises_commercial_race_group_hourly <- merit_raises_commercial_race_group_hourly %>% summarise(
  count = length(base_pay_change),
  median = median(base_pay_change, na.rm = TRUE)
)
suppress_median(merit_raises_commercial_race_group_hourly)

## # A tibble: 2 x 3
## race_grouping  count  median
## <chr>       <int>  <dbl>
## 1 white        85   0.42
## 2 person of color  223  0.35

merit_raises_commercial_gender_race_group_salaried <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'Commercial', pay_rate_type == 'Salaried')
merit_raises_commercial_gender_race_group_salaried <- merit_raises_commercial_gender_race_group_salaried %>% summarise(
  count = length(base_pay_change),
  median = median(base_pay_change, na.rm = TRUE)
)
suppress_median(merit_raises_commercial_gender_race_group_salaried)

## # A tibble: 4 x 4
## # Groups: race_grouping [2]
## race_grouping gender  count  median
## <chr>        <chr>  <int>  <dbl>
## 1 white       Female     69  1317.
## 2 person of color Female 27  1305
## 3 white       Male       41  1282.
## 4 person of color Male   33  1134.

merit_raises_commercial_gender_race_group_hourly <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'Commercial', pay_rate_type == 'Hourly')
merit_raises_commercial_gender_race_group_hourly <- merit_raises_commercial_gender_race_group_hourly %>% summarise(
  count = length(base_pay_change),
  median = median(base_pay_change, na.rm = TRUE)
)
suppress_median(merit_raises_commercial_gender_race_group_hourly)

## # A tibble: 4 x 4
## # Groups: race_grouping [2]
## race_grouping gender  count  median
## <chr>        <chr>  <int>  <dbl>
## 1 white       Female     44  0.515
## 2 person of color Female 126  0.375
## 3 white       Male       41  0.35
## 4 person of color Male   97  0.32
fifteen_raises_amount <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'Commercial', count = length(base_pay_change), median_raise = median(base_pay_change, na.rm = TRUE))
suppress(fifteen_raises_amount)

## # A tibble: 2 x 4
## # Groups: race_grouping [1]
## race_grouping gender count median_raise
## <chr> <chr> <int> <dbl>
## 1 white Female 7 937.
## 2 white Male 5 851.

fifteen_raises_score <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'Commercial', count = length('2015_annual_performance_rating'), median = median('2015_annual_performance_rating', na.rm = TRUE))
suppress(fifteen_raises_score)

## # A tibble: 0 x 4
## # Groups: race_grouping [0]
## # ... with 4 variables: race_grouping <chr>, gender <chr>, count <int>, median <chr>

sixteen_raises_amount <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'Commercial', count = length(base_pay_change), median_raise = median(base_pay_change, na.rm = TRUE))
suppress(sixteen_raises_amount)

## # A tibble: 4 x 4
## # Groups: race_grouping [2]
## race_grouping gender count median_raise
## <chr> <chr> <int> <dbl>
## 1 person of color Female 5 1729.
## 2 person of color Male 6 1507.
## 3 white Female 9 1683
## 4 white Male 7 1291.

sixteen_raises_score <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'Commercial', count = length('2016_annual_performance_rating'), median = median('2016_annual_performance_rating', na.rm = TRUE))
suppress(sixteen_raises_score)

## # A tibble: 0 x 4
## # Groups: race_grouping [0]
## # ... with 4 variables: race_grouping <chr>, gender <chr>, count <int>, median <chr>

seventeen_raises_amount <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'Commercial', count = length(base_pay_change), median_raise = median(base_pay_change, na.rm = TRUE))
suppress(seventeen_raises_amount)

## # A tibble: 4 x 4
## # Groups: race_grouping [2]
## race_grouping gender count median_raise
## <chr> <chr> <int> <dbl>
## 1 person of color Female 5 1729.
## 2 person of color Male 6 1507.
## 3 white Female 9 1683
## 4 white Male 7 1291.
count = length(base_pay_change),
median_raise = median(base_pay_change, na.rm = TRUE)
)
suppress(seventeen_raises_amount)
## # A tibble: 3 x 4
## # Groups: race_grouping [2]
## race_grouping gender count median_raise
## <chr> <chr> <int> <dbl>
## 1 person of color Male 8 1000
## 2 white Female 13 1398.
## 3 white Male 5 1415.

seventeen_raises_score <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'Commercial'

seventeen_raises_score <- seventeen_raises_score %>%
summarise(
count = length('2017_annual_performance_rating'),
median = median('2017_annual_performance_rating', na.rm = TRUE)
)
suppress(seventeen_raises_score)
## # A tibble: 0 x 4
## # Groups: race_grouping [0]
## # ... with 4 variables: race_grouping <chr>, gender <chr>, count <int>,
## # median <chr>

eighteen_raises_amount <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'Commercial'
eighteen_raises_amount <- eighteen_raises_amount %>%
summarise(
count = length(base_pay_change),
median_raise = median(base_pay_change, na.rm = TRUE)
)
suppress(eighteen_raises_amount)
## # A tibble: 4 x 4
## # Groups: race_grouping [2]
## race_grouping gender count median_raise
## <chr> <chr> <int> <dbl>
## 1 person of color Female 7 1416.
## 2 person of color Male 7 1050
## 3 white Female 21 1669.
## 4 white Male 8 1417.

eighteen_raises_score <- filter(reason_for_change_combined, merit_raises == TRUE, dept == 'Commercial',
eighteen_raises_score <- eighteen_raises_score %>%
summarise(
count = length('2018_annual_performance_rating'),
median = median('2018_annual_performance_rating', na.rm = TRUE)
)
suppress(eighteen_raises_score)
## # A tibble: 0 x 4
## # Groups: race_grouping [0]
## # ... with 4 variables: race_grouping <chr>, gender <chr>, count <int>,
## # median <chr>

merit_raises_15 <- filter(reason_for_change_combined, raise_after == '2015', merit_raises == TRUE)
merit_raises_16 <- filter(reason_for_change_combined, raise_after == '2016', merit_raises == TRUE)
merit_raises_17 <- filter(reason_for_change_combined, raise_after == '2017', merit_raises == TRUE)
merit_raises_18 <- filter(reason_for_change_combined, raise_after == '2018', merit_raises == TRUE)

merit_raises_15 <- merit_raises_15[,c('base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping')]
merit_raises_16 <- merit_raises_16[,c('base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping')]
merit_raises_17 <- merit_raises_17[,c('base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping')]
merit_raises_18 <- merit_raises_18[,c('base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping')]

names(merit_raises_15) <- c('base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping')
names(merit_raises_16) <- c('base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping')
names(merit_raises_17) <- c('base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping')
names(merit_raises_18) <- c('base_pay_change', 'pay_rate_type', 'gender', 'race_ethnicity', 'race_grouping')

merit_raises_combined <- rbind(merit_raises_15, merit_raises_16, merit_raises_17, merit_raises_18)

commercial_salaried_raises <- filter(merit_raises_combined, pay_rate_type == 'Salaried', dept == 'Commercial')
commercial_salaried_raises <- commercial_salaried_raises %>% summarise(
  count = length(base_pay_change),
  median = median(base_pay_change, na.rm = TRUE)
)
suppress(commercial_salaried_raises)

## A tibble: 4 x 4
## Groups: race_grouping [2]
## race_grouping gender count median
## <chr>       <chr> <int>  <dbl>
## 1 person of color Female 20  1360.  
## 2 person of color Male  24  1096.  
## 3 white Female  50  1344.  
## 4 white Male  25  1291.  

commercial_salaried_raises_scores <- filter(merit_raises_combined, pay_rate_type == 'Salaried', dept == 'Commercial')
commercial_salaried_raises_scores <- commercial_salaried_raises_scores %>% summarise(
  count = length(performance_rating),
  median = median(performance_rating, na.rm = TRUE)
)
suppress(commercial_salaried_raises_scores)

