Annie Jiang (Team AJ)
CIS/STA 3920
May 14, 2024

Data Science Job Salaries Final Report

Introduction

The aim of this project is to analyze a dataset containing the salaries of jobs in the data science field collected from 2020 to 2022, obtained from Kaggle. There are 607 observations and 11 variables in the dataset, which include:

- *work_year* (integer): the year the salary was paid
- *experience_level* (string): the experience level in the job during the year
- *employment_type* (string): the type of employment for the role
- *job_title* (string): the role worked in during the year
- *salary* (integer): the total gross salary amount paid
- *salary_currency* (string): the currency of the salary paid
- *salary_in_usd* (integer): the salary in USD
- *employee_residence* (string): employee's primary country of residence during the work year
- *remote_ratio* (integer): the overall amount of work done remotely
- *company_location* (string): the country of the employer's main office or contracting branch
- *company_size* (string): the average number of people that worked for the company during the year

With this dataset, I plan to use predictive modeling techniques such as linear regression, stepwise selection, regression tree, and random forest to predict *salary_in_usd* based on the other variables. The questions I intend to address with my analysis are:

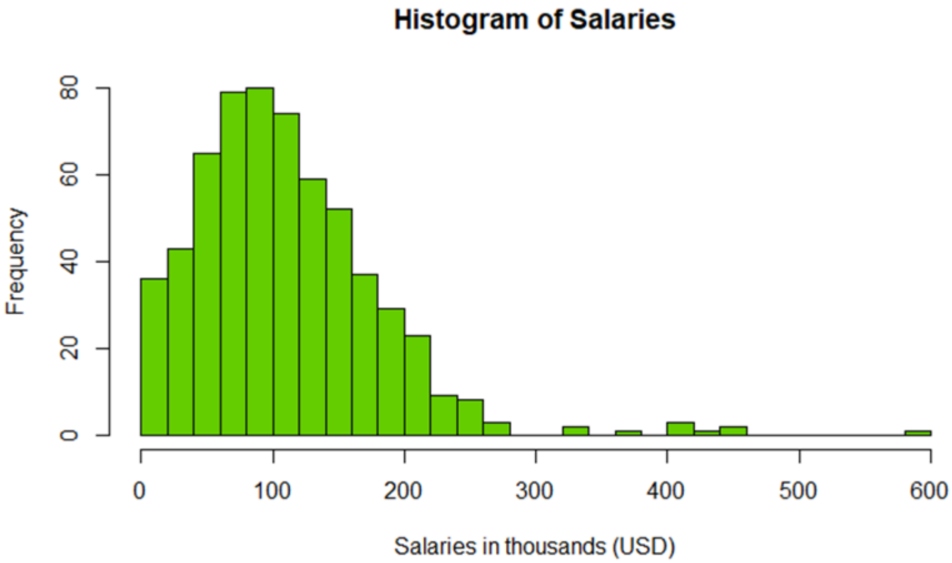
- Can we predict salary based on variables such as experience level, job title, employment type, location, and company size?
- Which variables influence salary the most?

My target audience for this analysis could be job seekers interested in understanding salary expectations based on several different factors such as role, location, and experience level. This analysis could also be helpful to employers that want insights into factors influencing compensation decisions so that they may compensate data science roles appropriately.

