Modern Salary Modeling Project

Justin Mai, Troy Russo, Isaac Muhlestein, Conan Li, Jian Kang

Contents

---

1. Introduction (Justin)

Description: The job market can be a hard place to navigate, especially with the search of data roles in the recent years. As statistics students, many of us are leaning towards opportunities within data roles. To understand the recent market we will be analyzing data job logistics to investigate the factors and predictors that most impact the salary of these roles. Within this report, we will be using Salary Index data reported by real people in the industry to (1) What predictors are the best for optimizing our model to predict salaries. We will discover the factors and variables within the job description that may influence a person's job salary the most to help students like us navigate the market (optimize using the AIC). (2) Based off our optimized model, how well does this predict the actual job salaries in the dataset? We will do this by splitting our data into training (80%) and testing (20%) data to test the final model.

Disclaimer: We have pivoted from using this dataset https://www.kaggle.com/datasets/uom190346a/ai-powered-job-market-insights which is comprised of synthetic data based off the current job market regarding AI jobs to this dataset https://www.kaggle.com/datasets/murilozangari/jobs-and-salaries-in-data-field-2024/data which consists of real survey data from various people in data roles, reporting through this website https://aijobs.net/salaries/2024/. We decided to make this change because we believe that variables such as `experience_level` and `job_category` which can be found in our current dataset would be strong predictors for `salary`. We also believe that using real survey data as supposed to synthetic data would give us results that are more related to real-life circumstances, making the report more applicable for all.

---

2. Methods

2.1 Data Description (Justin)

The first dataset was collected through https://aijobs.net/salaries/2024/, it consists of 14199 different observations, with each observation representing a person in their role in 2024. The **response variable** we are measuring is `salary_in_usd` which measures a person's annual gross salary. The **8 predictors** are `experience_level`, `employment_type`, `job_title`, `employee_residence`, `work_setting`, `company_location`, `company_size`, `job_category`. All of these variables are categorical where `company_size` is categorized as S for small, M for medium, and L for large.

The second dataset consists of cost of living index by country where an index of 100 represents the living cost of NYC, United States, with all indices relative to that value. We will merge the two datasets by `country`. The predictors we're looking at in this dataset are Cost of Living Index, Rent Index, Cost of Living Plus Rent Index, and Local Purchasing Power Index. We believe that the cost of living could be indicative of `salary_usd`.

2.2 Data Processing (Justin and Jian)

The primary dataset will be comprised of the two datasets described in (2.1). We are joining the two datasets on `employee_residence` which is in form of country. Now each row will consist of a specified job description along with the cost indexes for each respective resident. Having all of these predictors in one dataset will allow us to utilize the lm() function to uncover linear trends for all predictor variables in response to `salary`. It will also allow us to compare models easily which we will do using ANOVA tests and by calculating the F-statistic. The primary dataset consists of 14199 observations after joining

Data Manipulation: Rows that consisted of NAs were in countries that weren't listed in the `cost_of_living` data, this demonstrates that their rank is low when ordering by index and there weren't a sufficient number of samples for those countries. Therefore we removed those observations (14161 observations). We also removed exact duplicate rows from the dataset (7575 observations)

Mutations in the data were also made to create new predictors `us_resident` which is a binary variable that denotes if the job is in the U.S. or not, and `experience_numeric` which turns `experience_level` into numerical values (i.e. 1 - "Entry-Level", 2 - Mid-level", 3 - "Senior", 4 - "Executive"), this transformation will support our use of linear modeling and allow us to easily check assumptions such as linearity assumptions. Also, because we have too many different job titles, we decided to aggregate these job titles by keywords into 8 categories (Data Scientist, Data Analyst, Machine Learning, Data Engineer, Leadership, Business Intelligence, Research, Other). This will consolidate our data and make linear models more interpretable

We are also reordering values to ensure that our **baseline term** is what we want it to be (i.e. releveling small companies to be the first type and entry-level jobs to be the first job types).