## A tibble: 4 x 4
## Groups: race_grouping [2]
## race_grouping gender count median
## <chr>       <chr> <int>  <dbl>
## 1 person of color Female 20  3.5  
## 2 person of color Male  24  3.3  
## 3 white Female  50  3.4  
## 4 white Male  25  3.4  

commercial_hourly_raises <- filter(merit_raises_combined, pay_rate_type == 'Hourly', dept == 'Commercial')
commercial_hourly_raises <- commercial_hourly_raises %>% summarise(
  count = length(base_pay_change),
  median = median(base_pay_change, na.rm = TRUE)
)
suppress(commercial_hourly_raises)

## A tibble: 4 x 4
## Groups: race_grouping [2]
# race_grouping  gender  count  median
# <chr>       <chr>  <int>  <dbl>
# 1 person of color Female 102 0.37
# 2 person of color Male 89 0.28
# 3 white    Female 34 0.515
# 4 white    Male 37 0.35

commercial_hourly_raises_scores <- filter(merit_raises_combined, pay_rate_type == 'Hourly', dept == 'Commercial')
commercial_hourly_raises_scores <- commercial_hourly_raises_scores %>%
  group_by(race_grouping, gender) %>%
  summarise(count = length(performance_rating),
            median = median(performance_rating, na.rm = TRUE))
suppress(commercial_hourly_raises_scores)

## # A tibble: 4 x 4
## # Groups: race_grouping [2]
## race_grouping gender count median
## <chr>       <chr> <int> <dbl>
## 1 person of color Female 102 3.3
## 2 person of color Male 89 3.2
## 3 white    Female 34 3.4
## 4 white    Male 37 3.2

Regression

commercial_salaried_regression <- fastDummies::dummy_cols(commercial_salaried_regression, select_columns = c('gender', 'race_ethnicity', 'age_group_5', 'years_of_service_grouped', 'dept', 'desk', 'tier', 'race_grouping'))
names(commercial_salaried_regression) <- gsub('_', '-', names(commercial_salaried_regression))
names(commercial_salaried_regression) <- gsub('+', '_over', names(commercial_salaried_regression))
names(commercial_salaried_regression) <- gsub('-', '_under', names(commercial_salaried_regression))
linearMod41 <- lm(formula = current_base_pay ~ gender_Female + gender_Male, data = commercial_salaried_regression)
summary(linearMod41)

## Call:
## lm(formula = current_base_pay ~ gender_Female + gender_Male, 
##     data = commercial_salaried_regression)
## Residuals:
##    Min     1Q Median     3Q    Max
## -42573 -22322  -9445   9259 115917
## Coefficients: (1 not defined because of singularities)
##            Estimate Std. Error t value Pr(>|t|)
## (Intercept) 94863     5051 18.780    <2e-16 ***
## gender_Female 1739     6282  0.277     0.782
## gender_Male NA     NA   NA     NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error: 34630 on 131 degrees of freedom
## Multiple R-squared:  0.0005845,  Adjusted R-squared: -0.007045
## F-statistic: 0.07662 on 1 and 131 DF,  p-value: 0.7824
linearMod42 <- lm(formula = current_base_pay ~ race_grouping_white + race_grouping_person_of_color, data = commercial_salaried_regression)
summary(linearMod42)

## Call:
## lm(formula = current_base_pay ~ race_grouping_white + race_grouping_person_of_color, 
## data = commercial_salaried_regression)
##
## Residuals:
##    Min     1Q   Median     3Q    Max
## -44088 -23088   -8978    9692 111692
##
## Coefficients: (1 not defined because of singularities)
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)          78404.0    24283.3  3.229    0.00157 **
## race_grouping_white  20684.2    24527.3  0.843    0.40059
## race_grouping_person_of_color  9090.6    25030.1  0.363    0.71709
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 34340 on 130 degrees of freedom
## Multiple R-squared: 0.02468, Adjusted R-squared: 0.009673
## F-statistic: 1.645 on 2 and 130 DF,  p-value: 0.1971

linearMod43 <- lm(formula = current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color, data = commercial_salaried_regression)
summary(linearMod43)

## Call:
## lm(formula = current_base_pay ~ gender_Female + gender_Male + 
##     race_grouping_white + race_grouping_person_of_color, data = commercial_salaried_regression)
##
## Residuals:
##    Min     1Q   Median     3Q    Max
## -44357 -23357   -8858    9423 112255
##
## Coefficients: (1 not defined because of singularities)
##                         Estimate Std. Error  t value Pr(>|t|)
## (Intercept)          77571.1    25184.5  3.080    0.00253 **
## gender_Female       832.281    6333.3   0.131    0.89565
## gender_Male          NA        NA     NA     NA
## race_grouping_white 20953.5    24705.1  0.848    0.39793
## race_grouping_person_of_color  9479.6    25300.1  0.375    0.70851
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 34470 on 129 degrees of freedom
## Multiple R-squared: 0.02481, Adjusted R-squared: 0.00213
## F-statistic: 1.094 on 3 and 129 DF,  p-value: 0.3542

linearMod44 <- lm(formula = current_base_pay ~ gender_Female + gender_Male + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = commercial_salaried_regression)
summary(linearMod44)

## Call:
## lm(formula = current_base_pay ~ gender_Female + gender_Male +
##     age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 +
##     age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
##     age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
##     age_group_5_65_over, data = commercial_salaried_regression)
##
## Residuals:
##    Min 1Q Median 3Q Max
##-69838 -19792  -4420 13357 101706
##
## Coefficients: (2 not defined because of singularities)
##   Estimate Std. Error t value Pr(>|t|)
## (Intercept) 111172.9   21450.2   5.183  8.77e-07 ***
## gender_Female  9550.2    5934.5   1.609   0.1101
## gender_Male NA NA NA NA
## age_group_5_under_25 -54303.5   23972.3  -2.265   0.0253 *
## age_group_5_25to29 -41249.7   22596.0  -1.826   0.0704 .
## age_group_5_30to34 -2099.2   22994.9  -0.091   0.9274
## age_group_5_35to39 -814.7   22804.3  -0.036   0.9716
## age_group_5_40to44  5922.0   24293.7   0.244   0.8078
## age_group_5_45to49 -13931.3  22984.0  -0.599   0.5504
## age_group_5_50to54 -24879.8  23861.9  -1.043   0.2992
## age_group_5_55to59 -21494.8  23684.1  -0.908   0.3659
## age_group_5_60to64 -21443.9  23617.1  -0.908   0.3657
## age_group_5_65_over NA NA NA NA
##
## Residual standard error: 30340 on 122 degrees of freedom
## Multiple R-squared: 0.2858, Adjusted R-squared: 0.2272
## F-statistic: 4.882 on 10 and 122 DF, p-value: 6.47e-06

linearMod45 <- lm(formula = current_base_pay ~ race_grouping_white + race_grouping_person_of_color +
##     age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 +
##     age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
##     age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
##     age_group_5_65_over, data = commercial_salaried_regression)
##
## Call:
## lm(formula = current_base_pay ~ race_grouping_white + race_grouping_person_of_color +
##     age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 +
##     age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
##     age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
##     age_group_5_65_over, data = commercial_salaried_regression)
##
## Residuals:
##    Min 1Q Median 3Q Max
##-62296 -18193  -3480 12866 90105
##
## Coefficients: (1 not defined because of singularities)
##   Estimate Std. Error t value Pr(>|t|)
## (Intercept)  70232.3  30123.3  2.331  0.0214 *
## race_grouping_white  49331.5  21787.6  2.264  0.0253 *
## race_grouping_person_of_color  32549.8  22264.8  1.462  0.1464
## age_group_5_under_25 -53376.0  22895.2 -2.331  0.0214 *
## age_group_5_25to29 -38371.2  21443.9 -1.789  0.0761 .
## age_group_5_30to34 1111.7  22157.4  0.050  0.9601

