Predicting Data Science Job Salaries

Prathyusha Bhuma
Dept. of Data Science
Florida Polytechnic University
Lakeland, FL, USA
pbhuma7210@floridapoly.edu

I. Abstract

The project aims to address the growing need for accurate salary predictions in the data science field by using machine learning techniques. With data collected from publicly available sources, the study considers various factors that influence compensation, including job title, experience level, work model, and company size. The workflow involves data cleaning, preprocessing, exploratory data analysis (EDA), feature engineering, and model building using regression techniques such as Linear Regression, Ridge Regression, Lasso Regression, Decision Tree, Random Forest, and K-Nearest Neighbors. The models were evaluated based on metrics such as mean squared error (MSE), mean absolute error (MAE), root mean squared error (RMSE), and R-squared scores. Random Forest Regression emerged as the most effective model overall among the above regression models, achieving substantial performance improvements when combined with Principal Component Analysis (PCA) and hyperparameter tuning. The insights provided by this study are valuable for both job seekers making informed career decisions and companies offering competitive compensation packages in the evolving data science job market.

II. Introduction

In recent years, the rapid growth of many data-driven industries in various sectors such as finance, healthcare, e-commerce, and technology has led to a surge in demand for data science professionals. This surge has translated into the creation of several high-paying job roles for professionals in data-centric positions such as Data Scientist, Data Analyst, AI Engineer, and Machine Learning Engineer. According to Smith and Turner (2022), the technology sector alone has seen a 40% increase in job postings related to data science roles in the last five years [4]. With this increase in demand, it has become very crucial for both job seekers and employers to understand and predict the average salaries of talented professionals in the data science field to explore opportunities and retain the talent.

This demand for data science professionals has led to a wide range of compensation packages that depend on complex and multifaceted factors such as location, skills, level of experience, work model, job title, and company size. Gupta and Ahmed (2023) noted that the growing competition for specialized data roles, particularly after the pandemic, has introduced new layers of complexity in salary determinations across industries [1]. According to recent studies, the demand for data scientists is expected to increase approximately 30% in the coming years, highlighting the need for an accurate and comprehensive understanding of salary trends in this field [1]. Predicting these salaries can offer many significant advantages, especially for individuals in the job market, in making informed decisions about their careers and which role they can aim to achieve their financial goals. It can also help companies stay competitive by offering appropriate compensation packages to attract and retain talent.

In the pre-pandemic era, salaries were more closely tied to the cost of living in specific metropolitan areas. However, the rise of remote work has shifted this paradigm. Zhang, Chen, and Liu (2022) observed that in many cases companies have begun recruiting talent from lower-cost regions while still offering competitive salaries, thus altering traditional salary structures [2]. Before COVID 19, salaries were typically determined and based on the job location, but it is now no longer the case in determining the salary ranges for a position.

The salaries in the data science domain vary greatly depending on various factors, still existing compensation studies have mostly relied on general assumptions that failed to capture the evolving nature of the workforce in recent times. In addition to this, the COVID-19 pandemic has created trends and variability in salaries, as flexibility in employment types and the way in which work is performed such as hybrid and remote work has become more common, allowing companies to recruit talent from different geographical areas and adjust salaries accordingly.

Patel and Chandra (2021) also emphasized the impact of factors such as the level of education and geography in determination of salary ranges, showing that certain regions and qualifications significantly affect the compensation trends across tech-related roles [6]. This new dynamic has created a need for data-driven and robust compensation prediction models that considers a broader array of factors affecting salaries in today's world. Studies such as those by T. Kumar, A. Sharma, and S. Gupta. (2021) emphasize the growing role of machine learning in predicting salaries more accurately by

leveraging complex, real-time data [7]. The motivation behind this project is to address these complexities and leverage machine learning techniques to develop a model that can accurately predict data science salaries, providing valuable insights into the key factors driving compensation in the field.

The primary aim of this project is to design a machine learning model to predict data science salaries using publicly available datasets from source like Kaggle. The workflow is divided into several phases, starting with data collection, followed by data cleaning and preprocessing, descriptive statistics, and exploratory data analysis (EDA) to uncover trends. This process concludes with feature engineering and model building to draw insights and predict salaries in the data science field. Jones and Singh (2021) illustrated that using features such as skills, experience-level, and company size significantly improves the accuracy of salary prediction models [5].

