CS 439: Intro to Data Science - Project Proposal

Predicting Data Science Salaries: Understanding Compensation Factors

Student Name: Ashwinder Bhupal
Date: November 13, 2025

Define Project

The objective of this project is to forecast the salary of data science according to occupation and organizational aspects such as level of experience, job position, working type and how the job is done remotely, the size of the organization, and geographical location. The job market has a lot of information asymmetry, information asymmetry where job seekers do not know what is good to compensate them based on their qualifications, and employers have difficulties when setting competitive pay. This makes the market inefficient where negotiations are more related to the access of information than the true value.

The strategic part is the knowledge of what has the most influence on compensation. Location is no longer as important now that remote work has become a reality after the pandemic. What is the degree of impact on pay between experience level and company size? Using a prediction model, I will be able to measure the influence of each of these factors and offer evidence-based career advice and recruitment.

The project also has direct applications in concepts we have learned in class. It adheres to the entire data science chain, including data gathering to model implementation. The phase of data cleaning is based on the strategies that we studied to work with missing data and duplication. Exploratory analysis uses the visualization techniques and the statistical techniques we have discussed to understand the patterns. The modeling applies the techniques of regression that we discussed, both simple linear models and ensemble techniques. The concept of feature engineering is interconnected with our talks on making meaningful variables out of raw data and evaluation is done according to the principles we learned measuring the accuracy using the MAE, RMSE and R 2 score.

Novelty and Importance

I am enthusiastic about this project as it is actually a real-life issue - getting to know how much to expect to receive as a (data scientist) applicant. The majority of online sources are more general in their coverage or offer average values without considering particular combinations of factors. The work in this project is based on machine learning to create personalized salary estimates using various features.

The existing practices in the current salary datas are severely lacking. Such websites as Glassdoor or LinkedIn indicate simple averages but do not show the interaction of various factors. They may indicate that an average Data Scientist earns $100k, but do not specify how the size of the company, the distance of work, or the location alters this number. Different niche jobs such as Data Engineer, ML Engineer, Data Analyst, Research Scientist have also emerged and compensation disparities cannot be comprehended without complex analysis.

There has been previous research on salary prediction largely on more general job sites or in scholarly research of wide tech salaries, often with relatively basic statistical techniques. This project is particular to the data science roles, yet the comparison of the prediction accuracy is carried out using several machine learning algorithms. I would like to make correct predictions as well as know what features are important, what matters most when compensating.

Plan

Data: The dataset I am working with is an example of a Kaggle dataset that contains 1,000 or more data science salary data samples in 2020-2023 in various nations. All the records contain such information as job title, experience level (entry/mid/senior/executive), employment type, wage in USD, a percentage of remote work (0%/50%/100%), locations, and the number of employees. The data is in the form of a CSV file which I will load into Jupyter notebook under pandas.

Models and Techniques: I will use and compare six regression models. Linear Regression is a standard to which its simplicity and interpretability are attributed. Ridge and Lasso introduce regularization to avoid overfitting. The Non-linear relationships are captured in decision trees, and random forests, and gradient boosting - the ensemble models tend to work quite well on tabular data. I will apply scikit-learn to apply uniform implementations.

In the case of feature engineering, I will label-encode ordinal variables such as experience level and generate derived features: binary values of full remote work and location matching, grouped job variables as there are 150+ unique titles. On numerical features, I will use StandardScaler.

Implementation: First, I will examine the data structure and confirm the quality problems - missing values, duplicates, and outliers. Second, I will tidy by working with missing data (median in numbers, mode in categories), duplicates, and outliers. Third, the visualization of salary distributions using histograms, box products by experience and company size, and correlation matrices will be done by exploratory analysis. Fourth, I will prepare data, encode categories, scale features and divide into 80% training and 20% test sets. Fifth, I will be training all models with the help of hyperparameter tuning with GridSearchCV. Sixth, I will assess and compare performance.

Evaluation: I'll use three metrics: MAE (target <$15,000) shows average prediction error, RMSE (target <$20,000) penalizes large errors, and R^2 (target >0.75) shows explained variance. Five-fold cross-validation ensures generalization. Success means meeting these accuracy targets while extracting insights about feature importance. For tree-based models, I'll analyze importance scores to understand salary drivers. I'll create visualizations comparing predicted versus actual salaries to identify patterns in model performance.