164
```
## age_group_5_35to39  231.2  21945.2  0.011  0.9916
## age_group_5_40to44  15230.7  23150.7  0.658  0.5119
## age_group_5_45to49  -9434.3  22320.8 -0.423  0.6733
## age_group_5_50to54  -24907.3  22895.2 -1.088  0.2788
## age_group_5_55to59  -21754.2  22836.8 -0.953  0.3427
## age_group_5_60to64  -19302.0  22769.9 -0.848  0.3983
## age_group_5_65_over NA NA NA NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 29380 on 121 degrees of freedom
## Multiple R-squared:  0.3353,  Adjusted R-squared:  0.2749
## F-statistic:  5.549 on 11 and 121 DF,  p-value: 3.833e-07

linearMod46 <- lm(formula = current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = commercial_salaried_regression)

##
## Call:
## lm(formula = current_base_pay ~ gender_Female + gender_Male +
##    race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 +
##    age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 +
##    age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 +
##    age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over,
##    data = commercial_salaried_regression)
##
## Residuals:
##   Min     1Q Median     3Q    Max
## -63447  -17726  -2978   12784  95358
##
## Coefficients: (2 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)    67196     29978  2.242  0.0268 *
## gender_Female    9360      5747  1.629  0.1060
## gender_Male NA NA NA NA
## race_grouping_white    51960     21700  2.394  0.0182 *
## race_grouping_person_of_color   35994     22215  1.620  0.1078
## age_group_5_under_25  -60538    23162 -2.614  0.0101 *
## age_group_5_25to29   -45882    21792 -2.105  0.0373 *
## age_group_5_30to34    -3734    22208 -0.168  0.8667
## age_group_5_35to39   -4268    21971 -0.194  0.8463
## age_group_5_40to44     7429    23488  0.316  0.7523
## age_group_5_45to49   -14443    22382 -0.645  0.5200
## age_group_5_50to54   -31133    23059 -1.350  0.1795
## age_group_5_55to59   -26190    22845 -1.146  0.2539
## age_group_5_60to64   -22964    22728 -1.010  0.3143
## age_group_5_65_over NA NA NA NA
##---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 29190 on 120 degrees of freedom
## Multiple R-squared:  0.3497,  Adjusted R-squared:  0.2847
## F-statistic:  5.377 on 12 and 120 DF,  p-value: 3.101e-07
```
linearMod47 <- lm(formula = current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over + years_of_service_grouped_0 + years_of_service_grouped_1to2 + years_of_service_grouped_3to5 + years_of_service_grouped_6to10 + years_of_service_grouped_11to15 + years_of_service_grouped_16to20 + years_of_service_grouped_21to25 + years_of_service_grouped_25_over, data = commercial_salaried_regression)

summary(linearMod47)

## Call:
## lm(formula = current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over + years_of_service_grouped_0 + years_of_service_grouped_1to2 + years_of_service_grouped_3to5 + years_of_service_grouped_6to10 + years_of_service_grouped_11to15 + years_of_service_grouped_16to20 + years_of_service_grouped_21to25 + years_of_service_grouped_25_over, data = commercial_salaried_regression)

## Residuals:
##      Min   1Q Median 3Q    Max
## -60538 -17883  -3429 16197 91640

## Coefficients: (3 not defined because of singularities)
##                               Estimate Std. Error    t value Pr(>|t|)
## (Intercept)                    66064.9   31507.6     2.097  0.0382 *
## gender_Female                  9634.6    6095.5     1.581  0.1168
## gender_Male                    NA        NA         NA     NA
## race_grouping_white            53503.0   21924.2     2.440  0.0162 *
## race_grouping_person_of_color  39591.6   22445.3     1.764  0.0804 .
## age_group_5_under_25           -68107.0   26381.2     2.582  0.0111 *
## age_group_5_25to29             -54633.3   25065.3     2.180  0.0314 *
## age_group_5_30to34             -9711.9    25048.3    -0.388  0.6989
## age_group_5_35to39             -7693.3    24057.3    -0.320  0.7497
## age_group_5_40to44             -228.9     26494.5    -0.009  0.9931
## age_group_5_45to49             -17846.6   25040.8    -0.713  0.4775
## age_group_5_50to54             -29093.0   25702.4    -1.132  0.2601
## age_group_5_55to59             -30069.1   23273.9    -1.292  0.1990
## age_group_5_60to64             -25234.2   26257.9    -0.961  0.3386
## age_group_5_65_over            NA        NA         NA     NA
## years_of_service_grouped_0     5031.0     19054.1     0.264  0.7922
## years_of_service_grouped_1to2  9283.9     18949.1     0.490  0.6251
## years_of_service_grouped_3to5  10317.9     18590.9     0.555  0.5800
## years_of_service_grouped_6to10 -2878.6     18293.6    -0.157  0.8752
## years_of_service_grouped_11to15 -20650.9    21676.3    -0.953  0.3428
## years_of_service_grouped_16to20 -2368.4     22591.2    -0.105  0.9167
## years_of_service_grouped_21to25 -3082.4     22725.9    -0.136  0.8924
## years_of_service_grouped_25_over NA        NA         NA     NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Residual standard error: 29230 on 113 degrees of freedom
## Multiple R-squared:  0.3858, Adjusted R-squared:  0.2825
## F-statistic: 3.735 on 19 and 113 DF,  p-value: 5.587e-06

merit_raises_combined_salaried_regression <- filter(merit_raises_combined, dept == 'Commercial', pay_rate_type == 'Salaried')
merit_raises_combined_salaried_regression <- fastDummies::dummy_cols(merit_raises_combined_salaried_regression)
names(merit_raises_combined_salaried_regression) <- gsub(',', '', names(merit_raises_combined_salaried_regression))
names(merit_raises_combined_salaried_regression) <- gsub('-', 'to', names(merit_raises_combined_salaried_regression))
names(merit_raises_combined_salaried_regression) <- gsub('/\+', '_over', names(merit_raises_combined_salaried_regression))
names(merit_raises_combined_salaried_regression) <- gsub('<', 'under', names(merit_raises_combined_salaried_regression))

linearMod48 <- lm(formula = base_pay_change ~ gender_Female + gender_Male, data = merit_raises_combined_salaried_regression)
summary(linearMod48)

##
## Call:
## lm(formula = base_pay_change ~ gender_Female + gender_Male, data = merit_raises_combined_salaried_regression)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1442.2 -631.0 -253.3  258.0  4270.5
##
## Coefficients: (1 not defined because of singularities)
##            Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1349.6    144.8   9.319  8.22e-16 ***
## gender_Female   307.3    188.3   1.632    0.105
## gender_Male NA NA NA NA
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1014 on 118 degrees of freedom
## Multiple R-squared: 0.02208, Adjusted R-squared: 0.01379
## F-statistic: 2.664 on 1 and 118 DF, p-value: 0.1053

linearMod49 <- lm(formula = base_pay_change ~ race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_salaried_regression)
summary(linearMod49)

##
## Call:
## lm(formula = base_pay_change ~ race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_salaried_regression)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1374.7 -642.8 -256.7  329.2  4338.1
##
## Coefficients:
##            Estimate Std. Error t value Pr(>|t|)
## (Intercept)     1400.00    1026.78   1.363    0.175
## race_grouping_white  189.38    1033.60   0.183    0.855
## race_grouping_person_of_color  35.73    1038.38   0.034    0.973
##
## Residual standard error: 1027 on 117 degrees of freedom
## Multiple R-squared: 0.005419, Adjusted R-squared: -0.01158
## F-statistic: 0.3188 on 2 and 117 DF, p-value: 0.7277

linearMod50 <- lm(formula = base_pay_change ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_salaried_regression)
summary(linearMod50)

