Impacts of Various Features on Starting Salary

Project Definition

The goal of this project was to see how various features impact a college graduate's starting salary upon entering the workforce. There are plenty of websites that do something akin to this - for example, Glassdoor and ZipRecruiter allow you to filter salaries by years of experience and/or by position title (such as entry level software engineer vs software engineer) - but I haven't seen any websites that look at the impact of the specific variables I was interested in when taken together.

The features I wanted to examine were as follows: major (specifically computer science, engineering, mathematics/statistics, physical sciences, and biology), degree type (bachelor's, masters, or doctorate), private vs public school attendance, school ranking, GPA, graduating with a double major, graduating with internship experience, and location. I wrote a more in-depth excerpt explaining my process for incorporating these features in the Jupyter Notebook I submitted, but I'll give a brief overview here as well.

Initially, I was planning on gathering data from the web, but after searching for quite a while, I came up empty-handed. The problem was this: I couldn't find any data sets incorporating all of the features that I wanted to examine, and even when I found data that incorporated one or two of the features I was curious about, all I could find were summaries rather than raw data (for example, I found a website showing the average salary for each major for the graduating class of 2023 - which I ultimately used for creating salary ranges for each major).

Ultimately, I decided to generate my own data for the project, but I didn't want the data to be completely random. What I ended up doing was this: I found average salary data for each of the majors I was interested in, and I created a range of possible starting salary values based on these averages. For each data entry, a major is chosen from the aforementioned list (there are a set number of entries for each major - 20000 by default), an initial starting salary is chosen randomly from the range of values specified for the chosen major, and then the other features I ended up using (degree type, school type, school ranking, internship experience, and location - note that GPA and double major were found to not affect starting salary) were generated randomly as well (with specific probability rates for each degree type and school type). From here, the initial starting salary value is multiplied by a "multiplier" value for each of the other features, which gives us an "adjusted" starting salary value. The idea is that this adjusted starting salary value will be a good representation of what a new grad with the given major, degree type, school type, school ranking, internship experience, and location values should expect to make in the real world. As a side note, if you're curious about how I came up with the starting salary ranges and "multiplier" values, I have a more in-depth overview at the top of the Jupyter Notebook file I submitted (which also includes links to research that I referenced).

After generating the data with the process above, I stored it all in a csv file, which I then fed into a SQLite database for querying. From here, I created a linear regression model and a

logarithmic regression model, both of which were trained on the dataset stored in the database (after removing outliers). Finally, I evaluated the models by looking at the weights of each feature, as well as statistical measures such as MSE, RMSE, and R^2. I also visualized the line-of-best-fit and residual plots for each model, and I created box-and-whisker plots for specific subsets of the dataset to give more insights about the impact of each feature.

Overall, I feel that this project is a great display of the material we learned in class, as it touches on just about every one of the main topics we've discussed. The entire project is written in Python, I made extensive use of libraries such as NumPy and Pandas, I used both numerical and categorical data, I stored data into a SQL database and made queries using SQLalchemy, I created and trained regression models, I evaluated these models with strategies that we learned in class, and I provided visuals using Matplotlib to give further insights about the data.

Novelty and Importance

I'm excited about this project because I think it's interesting to see how different variables can impact one's salary potential, and as I made clear in the section above, I haven't been able to find any reliable website that incorporates all of the features that I used. As a student who's graduating next year with a bachelor's in computer science, I found it particularly interesting to see how internship experience and location affected the salary values. I'm currently in the process of applying for internship positions, so I was curious to see some sort of quantifiable measure demonstrating the impact these experiences could have on my future.