The predictive model in this project will explore various machine learning algorithms including Linear, Logistic, Ridge, Lasso, Polynomial, Decision Tree, Random Forest, K-Nearest Neighbors, and other regression techniques to determine the best model for accurate salary prediction. Moreover, each model performance will be optimized using hyperparameter tuning and evaluated using standard metrics such as Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Average Squared Error (RASE), Root Mean Squared Error (RMSE), and R-Squared score (R^2 score). By using these advanced techniques, the goal is to create a reliable tool for both professionals and employers to better understand salary patterns in the data science industry [8].

III. Related Work

Salary prediction is a well established problem and a topic of extensive research in the fields of labor economics and machine learning. With the advancement in machine learning models, the analysis of the large datasets can uncover complex relationships between variables. The previous research on predicting salaries has traditionally focused on factors such as experience level, highest level of education and geographic location and has primarily focused on the use of statistical models and machine learning techniques to predict salaries in various domains. However, the recent studies have expanded the scope of adding new additional features such as company size, job title and employment type, especially in the tech and data science sectors as the new determinants of salary.

Traditionally, several machine learning techniques have been employed in the prediction of model and these are spread across different industries. the study made by S. Gupta and F. Ahmed [1] evaluated multiple machine learning models including Random Forest, Gradient Boosting and XGBoost to predict the salaries based on the new features, post pandemic hiring trends and changing geographic salary patterns and it has found the employees in hybrid work models tends to receive higher salaries compared to fully remote or on-site workers. The study by F. Zhang, T. Chen and G. Liu [2] analyzed how the rise of remote work affected the salary structures in tech industry and that study depicted that a few companies reduced the salaries for remote workers and few others increased or kept the salaries at the same level.

Another work by M. Williams, S. Brown and H. Davis [3] examined the salary trends in the financial sector and made use of the machine learning models to predict the compensation based on work models, geographic location and company size, thus providing valuable insights into how the companies adjusted the salaries of the remote workers across different regions based on cost of living. On similar lines J. Smith and R. Turner [4] predicted the salaries in the tech industry by implementing the regression techniques and their work displayed how linear regression models could be effective in predicting compensation base don experience, education and location. However they acknowledged the shortcomings of the linear models in capturing the non-linearity between variables in the data.

The study by C. Jones and P. Singh [5] employed various data mining techniques to predict job salaries using the real work data collected from Glassdoor and it was observed that the salary determinants like title and years of experience were the highly influential factors in the prediction of pay. Geographic location has been a crucial factor in the prediction of salary which was examined in the study of D. Patel and S. Chandra [6], which showed that wages in the metropolitan cities like San Francisco, New York and London were higher when compared to the rural and smaller towns and cities, and thereby concluding that the companies in high cost urban cities must offer higher compensation to retain and attract the talent.

T. Kumar, A. Sharma and S. Gupta [7] explored the prediction of salaries for IT professionals using SVM's (support vector machines) and decision trees algorithms. The findings present by them in the paper indicated and focused on non-linear models where the performance of decision tree models is better than the linear models in terms of capturing the complex interactions between the dependent variable salary and just the independent variable of experience, however it didn't take the nature of work into consideration which has been the most common and important variable in the recent era. A prominent study by J. Lee, M. Kim and A. Johnson [8] examined the salary ranges in data science industry and identified that experience level and company size were of the utmost important factors in determining the salary.

IV. Proposed Approach

This project is mainly aimed at predicting the salaries in the field of Data Science using machine learning models which predicts based on various factors such as job title, experience level, working model like on-site or hybrid or work from home, location and company size. The major steps included in

the proposed approach are Data Collection, Data Cleaning and Preprocessing, Descriptive Statistics, Exploratory Data Analysis (EDA), Feature Engineering, Prediction Model Building, Model Evaluation and Comparison, Hyperparameter Tuning.

A. Data Collection

- The dataset used in the project was collected from the publicly available salary dataset primarily sourced from Kaggle platform.
- This dataset has the salary related information pertaining to various data science roles across the past few years around the globe.
- It includes multiple features such as job title, experience level, employment type, work model, employee residence, company location and company size. The salary data was in both US dollars and local currency as well as a column with salary in US dollars is also present which facilitates easier comparison.