##
linearMod51 <- lm(formula = base_pay_change ~ gender_Female + gender_Male + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_salaried_regression)
```r
## F-statistic: 1.463 on 9 and 110 DF, p-value: 0.1706
linearMod52 <- lm(formula = base_pay_change ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_salaried_regression)
summary(linearMod52)
```
```r
## Call:
## lm(formula = base_pay_change ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_salaried_regression)
## Residuals:
##    Min     1Q Median     3Q    Max
## -1745.2 -580.7 -163.9  286.8  3988.9
## Coefficients: (2 not defined because of singularities)
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)       755.3     1075.9  0.702  0.4841
## race_grouping_white  412.0     1036.1  0.398  0.6917
## race_grouping_person_of_color  174.9     1048.7  0.167  0.8678
## age_group_5_under_25  148.3     1051.7  0.141  0.8881
## age_group_5_25to29    644.7     369.7  1.744  0.0840 .
## age_group_5_30to34    237.5     380.1  0.625  0.5335
## age_group_5_35to39    786.4     398.8  1.972  0.0511 .
## age_group_5_40to44   1217.3     465.2  2.617  0.0101 *
## age_group_5_45to49    288.2     372.2  0.774  0.4404
## age_group_5_50to54    213.1     405.6  0.525  0.6004
## age_group_5_55to59   630.9     507.9  1.242  0.2168
## age_group_5_60to64 NA          NA  NA   NA
## age_group_5_65_over NA          NA  NA   NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error: 1010 on 109 degrees of freedom
## Multiple R-squared:  0.1029, Adjusted R-squared:  0.02056
## F-statistic: 1.25 on 10 and 109 DF, p-value: 0.2681
linearMod53 <- lm(formula = base_pay_change ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_salaried_regression)
summary(linearMod53)
```
```r
## Call:
## lm(formula = base_pay_change ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_salaried_regression)
## Residuals:
##     Min      1Q     Median      3Q      Max
## -1726.0  -516.9   -190.1   298.9  3825.7
## ```
## Coefficients: (3 not defined because of singularities)

|                | Estimate | Std. Error | t value | Pr(>|t|) |
|----------------|----------|------------|---------|----------|
| (Intercept)    | 628.95   | 1078.72    | 0.583   | 0.5611   |
| gender_Female  | 283.57   | 235.15     | 1.206   | 0.2305   |
| gender_Male    | NA       | NA         | NA      | NA       |
| race_grouping_white | 486.84 | 1035.84    | 0.470   | 0.6393   |
| race_grouping_person_of_color | 300.96 | 1051.74    | 0.286   | 0.7753   |
| age_group_5_25to29 | 487.48 | 391.31     | 1.246   | 0.2155   |
| age_group_5_30to34 | 35.62  | 414.64     | 0.086   | 0.9317   |
| age_group_5_35to39 | 754.73 | 398.79     | 1.893   | 0.0611   |
| age_group_5_40to44 | 966.05 | 508.81     | 1.899   | 0.0603   |
| age_group_5_45to49 | 168.08 | 384.54     | 0.437   | 0.6629   |
| age_group_5_50to54 | 34.69  | 430.98     | 0.080   | 0.9360   |
| age_group_5_55to59 | 381.78 | 547.29     | 0.698   | 0.4869   |
| age_group_5_60to64 | 754.73 | 398.79     | 1.893   | 0.0611   |
| age_group_5_65_over | NA    | NA         | NA      | NA       |

---

Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Residual standard error: 1008 on 108 degrees of freedom
## Multiple R-squared: 0.1148, Adjusted R-squared: 0.02463
## F-statistic: 1.273 on 11 and 108 DF, p-value: 0.2497

linearMod54 <- lm(formula = performance_rating ~ gender_Female + gender_Male, data=merit_raises_combined_salaried_regression)

summary(linearMod54)

## Call:
## lm(formula = performance_rating ~ gender_Female + gender_Male, 
##    data = merit_raises_combined_salaried_regression)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##-0.64714 -0.19583 -0.04714  0.20417  0.90417
##
## Coefficients: (1 not defined because of singularities)
##                Estimate  Std. Error t value Pr(>|t|)
## (Intercept)    3.39583  0.04602  73.786  <2e-16 ***
## gender_Female  0.05131  0.05975  0.859  0.392
## gender_Male NA   NA       NA       NA
##
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3189 on 116 degrees of freedom
## (2 observations deleted due to missingness)
## Multiple R-squared: 0.006316,  Adjusted R-squared: -0.00225
## F-statistic: 0.7373 on 1 and 116 DF,  p-value: 0.3923

linearMod55 <- lm(formula = performance_rating ~ race_grouping_white + race_grouping_person_of_color, data=merit_raises_combined_salaried_regression)

summary(linearMod55)

## Call:
## lm(formula = performance_rating ~ race_grouping_white + race_grouping_person_of_color,
## data = merit_raises_combined_salaried_regression
##
## Residuals:
## Min 1Q Median 3Q Max
## -0.63514 -0.23514 -0.03514 0.18837 0.88837
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.40000 0.32104 10.591 <2e-16 ***
## race_grouping_white 0.03514 0.32320 0.109 0.914
## race_grouping_person_of_color 0.01163 0.32475 0.036 0.971
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.321 on 115 degrees of freedom
## (2 observations deleted due to missingness)
## Multiple R-squared: 0.001325, Adjusted R-squared: -0.01604
## F-statistic: 0.07628 on 2 and 115 DF, p-value: 0.9266

linearMod56 <- lm(formula = performance_rating ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_salaried_regression)

summary(linearMod56)

## Call:
## lm(formula = performance_rating ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_salaried_regression)
##
## Residuals:
## Min 1Q Median 3Q Max
## -0.65112 -0.20182 -0.04513 0.20716 0.91016
##
## Coefficients: (1 not defined because of singularities)
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.35070 0.32750 10.231 <2e-16 ***
## gender_Female 0.04930 0.06209 0.794 0.429
## gender_Male NA NA NA NA
## race_grouping_white 0.05112 0.32435 0.158 0.875
## race_grouping_person_of_color 0.03915 0.32712 0.120 0.905
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3216 on 114 degrees of freedom
## (2 observations deleted due to missingness)
## Multiple R-squared: 0.006818, Adjusted R-squared: -0.01932
## F-statistic: 0.2609 on 3 and 114 DF, p-value: 0.8535

linearMod57 <- lm(formula = performance_rating ~ gender_Female + gender_Male + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + data = merit_raises_combined_salaried_regression)

summary(linearMod57)

## Call:
## lm(formula = performance_rating ~ gender_Female + gender_Male + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + data = merit_raises_combined_salaried_regression)
##
## Residuals:
## Min 1Q Median 3Q Max
## -0.63514 -0.23514 -0.03514 0.18837 0.88837
##
## Coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 3.40000 0.32104 10.591 <2e-16 ***
## gender_Female 0.03514 0.32320 0.109 0.914
## gender_Male NA NA NA NA
## age_group_5_under_25 0.01163 0.32475 0.036 0.971
## age_group_5_25to29 0.01163 0.32475 0.036 0.971
## age_group_5_30to34 0.01163 0.32475 0.036 0.971
## age_group_5_35to39 0.01163 0.32475 0.036 0.971
## age_group_5_40to44 0.01163 0.32475 0.036 0.971
## age_group_5_45to49 0.01163 0.32475 0.036 0.971
## age_group_5_50to54 0.01163 0.32475 0.036 0.971
## age_group_5_55to59 0.01163 0.32475 0.036 0.971
## age_group_5_60to64 0.01163 0.32475 0.036 0.971
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.321 on 115 degrees of freedom
## (2 observations deleted due to missingness)
## Multiple R-squared: 0.001325, Adjusted R-squared: -0.01604
## F-statistic: 0.07628 on 2 and 115 DF, p-value: 0.9266

linearMod56 <- lm(formula = performance_rating ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_salaried_regression)

summary(linearMod66)
linearMod58 <- lm(formula = performance_rating ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_salaried_regression)

summary(linearMod58)
## age_group_5_50to54  0.299782  0.125425  2.390  0.0186 *
## age_group_5_55to59  0.330065  0.157051  2.102  0.0379 *
## age_group_5_60to64  NA  NA  NA  NA  NA
## age_group_5_65_over  NA  NA  NA  NA  NA
## ---
## Signif. codes:  0 '***'  0.001 '**'  0.01 '*'  0.05 '.'  0.1  ' ' 1
##
## Residual standard error: 0.3124 on 107 degrees of freedom
## (2 observations deleted due to missingness)
## Multiple R-squared: 0.1203, Adjusted R-squared: 0.03808
## F-statistic: 1.463 on 10 and 107 DF,  p-value: 0.1633

```
linearMod59 <- lm(formula = performance_rating ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_salaried_regression)
```