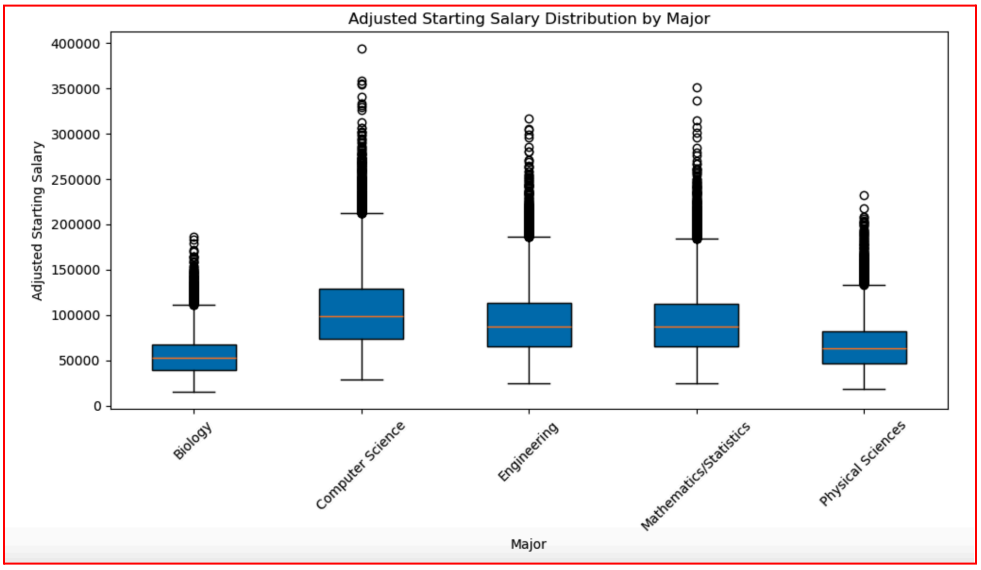
As for issues in current data management practices, I'd say the main issue I came across was the lack of availability of organized data. As I explained earlier, I couldn't find any datasets containing the features that I wanted to examine all taken together, even after using resources such as Kaggle (which attempts to solve this issue of data availability). I was able to find a few datasets containing a handful of the features I was interested in, but using an incomplete data set wouldn't have yielded any particularly accurate or useful results. In the same vein of discussion, I was not able to find any previous works that accomplished what my project aimed to do - only research showing the impact of each feature individually on starting salaries (which I used in order to come up with the multiplier values discussed in the project overview).

Experiments, Results, and Key Findings

I conducted a few "experiments" on the dataset, the first of which consisted of simple SQL queries.

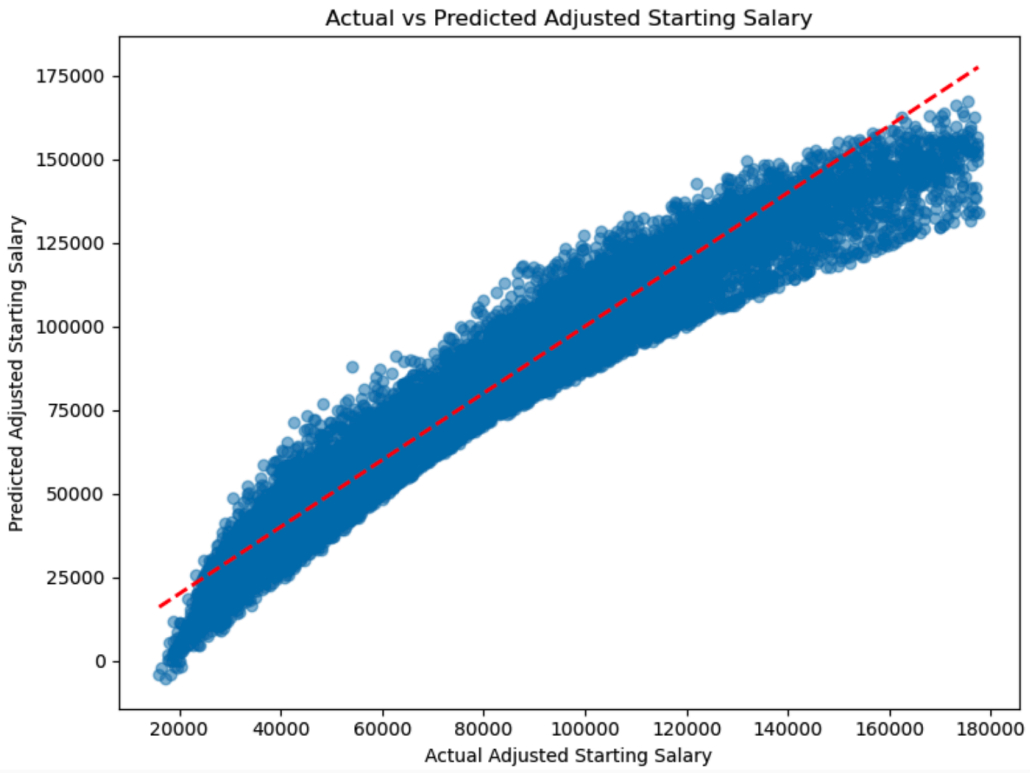
```
                    major  total_entries  avg_initial_salary  \
0                 Biology          20000         47844.55260
1        Computer Science          20000         91120.93885
2             Engineering          20000         79892.84810
3  Mathematics/Statistics          20000         79908.91915
4       Physical Sciences          20000         57833.71695

   avg_adjusted_salary
0          55264.90695
1         105278.04885
2          92526.05850
3          92267.91905
4          66715.82065
```

The above query, for instance, shows the number of entries in the database (total_entries), the average initial starting salary value (avg_initial_salary), and the average adjusted salary value (avg_adjusted_salary) for each major. As anticipated, the avg_initial_salary values are all quite accurate; every initial salary is within a few hundred dollars of the actual average salary values found in the research I referenced when deciding each major's salary range (for easy access, here is the link). On the contrary, the average adjusted salary values are inflated - anywhere from $8000 up to $14000. This suggests there are outliers skewing the data on the high end, which is confirmed when we look at the box-and-whisker box plots generated later in the code.