2.3 Model Diagnostics (Justin and Jian)

Linear Modeling Assumptions:

- **Linearity**: The relationship between the predictor and response is linear.
- **Independence**: All observations are independent of one another (pair-wise independence).
- **Homoscedasticity**: The variance of residuals is constant across predictor levels.
- **Residual Normality**: The residuals follow a normal distribution.

We know that the reported job descriptions are all independent of one another through the data description.

The model that we are using seems highly categorical, even after our data transformation process, which would result in very discrete predictions. To fix this issue we want to create a new interaction term and turn it into a predictor variable, adding a continuous predictor for `salary_usd`. We hypothesize that salary growth will vary by location, therefore, we are creating an interaction term between `experience_numeric` and the Cost of Living Index to test this (experience combined with living costs could impact salary growth differently).

The residuals when using a non-transformed model is skewed due to the deviations within the tails in our qq-plot, to fix this issue, we would have to use a **log-transformation** on the data. The variance of the residuals is also constant and there is a clear linear and positive relationship between the predictor and response.

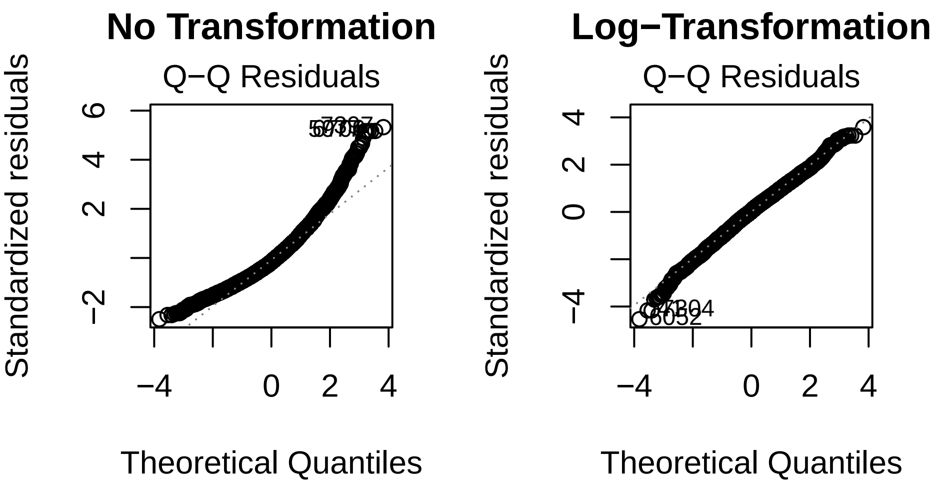


Figure 1: Normality check for Experience and Cost of Living Index Interaction variable on Salary, comparison between no transformation and log-transformation

To test the assumptions for our categorical variables we must look at the average salary by category, to see if there is a clear trend between the different experience levels in respective company size. This checks off the linearity assumption because there is a (somewhat) linear increasing trend for all the experience levels. We may explore an interaction term between `company_size` and `experience_level` because it seems like the trend for each `company_size` is increasing but at a different rate.