```
summary(linearMod59)
```

```
## Call:
## lm(formula = performance_rating ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_salaried_regression)
##
## Residuals:
##     Min     1Q Median     3Q    Max
## -0.62357 -0.18119 -0.04026  0.15697  0.86251
##
## Coefficients: (3 not defined because of singularities)
##                    Estimate      Std. Error     t value Pr(>|t|)
## (Intercept)         3.365646      0.336384    10.005  <2e-16 ***
## gender_Female       0.001768      0.075435     0.023   0.9813
## race_grouping_white NA   NA   NA   NA
## race_grouping_person_of_color -0.042505     0.328885    -0.129   0.8974
## age_group_5_under_25 -0.043029     0.327001    -0.132   0.8956
## age_group_5_25to29   0.032585      0.122617     0.266   0.7909
## age_group_5_30to34   0.044783      0.133103     0.336   0.7372
## age_group_5_35to39   0.140115      0.124436     1.126   0.2627
## age_group_5_40to44   0.137660      0.159528     0.863   0.3901
## age_group_5_45to49  -0.007312     0.120416    -0.061   0.9517
## age_group_5_50to54   0.328498      0.171364     1.917   0.0579 .
## age_group_5_55to59   0.137660      0.159528     0.863   0.3901
## age_group_5_60to64   0.137660      0.159528     0.863   0.3901
## age_group_5_65_over  NA   NA   NA   NA
##
## Residual standard error: 0.3138 on 106 degrees of freedom
## (2 observations deleted due to missingness)
## Multiple R-squared: 0.1203, Adjusted R-squared: 0.02901
## F-statistic: 1.318 on 11 and 106 DF,  p-value: 0.2247
```