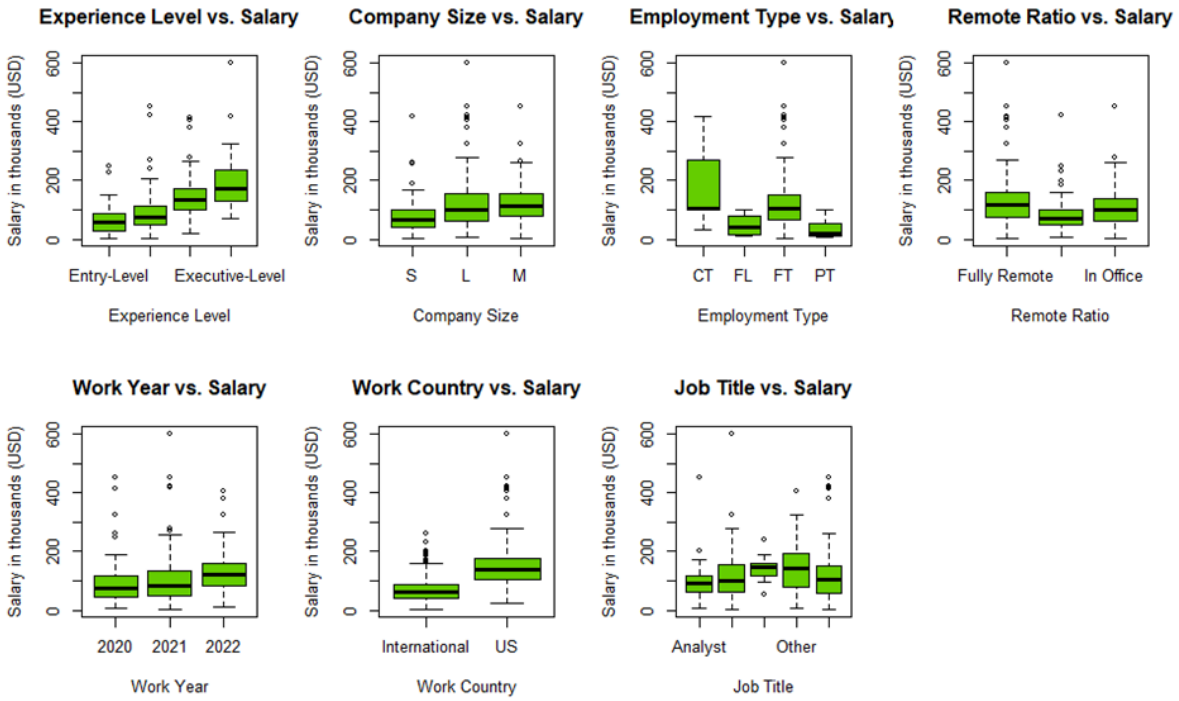
Data Processing and Transformation

There are several steps I took to make sure that the data was ready for analysis. First, I checked for null values, and luckily there were none. Secondly, I converted *salaries_in_usd* to thousands, so that future calculations and plots would be easier to understand. Next, I noticed that the values in the column *work_year* were not continuous, despite it being an integer data type, since the only values were 2020, 2021, and 2022. Therefore, I changed the data type of the column to integer to factor so that it would be treated as a categorical variable rather than a continuous one. For the column *remote_ratio*, which only contained the values 0, 50, and 100, I opted to create a new column called *work_mode* that renamed the values in the former column. Thus, 0 in the *remote_ratio* column would be renamed to In Office in the *work_mode* column, 50 would correspond to Hybrid, and 100 would mean Fully Remote. Moreover, since there were too many differing job titles in the *job_title* column, I assigned each job title into a category based on key words in the title so that the analysis would be more accurate and easier to understand. There are five categories that I assigned each job to: Scientist, Analyst, Engineer, Manager, and Other, and I created a new column called *job_type* to store this information. In addition, I simplified the column *employee_residence*, which contained over 50 different countries, by only differentiating between those who worked in the United States and those who worked internationally and storing this in a new column called *work_country*. Finally, I dropped the irrelevant columns, so that my dataset contained 8 columns (*work_year, experience_level, employment_type, job_type, salary_in_usd, work_country, work_mode, company_size*) compared to the original 11. Now that the data has been transformed, this makes it easier to perform analysis on the data.

Exploratory Data Analysis



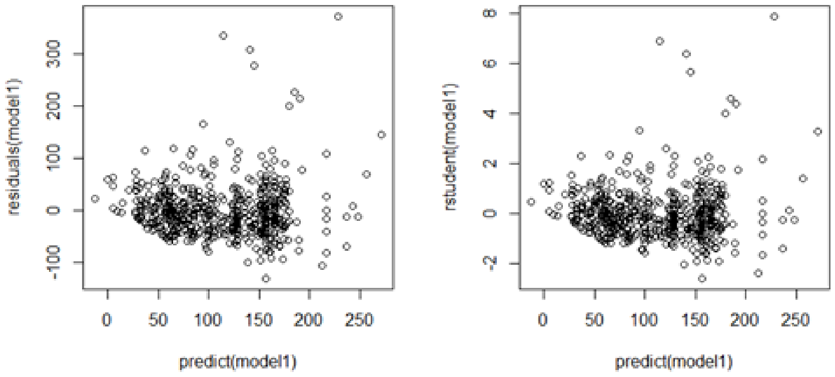
To start the exploratory data analysis, I created a histogram using salaries as the x-axis, as shown above. From this, we can see that the most common salaries range between 60K-120K USD. The data is also highly skewed to the right, which means that there are more observations on the lower end of the spectrum and fewer observations with higher salaries. This could be expected, since most employees are not in the highest-paying roles. Also, there are very few salaries exceeding 300K, which means that they are outliers. There are also several bins between 300K and 600K that are almost empty, which shows how rare these higher salaries are.



I also created box plots to see the relationship between each variable and salary. At a glance, we can see how some variables may be better predictors of salary than others. For example, the experience level vs. salaries box plot shows how as the experience level of the role rises, the median salary tends to increase as well, and there is also more variability in salary as we go higher up in level. In addition, there's a lot of variability in the work country and job title plots that might make them good predictors as well. But on the other hand, variables such as work year and company size may not be as good at predicting salary since their box plots show less variability.