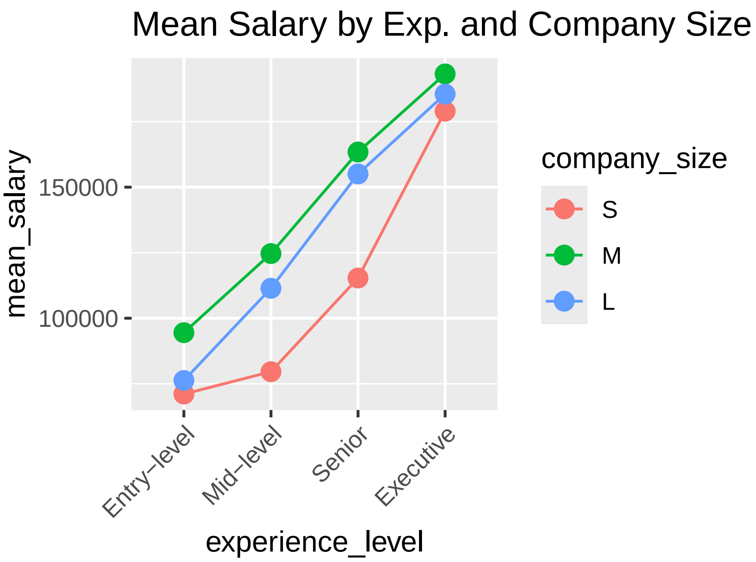


Figure 2: Interaction between Experience Level and Company Size on Salary

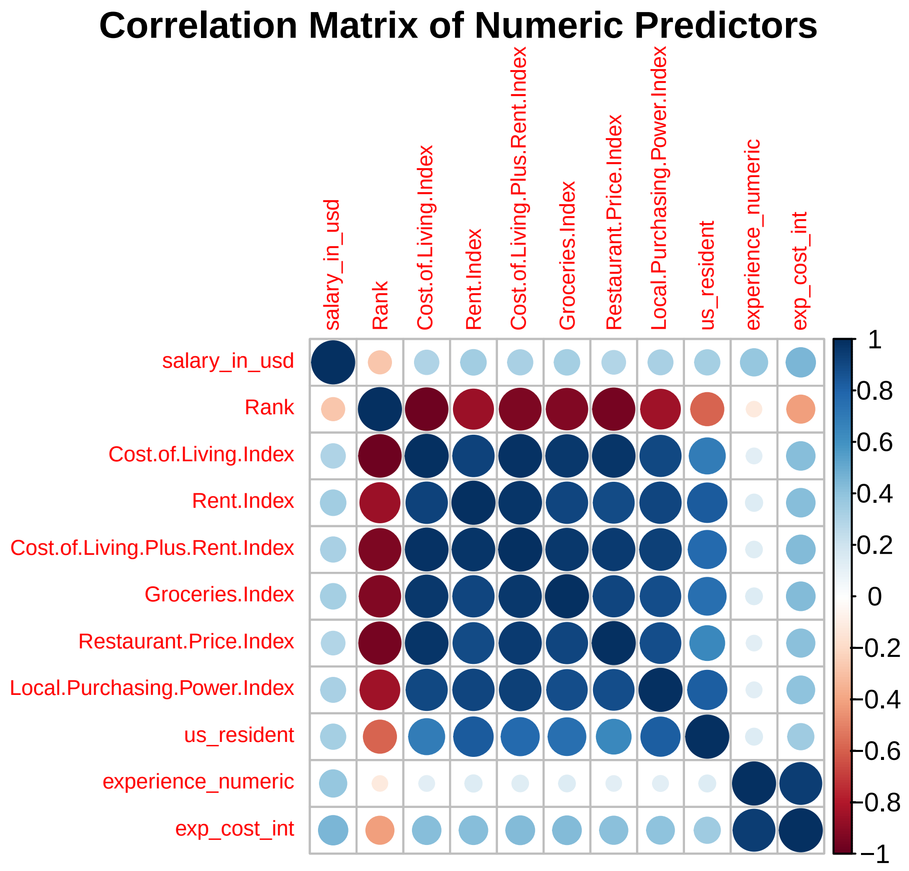
Correlation and Multicollinearity Analysis (Isaac)

One of the assumptions we are making in our model is that the predictors we are using are independent. Thus, it becomes expedient to test the multicollinearity of our predictors, to ensure that they are each independent of one another and add new information. Because a large majority of our predictors are categorical, we can use The VIF (Variance Inflation Factor) to test the multicollinearity of those:

```
##                      GVIF Df GVIF^(1/(2*Df))
## experience_level 1.209342  3        1.032187
## company_location 1.448979 62        1.002995
## company_size     1.302088  2        1.068218
## job_category     1.170496  7        1.011308
```

With the exception of `company_location`, all of these predictors have a VIF value that is lower than 5, so we can safely say that there are no issues of multicollinearity among them. Even `company_location` has a VIF of 6.527, so while it is above the baseline of 5 that we would like to see, it is not egregious. This implies that each of these predictors adds some new information to the model and thus explains (or doesn't) its own variation, which means that we need not fear that coefficients become over or underinflated, explaining variation that should be attributed to another predictor, or having that happen to them.