```
commercial_hourly_regression <- fastDummies::dummy_cols(commercial_hourly_regression, select_columns = c('department', 'gender', 'race_ethnicity', 'age_group_5', 'years_of_service_grouped', 'dept', 'desk', 'tier', 'race_grouping'))
```
names(commercial_hourly_regression) <- gsub('|', '_', names(commercial_hourly_regression))
names(commercial_hourly_regression) <- gsub('-', 'to', names(commercial_hourly_regression))
names(commercial_hourly_regression) <- gsub('\|+', '_over', names(commercial_hourly_regression))
names(commercial_hourly_regression) <- gsub('<', 'under', names(commercial_hourly_regression))

linearMod60 <- lm(formula = current_base_pay ~ gender_Female + gender_Male, data = commercial_hourly_regression)
summary(linearMod60)

##
## Call:
## lm(formula = current_base_pay ~ gender_Female + gender_Male, 
##     data = commercial_hourly_regression)
##
## Residuals:
## Min  1Q  Median   3Q  Max
## -10.731 -4.314 -1.518 3.761 29.419
##
## Coefficients: (1 not defined because of singularities)
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 25.7881   0.7581 34.019  < 2e-16 ***
## gender_Female 3.9126   1.0684  3.662   0.00035 ***
## gender_Male NA       NA      NA    NA
##
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.477 on 145 degrees of freedom
## Multiple R-squared: 0.08466, Adjusted R-squared: 0.07834
## F-statistic: 13.41 on 1 and 145 DF,  p-value: 0.0003499

linearMod61 <- lm(formula = current_base_pay ~ race_grouping_white + race_grouping_person_of_color, data = commercial_hourly_regression)
summary(linearMod61)

##
## Call:
## lm(formula = current_base_pay ~ race_grouping_white + race_grouping_person_of_color, 
##     data = commercial_hourly_regression)
##
## Residuals:
## Min  1Q Median   3Q  Max
## -11.2002 -4.4456 -0.9006 3.5548 28.1098
##
## Coefficients:
## Estimate Std. Error t value  Pr(>|t|)
## (Intercept) 22.1133   3.7104  5.961  1.85e-08 ***
## race_grouping_white 8.8971   3.8370  2.319   0.0218 *
## race_grouping_person_of_color 4.4274   3.7647  1.176   0.2415
##
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.425 on 144 degrees of freedom
## Multiple R-squared: 0.1054, Adjusted R-squared: 0.09293
## F-statistic: 8.479 on 2 and 144 DF,  p-value: 0.0003499

linearMod62 <- lm(formula = current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color, data = commercial_hourly_regression)
summary(linearMod62)

174
## Call:
lm(formula = current_base_pay ~ gender_Female + gender_Male +
    race_grouping_white + race_grouping_person_of_color, data = commercial_hourly_regression)
##
## Residuals:
##    Min 1Q Median 3Q Max
## -12.330 -3.851 -1.531 2.554 26.270
##
## Coefficients: (1 not defined because of singularities)
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    22.113     3.559   6.213 5.36e-09 ***
## gender_Female  3.767     1.028   3.665 0.000348 ***
## gender_Male NA NA NA NA
## race_grouping_white  6.969     3.719   1.874 0.062943 .
## race_grouping_person_of_color  2.488     3.650   0.682 0.496647
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.165 on 143 degrees of freedom
## Multiple R-squared: 0.1822, Adjusted R-squared: 0.165
## F-statistic: 10.62 on 3 and 143 DF,  p-value: 2.4e-06

linearMod63 <- lm(formula = current_base_pay ~ gender_Female + gender_Male + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = commercial_hourly_regression)

## Call:
lm(formula = current_base_pay ~ gender_Female + gender_Male +
    age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 +
    age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
    age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
    age_group_5_65_over, data = commercial_hourly_regression)
##
## Residuals:
##    Min 1Q Median 3Q Max
## -11.3940 -3.8376 -0.9446 3.1079 28.4860
##
## Coefficients: (2 not defined because of singularities)
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)    24.30914 1.97881 12.285 < 2e-16 ***
## gender_Female  3.75388 1.08589 3.457 0.000729 ***
## gender_Male NA NA NA NA
## age_group_5_under_25 -0.03328 2.71181 -0.122 0.90225
## age_group_5_25to29  3.14429 2.35523 1.335 0.184098
## age_group_5_30to34  3.14160 2.86587 1.096 0.274924
## age_group_5_35to39  6.17705 2.60480 2.371 0.019123 *
## age_group_5_40to44  0.95164 2.47439 0.385 0.701137
## age_group_5_45to49  2.57102 2.45973 1.045 0.297765
## age_group_5_50to54 -0.40099 2.43587 -0.165 0.869487
## age_group_5_55to59  0.60055 2.49206 0.241 0.809931
## age_group_5_60to64 -0.80863 2.60371 -0.311 0.756604
## age_group_5_65_over NA NA NA NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

175
## Residual standard error: 6.356 on 136 degrees of freedom
## Multiple R-squared: 0.1733, Adjusted R-squared: 0.1125
## F-statistic: 2.851 on 10 and 136 DF, p-value: 0.00298

linearMod64 <- lm(formula = current_base_pay ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = commercial_hourly_regression)

summary(linearMod64)

## Call:
## lm(formula = current_base_pay ~ race_grouping_white + race_grouping_person_of_color +
## age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 +
## age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
## age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
## age_group_5_65_over, data = commercial_hourly_regression)
##
## Residuals:
## Min 1Q Median 3Q Max
## -10.4844 -3.8899 -0.9866 3.0028 27.2032
##
## Coefficients: (1 not defined because of singularities)
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 17.8158 4.1801 4.262 3.78e-05 ***
## race_grouping_white 10.8972 3.8021 2.866 0.00482 **
## race_grouping_person_of_color 6.6583 3.7576 1.772 0.07866 .
## age_group_5_under_25 -0.3745 2.6699 -0.140 0.88864
## age_group_5_25to29 3.5514 2.3257 1.527 0.12909
## age_group_5_30to34 5.0083 2.8306 1.769 0.07909 .
## age_group_5_35to39 6.0117 2.5857 2.325 0.02156 *
## age_group_5_40to44 3.3295 2.4475 1.360 0.17598
## age_group_5_45to49 3.2038 2.3333 1.317 0.19018
## age_group_5_50to54 -0.4921 2.3985 -0.205 0.83776
## age_group_5_55to59 1.2226 2.4532 0.498 0.61905
## age_group_5_60to64 0.1069 2.5775 0.041 0.96698
## age_group_5_65_over NA NA NA NA
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.261 on 135 degrees of freedom
## Multiple R-squared: 0.2035, Adjusted R-squared: 0.1386
## F-statistic: 3.136 on 11 and 135 DF, p-value: 0.0008472

linearMod65 <- lm(formula = current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = commercial_hourly_regression)

summary(linearMod65)

## Call:
## lm(formula = current_base_pay ~ gender_Female + gender_Male +
## race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 +
## age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 +
## age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 +
## age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over,
## data = commercial_hourly_regression)
##
## Residuals:
## Min 1Q Median 3Q Max
## -11.4252 -3.9045 -0.7517 2.7593 25.1857
##
## Coefficients: (2 not defined because of singularities)
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 18.35904 4.04063 4.544 1.22e-05 ***
## gender_Female 3.43733 1.04958 3.275 0.00135 **
## gender_Male NA NA NA NA
## race_grouping_white 8.78809 3.72820 2.357 0.01986 *
## race_grouping_person_of_color 4.55451 3.68554 1.236 0.21870
## age_group_5_under_25 -0.06206 2.58039 -0.024 0.98085
## age_group_5_25to29 2.83157 2.25695 1.255 0.21181
## age_group_5_30to34 4.27783 2.74288 1.560 0.12121
## age_group_5_35to39 6.09079 2.49738 2.439 0.01604 *
## age_group_5_40to44 2.34053 2.38306 0.982 0.32779
## age_group_5_45to49 3.34986 2.35050 1.425 0.15643
## age_group_5_50to54 -0.07588 2.32001 -0.033 0.97396
## age_group_5_55to59 0.85121 2.37207 0.359 0.72027
## age_group_5_60to64 0.08175 2.48936 0.033 0.97385
## age_group_5_65_over NA NA NA NA
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.047 on 134 degrees of freedom
## Multiple R-squared: 0.2625, Adjusted R-squared: 0.1965
## F-statistic: 3.975 on 12 and 134 DF, p-value: 2.988e-05

linearMod66 <- lm(formula = current_base_pay ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over + years_of_service_grouped_0 + years_of_service_grouped_1to2 + years_of_service_grouped_3to5 + years_of_service_grouped_6to10 + years_of_service_grouped_11to15 + years_of_service_grouped_16to20 + years_of_service_grouped_21to25 + years_of_service_grouped_25_over, data = commercial_hourly_regression)

# Residuals:
## Min 1Q Median 3Q Max
## -11.3880 -3.2873 -0.7906 2.6392 24.8288
##
## Coefficients: (3 not defined because of singularities)
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) 18.71412 4.45884 4.197 5.04e-05 ***
## gender_Female 3.04122 1.25333 2.427 0.0166 *
## gender_Male NA NA NA NA
## race_grouping_white 8.52247 4.05027 2.104 0.0373 *
## race_grouping_person_of_color 4.04668 3.96852 1.020 0.3098
## age_group_5_under_25 -0.43984 3.05384 -0.144 0.8857
## age_group_5_25to29 2.31526 2.80073 0.827 0.4100
## age_group_5_30to34 4.90639 3.20985 1.529 0.1289
## age_group_5_35to39 5.89797 2.79979 2.107 0.0371 *
## age_group_5_40to44 2.40918 2.73489 0.881 0.3800
## age_group_5_45to49 2.82012 2.60479 1.083 0.2810
## age_group_5_50to54 -0.30475 2.48080 -0.123 0.9024
## age_group_5_55to59 0.92733 2.44641 0.379 0.7053
## age_group_5_60to64 -0.40829 2.66712 -0.153 0.8786
## age_group_5_65_over NA NA NA NA
## years_of_service_grouped_0 -0.14159 2.73822 -0.052 0.9588
## years_of_service_grouped_1to2 1.19324 2.56055 0.466 0.6420
## years_of_service_grouped_3to5 -0.97287 2.76173 -0.352 0.7252
## years_of_service_grouped_6to10 0.04366 2.54249 0.017 0.9863
## years_of_service_grouped_11to15 0.46224 2.81566 0.164 0.8699
## years_of_service_grouped_16to20 0.98286 2.41854 0.406 0.6851
## years_of_service_grouped_21to25 2.57982 2.83996 0.908 0.3654
## years_of_service_grouped_25_over NA NA NA NA
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 6.156 on 127 degrees of freedom
## Multiple R-squared: 0.2757, Adjusted R-squared: 0.1673
## F-statistic: 2.544 on 19 and 127 DF, p-value: 0.001075

merit_raises_combined_hourly_regression <- filter(merit_raises_combined, dept == 'Commercial', pay_rate_type == 'Hourly')

merit_raises_combined_hourly_regression <- fastDummies::dummy_cols(merit_raises_combined_hourly_regression, select_columns = c(gender, race_grouping, age_group_5, years_of_service_grouped_0, years_of_service_grouped_1to2, years_of_service_grouped_3to5, years_of_service_grouped_6to10, years_of_service_grouped_11to15, years_of_service_grouped_16to20, years_of_service_grouped_21to25, years_of_service_grouped_25_over))

names(merit_raises_combined_hourly_regression) <- gsub('_', '', names(merit_raises_combined_hourly_regression))

linearMod67 <- lm(formula = base_pay_change ~ gender_Female + gender_Male, data = merit_raises_combined_hourly_regression)

summary(linearMod67)