The above image shows the box and whisker plots for each major's adjusted salary values. As we can see, there are a good amount of outliers above the upper bound for each major (roughly 1 - 1.5% of the data points turned out to be outliers) - the most substantial of which are nearly 2x as high as the upper bound.

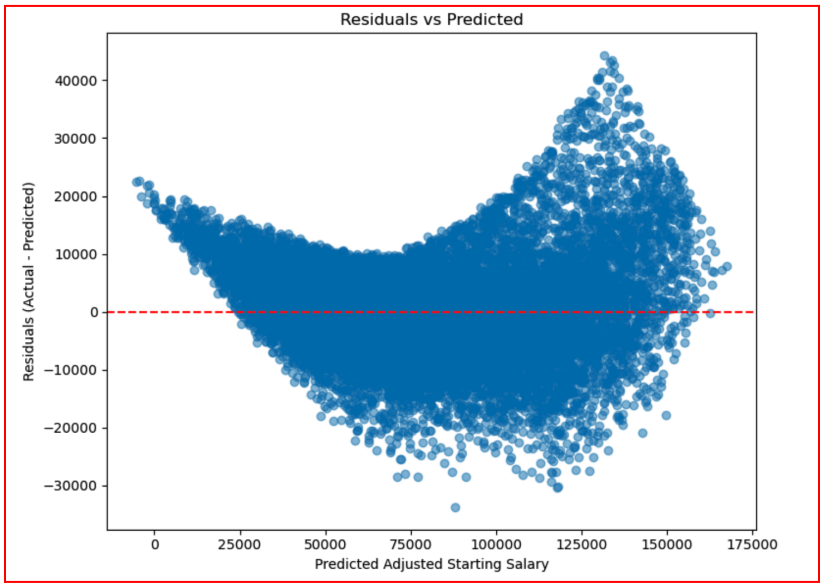
```
Overall Mean Squared Error (MSE): 69719993.43388921
Overall Root Mean Squared Error (RMSE): 8349.849904871897
Overall R2 Score: 0.936078732742795
```



After removing outliers using the IQR method (remove values above the Q3 + 1.5 * IQR and values below Q1 - 1.5 * IQR), I trained the linear regression model on the remaining data, which resulted in the above MSE, RMSE, R^2 values and the above line of best fit plot. Taken together, this shows that the model predicts an adjusted salary value with an average error of about $8349. This may seem fairly high, but the error becomes less significant the larger the salary is. For instance, for a datapoint whose actual adjusted salary is $200,000, an error of $8349 is only about a 4% error, but for a datapoint whose actual adjusted salary is $80,000 (roughly equal to the average adjusted salary across the entire dataset), an error of $8349 is about 10%. Additionally, the R^2 score suggests that about 93.6% of the variance in the dependent variable (adjusted salary) is explained by the model, which is a very strong r^2 score.