We are joining the cost of living dataset, which includes a number of variables that are related to cost of living metrics. We suspect that they might correlate together, being measures of the same changes in price and economy, so we can create a correlation matrix to examine any multicollinearity that may exist:



Examining our correlation matrix, we can see that the index variables are highly correlated with one another since the cost of living indices are joined by country, so there are several repeated values. To satisfy the multicollinearity assumption, we will be removing many of these highly correlated variables (only using `Cost.of.Living.Index`) for our MLR model.

From our data processing and model diagnostics step, to avoid repeating variables, satisfy the assumptions necessary for conducting a linear model, and to consolidate the number of predictors we are using. Variables like `employement_type` and `job_title` are removed because they are highly correlated with new variables we mutated or existing variables. We are also removing `company_location` and `employee_residence` to simplify our model since there are a lot of different categories and most of them aren't significant to model at $\alpha = 0.05$, consolidating our model down. We'll be using a dataframe with the following variables: experience_level, salary_in_usd, work_setting, company_size, job_category, Rank, Cost.of.Living.Index, us_resident, experience_numeric, exp_cost_int.

2.4 Model Selection (Troy and Conan)

To determine our model, we will fit multiple linear regression models and find the model that minimizes our **AIC (Akaike Information Criterion)**. We favor the model selected by lowest AIC because BIC penalizes models with a large number of observations and tends to predict less than the AIC.

We can use a step function from the MASS package that computes the best model purely based on the AIC by comparing every potential model combination and returning the model with the lowest AIC. This model may be overfit however, since AIC does not account for model complexity.

```
# Fit the full model with all predictors
full_model <- lm(salary_in_usd ~ . - salary_in_usd, data = model_df)
# Perform stepwise selection using AIC (both directions)
step_model <- stepAIC(full_model, direction = "both", trace = FALSE)
```

Now that we have the model that we have found from AIC we can test the assumptions of the model as well as performing cross validation to test the valididity of the model. We can also perform an anova test to see if our predictors are significant.