# Call:
# lm(formula = base_pay_change ~ gender_Female + gender_Male, data = merit_raises_combined_hourly_regression)
# # Residuals:
# #    Min  1Q Median  3Q    Max
# # -0.35809 -0.12809 -0.03789 0.07230 1.08191
# # # Coefficients: (1 not defined because of singularities)
# # Estimate Std. Error t value Pr(>|t|)
# # (Intercept) 0.34770 0.01886 18.434 <2e-16 ***
# # gender_Female 0.11039 0.02618 4.217 3.43e-05 ***
# # gender_Male NA NA NA NA
# # ---
# # Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
# # Residual standard error: 0.2117 on 260 degrees of freedom
# # Multiple R-squared: 0.06401, Adjusted R-squared: 0.06041
# # F-statistic: 17.78 on 1 and 260 DF, p-value: 3.427e-05
linearMod68 <- lm(formula = base_pay_change ~ race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_hourly_regression)
##
## Call:
## lm(formula = base_pay_change ~ race_grouping_white + race_grouping_person_of_color, 
##     data = merit_raises_combined_hourly_regression)
##
## Residuals:
##    Min     1Q  Median     3Q    Max
## -0.36944 -0.13105  0.04944  0.07895  1.07056
##
## Coefficients: (1 not defined because of singularities)
##             Estimate  Std. Error t value Pr(>|t|)
## (Intercept)  0.38105     0.01558  24.464  < 2e-16 ***
## race_grouping_white  0.08839     0.02992   2.954  0.00342 **
## race_grouping_person_of_color NA         NA         NA         NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2153 on 260 degrees of freedom
## Multiple R-squared: 0.03247,    Adjusted R-squared: 0.02875
## F-statistic: 8.727 on 1 and 260 DF,  p-value: 0.003423

linearMod69 <- lm(formula = base_pay_change ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_hourly_regression)
##
## Call:
## lm(formula = base_pay_change ~ gender_Female + gender_Male + 
##     race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_hourly_regression)
##
## Residuals:
##    Min     1Q  Median     3Q    Max
## -0.42912 -0.11989  0.05686  0.08011  1.01088
##
## Coefficients: (2 not defined because of singularities)
##             Estimate  Std. Error t value Pr(>|t|)
## (Intercept)  0.31989     0.02037  15.700  < 2e-16 ***
## gender_Female  0.11452     0.02573   4.450 1.28e-05 ***
## gender_Male NA         NA         NA         NA
## race_grouping_white  0.09471     0.02893   3.274  0.00121 **
## race_grouping_person_of_color NA         NA         NA         NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2079 on 259 degrees of freedom
## Multiple R-squared: 0.1012,    Adjusted R-squared: 0.09426
## F-statistic: 14.58 on 2 and 259 DF,  p-value: 9.987e-07

linearMod70 <- lm(formula = base_pay_change ~ gender_Female + gender_Male + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_hourly_regression)
##
## Call:
## lm(formula = base_pay_change ~ gender_Female + gender_Male + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_hourly_regression)
##
## Residuals:
##    Min     1Q  Median     3Q    Max
## -0.42912 -0.11989  0.05686  0.08011  1.01088
##
## Coefficients: (not defined because of singularities)
##             Estimate  Std. Error t value Pr(>|t|)
## (Intercept) NA         NA         NA         NA
## gender_Female NA         NA         NA         NA
## gender_Male NA         NA         NA         NA
## age_group_5_under_25 NA         NA         NA         NA
## age_group_5_25to29 NA         NA         NA         NA
## age_group_5_30to34 NA         NA         NA         NA
## age_group_5_35to39 NA         NA         NA         NA
## age_group_5_40to44 NA         NA         NA         NA
## age_group_5_45to49 NA         NA         NA         NA
## age_group_5_50to54 NA         NA         NA         NA
## age_group_5_55to59 NA         NA         NA         NA
## age_group_5_60to64 NA         NA         NA         NA
## age_group_5_65_over NA         NA         NA         NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2079 on 259 degrees of freedom
## Multiple R-squared: 0.1012,    Adjusted R-squared: 0.09426
## F-statistic: 14.58 on 2 and 259 DF,  p-value: 9.987e-07
## lm(formula = base_pay_change ~ gender_Female + gender_Male +
   age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 +
   age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
   age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
   age_group_5_65_over, data = merit_raises_combined_hourly_regression)
##
## Residuals:
##     Min      1Q  Median       3Q      Max
## -0.36671 -0.12659  -0.03807   0.09317  1.10473
##
## Coefficients: (2 not defined because of singularities)
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          0.28807    0.04187   6.880  4.76e-11 ***
## gender_Female        0.10705    0.02859   3.744   0.000224 ***
## gender_Male          NA          NA         NA       NA
## age_group_5_under_25 0.23052    0.10196   2.261   0.024625 *
## age_group_5_25to29   0.04015    0.06040   0.665   0.506887   
## age_group_5_30to34   0.09536    0.06767   1.409   0.160013
## age_group_5_35to39   0.18773    0.06092   3.082   0.002289 **
## age_group_5_40to44   0.08816    0.05992   1.471   0.142450
## age_group_5_45to49   0.08158    0.05712   1.428   0.154453
## age_group_5_50to54   0.01802    0.05554   0.324   0.745869
## age_group_5_55to59   0.04244    0.05497   0.772   0.440778
## age_group_5_60to64   0.02171    0.05503   0.394   0.693592
## age_group_5_65_over  NA          NA         NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2081 on 251 degrees of freedom
## Multiple R-squared: 0.127, Adjusted R-squared: 0.09219
## F-statistic: 3.651 on 10 and 251 DF, p-value: 0.000145

linearMod71 <- lm(formula = base_pay_change ~ race_grouping_white + race_grouping_person_of_color + age_5_under_25 + age_5_25to29 + age_5_30to34 + age_5_35to39 + age_5_40to44 + age_5_45to49 + age_5_50to54 + age_5_55to59 + age_5_60to64 + age_5_65_over, data = merit_raises_combined_hourly_regression)

##
## Call:
## lm(formula = base_pay_change ~ race_grouping_white + race_grouping_person_of_color +
##    age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 +
##    age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 +
##    age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 +
##    age_group_5_65_over, data = merit_raises_combined_hourly_regression)
##
## Residuals:
##     Min      1Q  Median      3Q      Max
## -0.34399 -0.12826 -0.03793  0.08002  1.09654
##
## Coefficients: (2 not defined because of singularities)
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)          0.28793    0.04257   6.763   9.42e-11 ***
## race_grouping_white  0.08634    0.03092   2.792   0.00564 **
## race_grouping_person_of_color NA          NA       NA       NA
## age_group_5_under_25  0.23480    0.10317   2.276   0.02370 *
## age_group_5_25to29   0.06919    0.05994   1.154   0.24953
## age_group_5_30to34   0.16213    0.06621   2.449   0.01502 *
## age_group_5_35to39  0.17782  0.06207  2.865  0.00452 **
## age_group_5_40to44  0.15605  0.05749  2.714  0.00710 **
## age_group_5_45to49  0.13514  0.05623  2.403  0.01697 *
## age_group_5_50to54  0.04568  0.05552  0.823  0.41139
## age_group_5_55to59  0.08178  0.05453  1.500  0.13492
## age_group_5_60to64  0.04124  0.05549  0.743  0.45802
## age_group_5_65_over NA NA NA NA
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2106 on 251 degrees of freedom
## Multiple R-squared: 0.106, Adjusted R-squared: 0.07036
## F-statistic: 2.975 on 10 and 251 DF, p-value: 0.001468

linearMod72 <- lm(formula = base_pay_change ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_hourly_regression)

summary(linearMod72)

## Call:
## lm(formula = base_pay_change ~ gender_Female + gender_Male +
##     race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 +
##     age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 +
##     age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 +
##     age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over,
## data = merit_raises_combined_hourly_regression)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.37347 -0.12809 -0.03730  0.07365  1.06653
##
## Coefficients: (3 not defined because of singularities)
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)   0.267098   0.041599  6.421  6.78e-10 ***
## gender_Female 0.115674   0.028184  4.104  5.50e-05 ***
## gender_Male NA NA NA NA
## race_grouping_white 0.097972   0.030125  3.252  0.00130 **
## race_grouping_person_of_color NA NA NA NA
## age_group_5_under_25 0.230173   0.100069  2.300  0.02226 *
## age_group_5_25to29 -0.007271   0.061049 -0.119  0.90529
## age_group_5_30to34 0.092451   0.066419  1.392  0.16518
## age_group_5_35to39 0.161822   0.060321  2.683  0.00779 **
## age_group_5_40to44 0.078351   0.058884  1.331  0.18453
## age_group_5_45to49 0.081860   0.056060  1.460  0.14548
## age_group_5_50to54 0.009004   0.054584  0.165  0.86911
## age_group_5_55to59 0.037571   0.053968  0.696  0.48697
## age_group_5_60to64 0.022601   0.054009  0.418  0.67596
## age_group_5_65_over NA NA NA NA
##
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2043 on 250 degrees of freedom
## Multiple R-squared:  0.1624, Adjusted R-squared:  0.1256
## F-statistic: 4.407 on 11 and 250 DF, p-value: 4.681e-06
linearMod73 <- lm(formula = performance_rating ~ gender_Female + gender_Male, data = merit_raises_combined_hourly_regression)
summary(linearMod73)

## Call:
## lm(formula = performance_rating ~ gender_Female + gender_Male,
##     data = merit_raises_combined_hourly_regression)
## ## Residuals:
##     Min       1Q   Median       3Q      Max
## -0.65410 -0.22548 -0.02541  0.14589  0.84589
## ## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.2254 0.02192 147.173  < 2e-16 ***
## gender_Female 0.1287 0.03047   4.223 3.34e-05 ***
## gender_Male NA    NA     NA     NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## ## Residual standard error: 0.246 on 259 degrees of freedom
## (1 observation deleted due to missingness)
## Multiple R-squared: 0.06441, Adjusted R-squared: 0.0608
## F-statistic: 17.83 on 1 and 259 DF,  p-value: 3.344e-05

linearMod74 <- lm(formula = performance_rating ~ race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_hourly_regression)
summary(linearMod74)

## Call:
## lm(formula = performance_rating ~ race_grouping_white + race_grouping_person_of_color,
##     data = merit_raises_combined_hourly_regression)
## ## Residuals:
##     Min       1Q   Median       3Q      Max
## -0.57737 -0.23099 -0.03099  0.16901  0.92263
## ## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.2774 0.01837 178.418  < 2e-16 ***
## race_grouping_white 0.0536 0.03522   1.522  0.129
## race_grouping_person_of_color NA    NA     NA     NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## ## Residual standard error: 0.2532 on 259 degrees of freedom
## (1 observation deleted due to missingness)
## Multiple R-squared: 0.008869, Adjusted R-squared: 0.005043
## F-statistic: 2.318 on 1 and 259 DF,  p-value: 0.1291

linearMod75 <- lm(formula = performance_rating ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_hourly_regression)
summary(linearMod75)

## Call:
```r
lm(formula = performance_rating ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color, data = merit_raises_combined_hourly_regression)