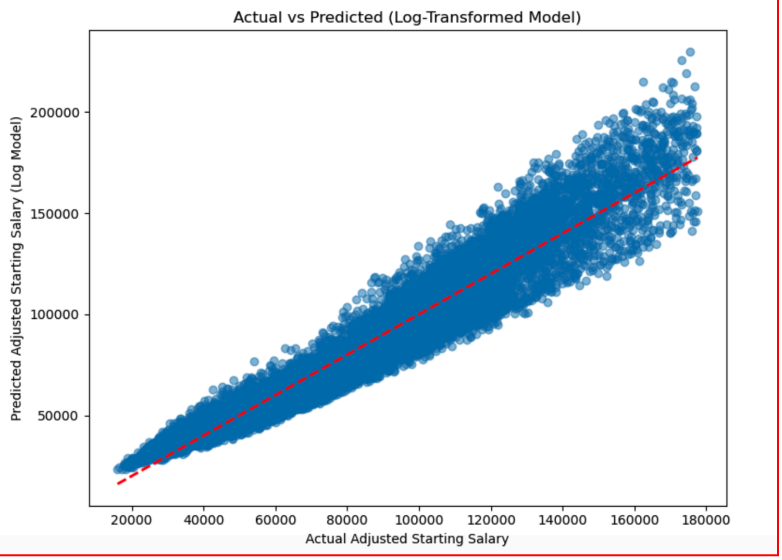
```
FEATURES RANKED BY IMPACT...
                         Feature   Coefficient
7           degree_type_doctorate  36170.061755
10  location_California Bay Area  21322.914885
8             degree_type_masters  19226.753373
12      location_NYC Greater Area   7995.643851
17       location_Northwest Urban   7843.939168
2           internship_experience   7291.196944
22       location_Southwest Urban   3681.081965
15       location_Northeast Urban   3625.759985
13        location_Non-Contiguous   3523.094649
20       location_Southeast Urban   3491.546958
9             school_type_private   2780.208677
3           major_Computer Science   1211.197151
4                major_Engineering    877.438668
5   major_Mathematics/Statistics    810.227581
6         major_Physical Sciences    470.279892
0         initial_starting_salary      1.094017
1                  school_ranking    -23.206412
16       location_Northwest Rural  -3914.150354
11         location_Midwest Rural  -4179.906805
21       location_Southwest Rural  -7191.908653
19       location_Southeast Rural  -7205.300804
14       location_Northeast Rural  -7383.128773
18                location_Plains  -7417.820246
```

I also made sure to inspect the magnitude of each of the features in terms of how much impact they have on the predicted adjusted salary value, and everything here lines up well with the multipliers. The only thing that sticks out to me is the fact that the school ranking value is negative and fairly minimal, which means that the school ranking value never increases the predicted salary value (even if the data point has a school ranked 1st in the nation). In hindsight, I think I should have made "school_ranking" a categorical variable divided into each of the main categories I defined with the multipliers (top 5%, upper quartile, midrange, lower quartile, and bottom 5%). This would have allowed the school rankings to have a more diverse impact on the predicted salary values, with a less linear relation.



The above image shows the residual plot for the linear regression mode. There is a notable curve in the plot, following a semi-quadratic pattern. To me, this implied a logarithmic model might be more accurate for the dataset, which is why I decided to develop one.

```
Overall Log-Model Mean Squared Error (MSE): 78874154.55354856
Overall Log-Model Root Mean Squared Error (RMSE): 8881.112236288232
Overall Log-Model R2 Score: 0.9276859382139183
```



Log-Model Metrics per Major:

	major	MSE	RMSE	R2
0	Mathematics/Statistics	7.350142e+07	8573.297069	0.927404
1	Physical Sciences	4.490744e+07	6701.301004	0.929293
2	Biology	5.181979e+07	7198.596637	0.886213
3	Engineering	7.436397e+07	8623.454810	0.928072
4	Computer Science	1.537554e+08	12399.815088	0.854944

To my surprise, the logarithmic model was less accurate by about $500 on average when compared to the linear model. However, for whatever reason, this model seemed to be better suited for approximating salary values for Physical Science and Biology majors in particular (as is shown above), but substantially worse for predicting salaries for CS majors.

```
{
    'major': 'Computer Science',
    'initial_starting_salary': 91411,
    'degree_type': 'bachelors',
    'school_type': 'public',
    'location': 'Northeast Urban',
    'internship_experience': 1,
    'school_ranking': 100
},
```

Overall, I'm under the impression that these adjusted salary values - as seen in the database and predicted by the above models (particularly the linear model), are a fairly accurate portrayal of how the features listed interact with one another. For example, I tried the above test case on the linear regression model (which should give a rough estimate for the average CS major's starting salary after graduating from a top 100 public school with internship experience and a bachelor's degree, as $91,411 was found to be the average CS major salary overall), and it predicted a salary of $116,906.60. I looked online to find a typical entry level software engineer salary in Jersey City (as this city is considered to be in the Northeast Urban region), and I found this statistic from Level.fyi, which asserts that the median salary in this area is $110,750. The main thing I would adjust if I were to update this project is that I would make "school_ranking" a categorical variable rather than an integer, as the impact of school ranking shouldn't be linear (as is seen in the overview in my Jupyter Notebook file).

Changes Made From Original Proposal

There weren't many changes I made to the project since submitting the proposal, but I did decide to exclude the GPA values and related minor / double major values. The reasoning for this (as explained in the overview of my Jupyter Notebook file) is that GPA and the presence of a double major doesn't seem to have a strong impact on starting salary from the research I looked into. That's not to say that these attributes don't impact one's salary at all - in fact, a higher GPA and graduating with a double major were both correlated with higher salary growth and more stable earnings over the course of one's career - but they didn't seem to directly affect starting salary in any consistent manner, so I didn't end up using them in the models I developed.