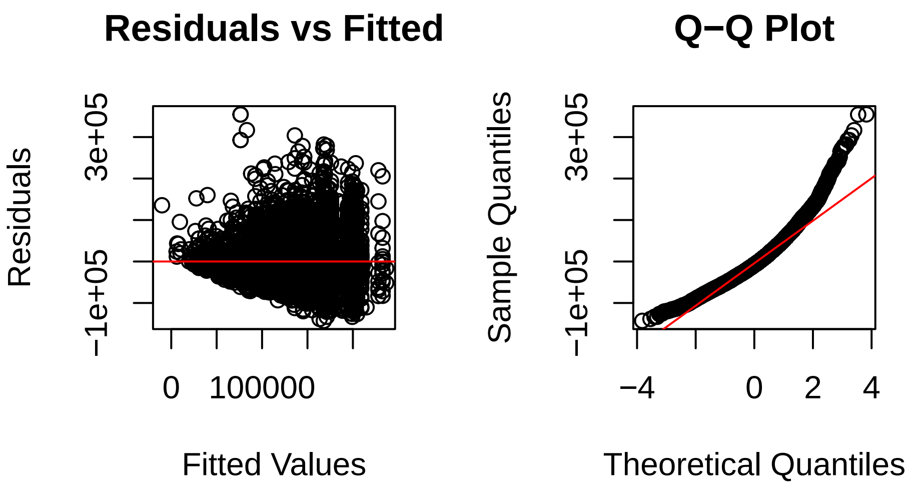


Table 1: 10-fold Cross-Validation Results

RMSE	Rsquared	MAE
54993.62	0.3514908	42142.88

Table 2: anova test

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
experience_level	3	5.358133e+12	1.786044e+12	557.587621	0.0000000
work_setting	2	4.298576e+11	2.149288e+11	67.098909	0.0000000
company_size	2	1.245853e+11	6.229263e+10	19.447219	0.0000000
job_category	7	2.455925e+12	3.508464e+11	109.531198	0.0000000
Rank	1	1.614384e+12	1.614384e+12	503.996786	0.0000000
Cost.of.Living.Index	1	7.728722e+11	7.728722e+11	241.284013	0.0000000
us_resident	1	2.787853e+11	2.787853e+11	87.034357	0.0000000
exp_cost_int	1	1.923540e+10	1.923540e+10	6.005125	0.0142869
Residuals	7556	2.420311e+13	3.203164e+09	NA	NA

The cross validation tells us that our model is not over fitting since the R^2 is only about 35%, but also suggests that we can improve our model because the RMSE is quite high at 54993, so we may need to add more interactions or do some non linear transformations of the data. Additionally, the anova test suggests that our predictors are significant.

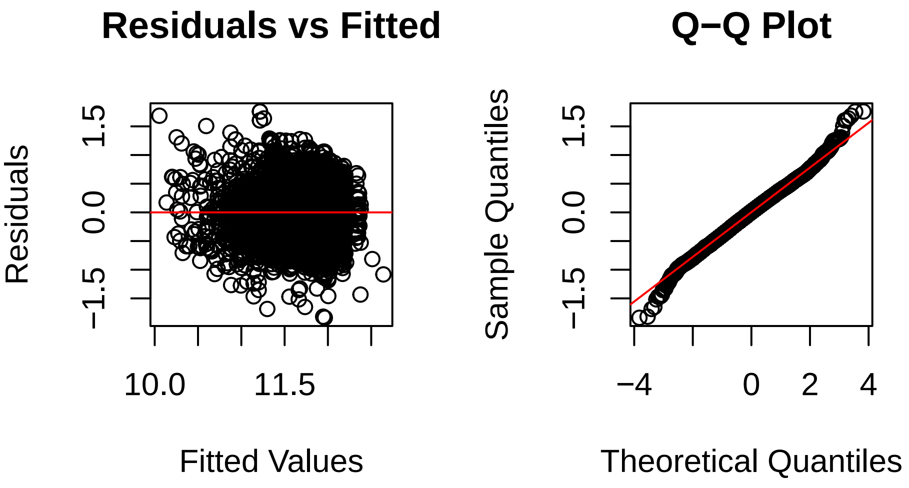
The residual plot shows a funnel shape, which suggests hetereoscadascity of the variance and the residuals do not appear to be centered at zero, which indicates that there is bias. The plots indicate that we may want to use the log of the salary instead of salary as our response.

We can now run the step AIC function again using the log(salary) as our response and perform the same model diagnostic tests as above. We also add an interaction term between experience_level and company size.

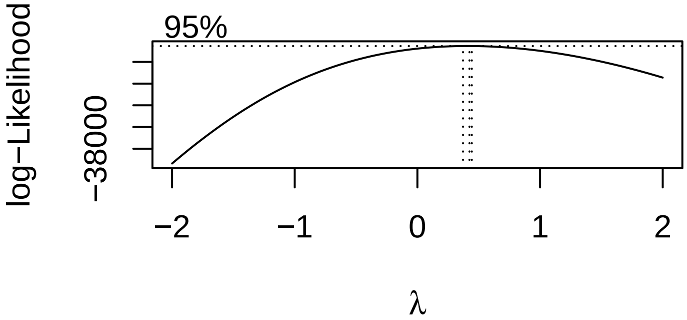
```
## Call:

## lm(formula = log_salary ~ experience_level + work_setting + company_size +
##     job_category + Rank + Cost.of.Living.Index + us_resident +
##     exp_cost_int + experience_level:company_size, data = model_df)

## Multiple R-squared: 0.4136208
```