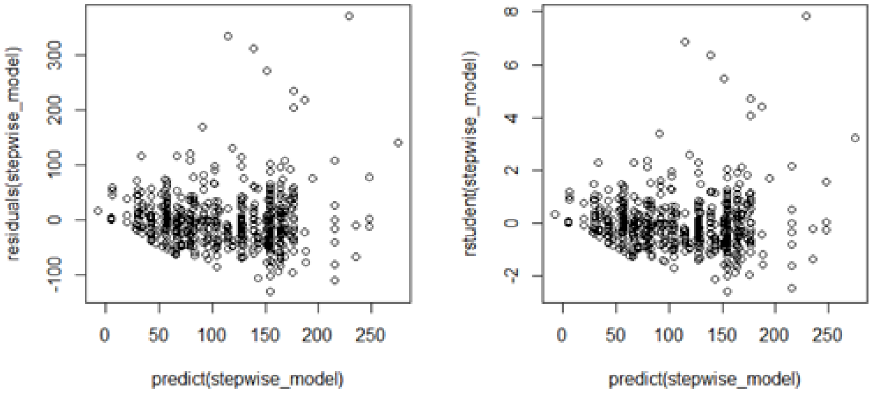
Linear Regression

Now onto the development of the predictive models, first I created a linear regression model using all 7 variables to predict salary. After I ran the model in R and looked at the summary, I saw that there were several insignificant predictors such as *work_year* and *work_mode*, based on the p-values. The residual standard error, which measures a lack of fit, was 51.07, and the adjusted-R squared was 0.4836. Using 10-fold cross-validation, the MSE turned out to be 2656.82. I also created the residual and studentized plots, as shown below. Looking at the studentized plot, we can see that there are many outliers that exceed the y-value of 3.

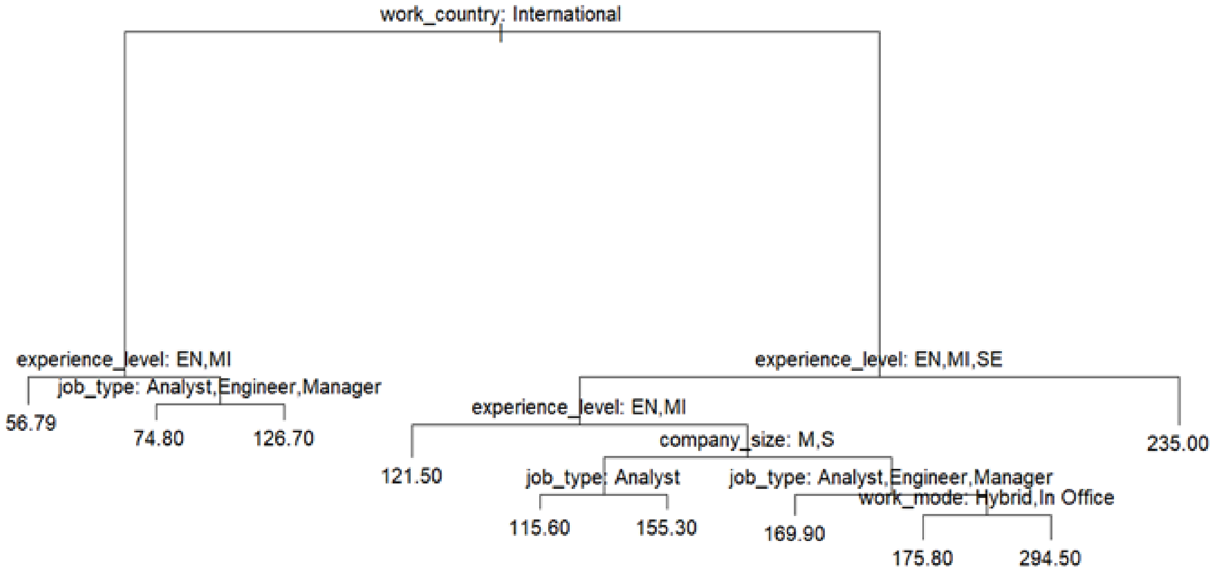


Linear Regression with Backward Stepwise Selection

After creating the initial linear regression model with all the variables, I decided to use backward stepwise selection to find a more optimal model. Backward stepwise selection is where all the candidate variables are included in the initial regression model, and variables are removed as they are identified to be insignificant to the model. This is done until no more variables can be removed. When I ran the stepwise selection, I ended up with only the five variables *experience_level*, *employment_type*, *company_size*, *work_country,* and *job_type* in the final model, which means that the variables *work_year* and *work_mode* were excluded. This corroborates with the previous model's findings, since we saw that *work_year* and *work_mode* were insignificant in the initial linear regression model. Now, all the remaining variables are significant to salary. The residual standard error of the new model is 51.03, which is slightly lower than the first model's, and the adjusted r-squared is 0.4844, which is slightly higher than the first model's and that is preferable. The MSE after 10-fold cross-validation was 2646.83, lower than the first model's, meaning that this model has a better performance. Still, the residual plots have not changed very drastically in shape, and there are still outliers.