Residuals:
Min 1Q Median 3Q Max
-0.63883 -0.20762 -0.03883 0.16117 0.86117

Coefficients: (2 not defined because of singularities)

(Intercept) 3.20762 0.02401 133.570 < 2e-16 ***
gender_Female 0.13121 0.03038 4.319 2.24e-05 ***
gender_Male NA NA NA NA
race_grouping_white 0.06053 0.03411 1.774 0.0772 .
race_grouping_person_of_color NA NA NA NA

---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.245 on 258 degrees of freedom (1 observation deleted due to missingness)
Multiple R-squared: 0.07569, Adjusted R-squared: 0.06853
F-statistic: 10.56 on 2 and 258 DF, p-value: 3.893e-05
```

```r
linearMod76 <- lm(formula = performance_rating ~ gender_Female + gender_Male + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_hourly_regression)
summary(linearMod76)

Call:
lm(formula = performance_rating ~ gender_Female + gender_Male + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_hourly_regression)

Residuals:
Min 1Q Median 3Q Max
-0.59991 -0.19415 -0.01871 0.15558 0.85594

Coefficients: (2 not defined because of singularities)

(Intercept) 3.10402 0.04884 63.560 < 2e-16 ***
gender_Female 0.09989 0.03341 2.990 0.00307 **
gender_Male NA NA NA NA
age_group_5_under_25 0.09600 0.11891 0.807 0.42023
age_group_5_25to29 0.14051 0.07046 1.994 0.04720 *
age_group_5_30to34 0.11372 0.07893 1.441 0.15091
age_group_5_35to39 0.20510 0.07105 2.887 0.00423 **
age_group_5_40to44 0.22366 0.06989 3.200 0.00155 **
age_group_5_45to49 0.11480 0.06662 1.723 0.08610 .
age_group_5_50to54 0.14015 0.06478 2.163 0.03145 *
age_group_5_55to59 0.19049 0.06437 2.959 0.00338 **
age_group_5_60to64 0.09014 0.06418 1.404 0.16142
age_group_5_65_over NA NA NA NA

---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```
## Residual standard error: 0.2427 on 250 degrees of freedom
## (1 observation deleted due to missingness)
## Multiple R-squared: 0.1209, Adjusted R-squared: 0.08575
## F-statistic: 3.439 on 10 and 250 DF,  p-value: 0.0003027

linearMod77 <- `lm(formula = performance_rating ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_hourly_regression)

```
## Call:
## lm(formula = performance_rating ~ race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_hourly_regression)
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.53801 -0.17477 -0.01916  0.16199  0.91980
## Coefficients: (2 not defined because of singularities)
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          3.11171   0.04980   62.48  < 2e-16 ***
## race_grouping_white  0.04145   0.03619    1.15   0.2532
## race_grouping_person_of_color NA NA NA NA
## age_group_5_under_25  0.10000   0.12070    0.83   0.4082
## age_group_5_25to29   0.18441   0.07013    2.63  0.0091 **
## age_group_5_30to34   0.17509   0.07745    2.26  0.0246 *
## age_group_5_35to39   0.20581   0.07261    2.83  0.0049 ***
## age_group_5_40to44   0.28863   0.06725    4.29  2.54e-05 ***
## age_group_5_45to49   0.16306   0.06578    2.48  0.0129 *
## age_group_5_50to54   0.16849   0.06495    2.59  0.0100 *
## age_group_5_55to59   0.22630   0.06414    3.52  0.0005 ***
## age_group_5_60to64   0.10745   0.06491    1.66  0.09918
## age_group_5_65_over NA NA NA NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error: 0.2464 on 250 degrees of freedom
## (1 observation deleted due to missingness)
## Multiple R-squared: 0.09424,  Adjusted R-squared: 0.05801
## F-statistic: 2.601 on 10 and 250 DF,  p-value: 0.00509
```

linearMod78 <- `lm(formula = performance_rating ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_hourly_regression)

```
## Call:
## lm(formula = performance_rating ~ gender_Female + gender_Male + race_grouping_white + race_grouping_person_of_color + age_group_5_under_25 + age_group_5_25to29 + age_group_5_30to34 + age_group_5_35to39 + age_group_5_40to44 + age_group_5_45to49 + age_group_5_50to54 + age_group_5_55to59 + age_group_5_60to64 + age_group_5_65_over, data = merit_raises_combined_hourly_regression)
## Residuals:
##     Min      1Q  Median      3Q     Max
##    0.001   0.01   0.05 '.' 0.1 ' 1
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          3.11171   0.04980   62.48  < 2e-16 ***
## gender_Female        0.04145   0.03619    1.15   0.2532
## gender_Male          NA NA NA NA
## race_grouping_white  0.04145   0.03619    1.15   0.2532
## race_grouping_person_of_color NA NA NA NA
## age_group_5_under_25  0.10000   0.12070    0.83   0.4082
## age_group_5_25to29   0.18441   0.07013    2.63  0.0091 **
## age_group_5_30to34   0.17509   0.07745    2.26  0.0246 *
## age_group_5_35to39   0.20581   0.07261    2.83  0.0049 ***
## age_group_5_40to44   0.28863   0.06725    4.29  2.54e-05 ***
## age_group_5_45to49   0.16306   0.06578    2.48  0.0129 *
## age_group_5_50to54   0.16849   0.06495    2.59  0.0100 *
## age_group_5_55to59   0.22630   0.06414    3.52  0.0005 ***
## age_group_5_60to64   0.10745   0.06491    1.66  0.09918
## age_group_5_65_over NA NA NA NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## Residual standard error: 0.2464 on 250 degrees of freedom
## (1 observation deleted due to missingness)
## Multiple R-squared: 0.09424,  Adjusted R-squared: 0.05801
## F-statistic: 2.601 on 10 and 250 DF,  p-value: 0.00509
## Residuals:

<table>
<thead>
<tr>
<th>Min</th>
<th>1Q</th>
<th>Median</th>
<th>3Q</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>-0.59314</td>
<td>-0.18358</td>
<td>-0.01585</td>
<td>0.13979</td>
<td>0.86727</td>
</tr>
</tbody>
</table>

## Coefficients: (3 not defined because of singularities)

|                     | Estimate | Std. Error | t value | Pr(>|t|) |
|---------------------|----------|------------|---------|---------|
| (Intercept)         | 3.09296  | 0.04932    | 62.706  | < 2e-16 *** |
| gender_Female       | 0.10436  | 0.03348    | 3.117   | 0.00204 ** |
| gender_Male         | NA       | NA         | NA      | NA      |
| race_grouping_white | 0.05172  | 0.03573    | 1.447   | 0.14903 |
| race_grouping_person_of_color | NA | NA | NA | NA |
| age_group_5_under_25| 0.09583  | 0.11865    | 0.808   | 0.42008 |
| age_group_5_25to29  | 0.11553  | 0.07239    | 1.596   | 0.11177 |
| age_group_5_30to34  | 0.11223  | 0.07876    | 1.425   | 0.15544 |
| age_group_5_35to39  | 0.19143  | 0.07152    | 2.677   | 0.00793 ** |
| age_group_5_40to44  | 0.21854  | 0.06983    | 3.129   | 0.00196 ** |
| age_group_5_45to49  | 0.11499  | 0.06648    | 1.730   | 0.08492 . |
| age_group_5_50to54  | 0.13542  | 0.06472    | 2.092   | 0.03743 * |
| age_group_5_55to59  | 0.18769  | 0.06426    | 2.921   | 0.00381 ** |
| age_group_5_60to64  | 0.09062  | 0.06404    | 1.415   | 0.15829 |
| age_group_5_65_over | NA       | NA         | NA      | NA      |

---

**Signif. codes:** 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## Residual standard error: 0.2422 on 249 degrees of freedom

(1 observation deleted due to missingness)

## Multiple R-squared: 0.1283, Adjusted R-squared: 0.08974

## F-statistic: 3.33 on 11 and 249 DF, p-value: 0.0002689