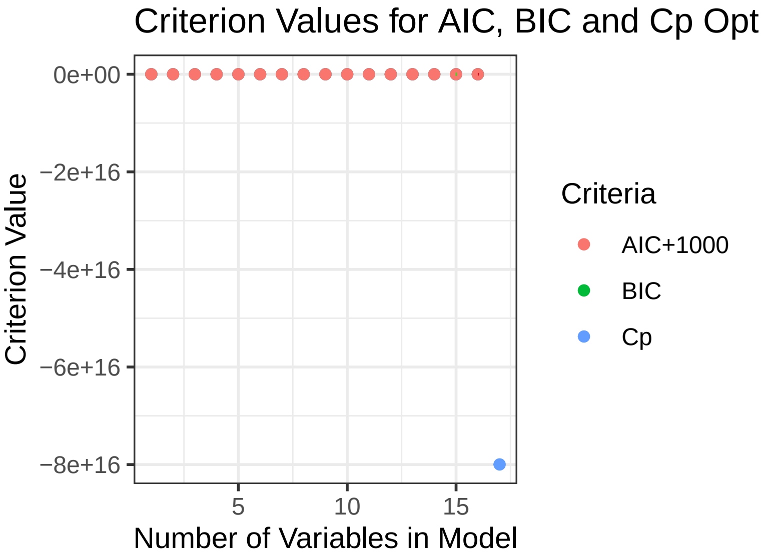


Our **log-transformed model** achieves a notably higher R^2, indicating it explains more variability in $\log(\text{salary})$ than the untransformed model. However, the residual plot still exhibits heteroscedasticity, suggesting that further transformations or alternative modeling approaches may be necessary.



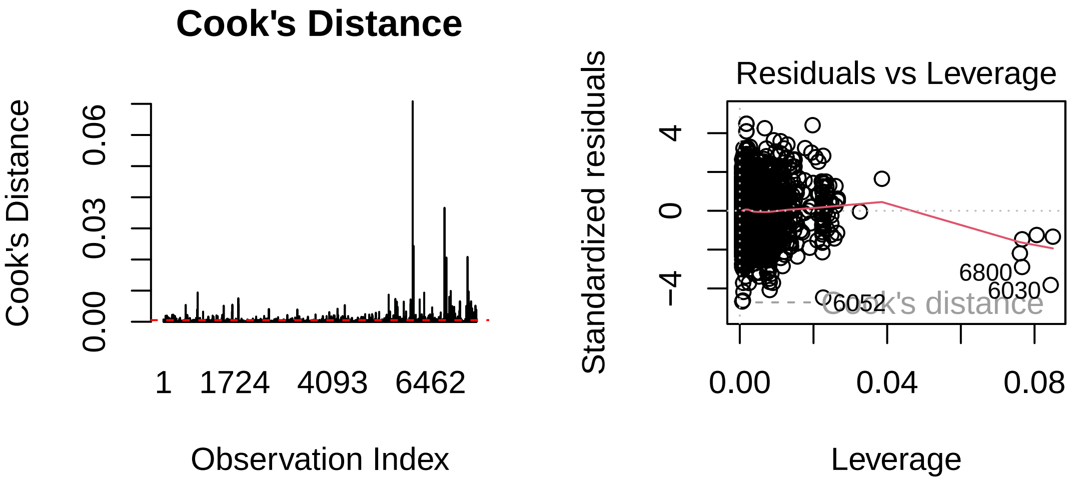
The box cox indicates that the log transformed model is the best power model for our data, so the changes we need to make will be on individual predictors and not the entire data.

Model Selection Variation (Isaac) One way to verify that this model is the optimal choice of predictors it to explore which predictors would be selected by the BIC and Cp criteria, and analyze any differences that exist.



The BIC criterion minimizes at 15 predictors, while the Cp and AIC criteria minimize at 16 predictors. We expect the BIC to minimize at a lower amount of predictors than the rest, but the fact that it is so close, alongside the fact that the Cp criteria minimizes at the same point as our AIC model, is a good indication that our model is optimal.