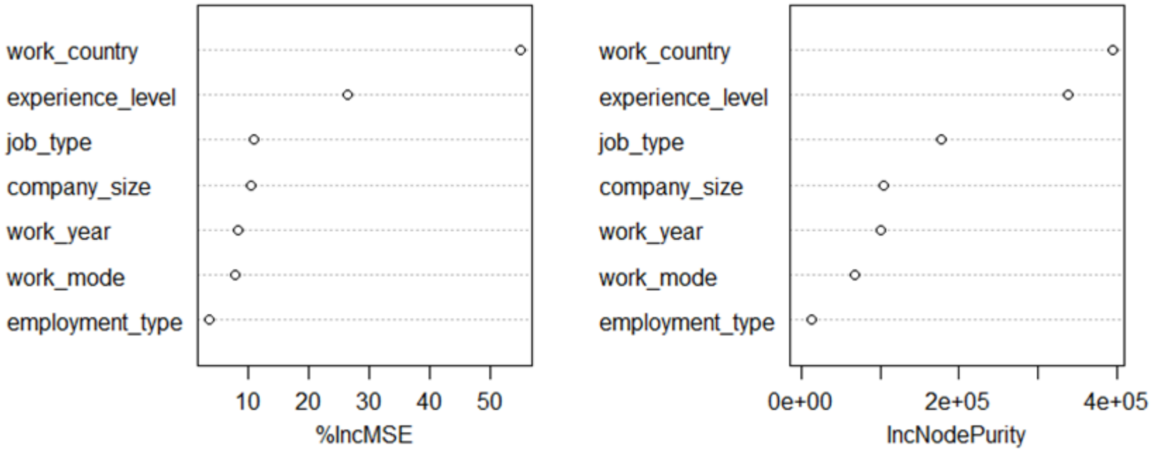
Regression Tree


work_country: International
experience_level: EN,MI
job_type: Analyst,Engineer,Manager
56.79
74.80
126.70
experience_level: EN,MI,SE
experience_level: EN,MI
121.50
company_size: M,S
job_type: Analyst
115.60
155.30
job_type: Analyst,Engineer,Manager
169.90
work_mode: Hybrid,In Office
175.80
294.50
235.00

After creating linear regression models, I wanted to create a regression tree to predict salary. Regression trees are easier to interpret and visualize compared to linear regression models. They are also relatively robust to outliers, which is good since a lot of them are present in the data. When I created the regression tree, I used a 50/50 hold out so that 50% of the data would be used for training and the other 50% would be used for testing. I also pruned the tree to a size of 10, and this was because the value contained the lowest cross validation error rate. Pruning is a technique that removes sections of the decision tree to reduce its complexity and prevent overfitting. As we can see from the resulting tree, there are 10 leaf nodes that arrive at a decision output. Also, since work_country is the root node, we can conclude that it is the most important factor in determining salary in the regression tree. Finally, the MSE of the regression tree is 2609.62, which is lower than both linear regression models in the earlier slides.

Random Forest

For my final model, I decided to build a random forest. Random forest basically builds multiple decision trees during training and merges them to improve the overall performance and robustness compared to a single decision tree. Bagging is included in the process since each tree is trained on a different bootstrap sample of the data, which is created by sampling with replacement. To start, I used bagging with all 7 variables, which returned an MSE of 2767.48. Then, I implemented random forest with 3 variables, which I got from dividing the total number of predictors, which is 7, by 3 since this is a regression problem, and rounding up the result. This returned an MSE of 2518.18, which is the lowest out of all the models I built and is therefore the best performing model. I plotted the importance measures, and from them we can see the importance of each variable in the increase in MSE and the total decrease in node impurity from splits involving the variable. From this, we see that *work_country* and *experience_level* are most important in determining salary.



Conclusion

To conclude, here are the rankings for the models I built based on their MSE: first is the random forest model with an MSE of 2518.18, second is the regression tree with an MSE of 2609.62, third is the linear regression model with backward stepwise selection with an MSE of 2646.83, and last is the initial linear regression model with all variables included with an MSE of 2656.82. To address the questions asked earlier, if we wanted to predict salary based on factors such as experience level, job title, employment type, location, and company size, then random forest would be the most optimal way to do so. In addition, the variables that influence salary the most are work country and experience level, based off the importance plots obtained from random forest.