Cook's Distance Analysis (Isaac) We can also test to make sure that there are no high leverage points, that might be skewing the values of the coefficients. To do this, we can calculate the Cook's Distance on this model, and create a plot that visualizes these distances for each point.



Considering we have 7575 observations, a good threshold is that Cook's distance values should be lower than $\frac{4}{7575} \approx 0.00053$. Visibly, we can see that there are definitely some points that exceed this threshold, especially through the residual vs leverage plot there are some data points that apparently have high leverage. When plugging this into R, we get that 457 observations have significantly high values. This suggests that a non-trivial section of the data set is having a large impact on the regression model. We can take a quick look at the points associated with these highest values to make sure there is no mismeasurement going on.

```
##      salary_in_usd     job_category Rank Cost.of.Living.Index us_resident
## 6845        125404     Data Analyst   85                 31.7           0
## 7357        120402   Data Scientist    1                101.1           0
## 6052         15000     Data Analyst   86                 31.4           0
## 6800        104697 Machine Learning    1                101.1           0
## 6030         56536 Machine Learning    1                101.1           0
```

Upon further analysis, it appears that the `Rank` predictor significantly increases the Cook's distance of high-leverage points. Removing this predictor from our model, we decrease the Cook's distances across the board, resulting in 84 fewer high-leverage points. While it is not ideal that we still have a significant number of high-leverage points, we are working with a very large data set, so it isn't too concerning.

2.5 Final Model Description (Troy, Conan, Isaac)

After comparing multiple candidate models, our final chosen model regresses **log(salary_in_usd)** on several predictors:

$$\begin{aligned}\log(\text{salary\_in\_usd}) = \beta_0 &+ \beta_1 \cdot \text{experience\_level} \\ &+ \beta_2 \cdot \text{job\_category} + \beta_3 \cdot \text{us\_resident} \\ &+ \beta_4 \cdot \text{work\_setting} + \beta_5 \cdot \text{company\_size} \\ &+ \beta_6 \cdot (\text{experience\_level} \times \text{company\_size}) \\ &+ \beta_7 \cdot \text{exp\_cost\_int}\end{aligned}$$

1. Log Transformation

We transformed the salary to $\log(\text{salary\_usd})$ to address heteroscedasticity and non-normal residuals discovered in the initial model. Interpreting coefficients on the log-scale means that each unit increase in a predictor corresponds to a *multiplicative* change in salary, rather than an additive one. For example: - If $\beta_1 = 0.10$ for a given predictor X, then a **1-unit** increase in X is associated with about a **10.5%** increase in salary $(e^{0.10} - 1 \approx 0.105)$.

2. Significant Predictors

Based on the final model summary (not shown here in detail), we typically see the following patterns:

- **experience_level**: Highly significant, with more senior roles (e.g., "Senior," "Executive") associated with higher $\log(\text{salary})$. This implies strong upward salary trends as experience grows.
- **company_size**: Medium or large companies might pay more, on average, than small companies—but the magnitude and significance can vary by the data.
- **job_category**: Certain roles (e.g., "Data Scientist," "Machine Learning") often command higher salaries relative to baseline roles.
- **us_resident**: Countries or regions with different cost-of-living indices can have substantially different salary norms (US vs non-US).
- **work_setting**: If "Remote" vs. "In-Office" or other setups are included, each may show a different average salary level.
- **exp_cost_int**: People with varying experiences will likely have a different cost of living rate.

3. Interaction: `experience_level : company_size`

We included an interaction between **experience_level** and **company_size** to test whether the *effect* of experience on salary depends on the size of the company. Interpreting an interaction on the log-scale:

- The coefficient of `(Senior : Large)` tells us the additional log-salary *beyond* simply adding the main effects of "Senior" and "Large" separately.

3 Testing and Results (Troy)

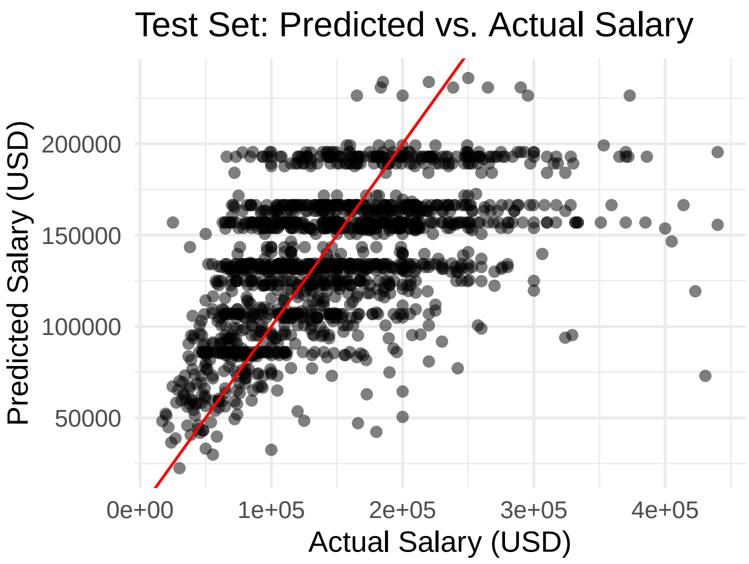
```
## [1] 0.4008326
```

```
## [1] 59804.53
```

Table 3: Root Mean Squared Error (RMSE) Results

Metric	Value
Log-Scale RMSE	0.4008
Original Salary RMSE	59,804.53

Our final model exhibits an RMSE of approximately 0.4 in log-scale, which translates to substantial error when converted back to the original salary scale. Specifically, the RMSE in USD is $59,761.05, which is about 0.4 times the median salary in our dataset. While this indicates that our model captures general salary trends, the remaining variability suggests inherent limitations in the available data, such as unobserved factors influencing salary that our model cannot account for.



The scatter plot of predicted salaries (y-axis) versus actual salaries (x-axis) shows that while our model captures the overall upward trend (points cluster around the diagonal), there is still considerable spread—particularly at higher salary levels. This indicates that although many predictions are reasonable, significant variability remains, suggesting unmodeled factors or inherent randomness in the data.

4. Discussion

4.1 Limitations (Justin)

The primary limitation to our study is that our data comes from self-reported surveys which could possibly hinder the accuracy of true data science and AI job salaries and resulted in many duplicates. Recognizing the many duplicates, we've removed exact duplicate observations, ensuring that each row is unique because its unlikely that two people have the exact same job description. We are also consolidating our data to US based workers vs. non-US based workers. This leads to issues because there are more US based observations in our dataset but there are still a significant number of reports outside of the US. Although almost all of the categorical predictors regarding `company_location` were considered insignificant to our model, there were still a few that were significant that we removed, which may cause slight inaccuracies to our prediction. Lastly, our Cost of Living and Job Salaries data are country-based, which isn't ideal for predicting salary because within each country, there are cities or provinces that pay higher salaries than others (NYC, San Francisco, and Seattle in the U.S.). This made our model one-dimensional since we aren't able to access which city each observation is from.

4.2 Conclusion (Conan)

In this study, we explored some key factors that influence salaries in data science related jobs using real world data. Through extensive data processing, model selection and log transformation, we were able to develop a model that improves the normality and heteroscedasticity in predicting salaries. The analysis focused on multiple predictors, as we found out **experience_level**, **company_size**, **job_category**, **us_resident**, **exp_cost_int** and **work_setting** are the most significant in predicting **salary**. We also added an interaction term between **experience_level** and **company_size** to see if the impact company size has on salary varies significantly based on experience. The relatively high RMSE from our model suggests that other unobserved factors may also play a significant role in predicting salaries. Given the limitations, such as self-reported survey and locations, they suggest further research on additional predictors that may potentially have a significant impact on salaries. Overall, our model provides close insights into the salary trend in data science related job market that may help people searching for appropriate positions and making career decisions, and future exploration on location and cost of living may identify more complex relationships that will improve model accuracy.