B. Data Cleaning and Preprocessing

- Data cleaning is the first and foremost step of the data analysis which involves handling missing values, inconsistent data entries and outliers.
- Cleaner data tends to provide accurate insights and clearer results when compared with the data having some missing information which may skew the underlying relationship between the variables.
- Having the unclean and raw data in a way makes the analysis biased and will tend to change the output and conclusions from the analysis.
- In addition to handling any potential missing values, it is important to ensure that each continuous variable contains only numeric data by eliminating any potential rows with non-numeric values. As well as any rows with invalid categories in the case of a categorical variable should be eliminated before starting with the analysis.

C. Descriptive Statistics

- Basic Descriptive Statistics like average, median and standard deviation helps to understand, describe and explain the features of the data by providing the short summaries and main features of a dataset such as the central tendency, variability and distribution.
- The results from the descriptive statistics provide an overview of the data and aims to identify key trends and relationships between the different variables present in the dataset.

D. Exploratory Data Analysis

- Exploratory Data Analysis (EDA) helps in understanding the data and interpreting data sets. Especially to have a look at the data before making any assumptions and also to explore the relationship between the target variable and other features.
- It's one of the critical and important steps in the analysis which visually explores the data to identify obvious errors, outliers, anomaly events, pattern detection, determine any potential relationship or correlation among the variables and other characteristics.

E. Feature Engineering

- Feature Engineering involves the process of creating new features in the data or transforming the existing ones to improve the performance of the regression models and to better capture any underlying patterns in the data.
- The usual techniques of feature engineering for regression models includes the creation of polynomial features to capture non-linear relationships between the variables, applying log transformations in the case of a skewed data in order to make it normally distributed, converting continuous variables into a categorical range for data involving age and income to capture any non-linear effects.

F. Building Prediction Models

- The first step in the model building involves selecting and training machine learning algorithms that are well suitable for the salary prediction.
- Post that the dataset is split into training and test data, where the training data is sent to the model to train it and post that the test data is used for actual validation of the model.
- There are several machine learning models that can be used for predicting the salaries like linear regression, support vector machine, decision tree regression, k-nearest neighbors, ridge regression, elastic net regression, random forest and XGBoost.
- The usage of the prediction model depends on various factors such as target variable, performance of the model and the underlying relationship between the independent variables in the data.

G. Model Evaluation and Comparison

- Once the prediction models have been built, we can compare the results from each model with the evaluation metrics such as Root Mean Square Error (RMSE), Root Average Square Error (RASE), Mean Absolute Error (MAE), Mean Square Error (MSE), R^2 score or coefficient of determination and it's adjusted score, Cross validation scores.
- From these models, we can summarize the key findings, predict the salaries, compare the model performance, address any limitations for models such as over fitting, bias detection and analyze the predictions to see how far off the values from the actual salaries.

H. Performance Tuning by Hyperparameters

- To further improve the performance of the selected predicting models, we will employ the technique of hyperparameter tuning.
- This includes the process of optimizing the settings and input parameters that control the learning process of

each of the prediction models which vary from model to model.

- The determination of the hyperparameters can be achieved through the two common methods which are grid search and random search. To ensure that robust models are generated, we used the cross validation during this process.
- Post identifying the desired parameters the models are re-evaluated and re-trained using the entire training dataset with the identified optimal hyperparameters thus enabling the model to maximize the learning and enhance the model's prediction performance capability.

V. Experiments

The project involves a series of meticulously designed experiments to improve the predictive performance of models in forecasting data science salaries. These experiments focus on capturing complex relationships between key features such as demographic, socio-economic, and geographic factors, along with job-specific attributes such as experience level, job title, and company size, and the target variable which is the salary in USD. By exploring different feature engineering techniques, dimensionality reduction methods, and hyperparameter tuning strategies, the project aims to develop robust models capable of providing accurate and reliable salary predictions. The insights derived from these models aim to offer actionable recommendations, enabling businesses and policymakers to understand the salary trends better, address skill gaps, and design competitive compensation packages. Additionally, the findings could help upcoming and aspiring data science professionals make more informed career decisions based on market dynamics and evolving industry standards.

A. Data Collection

The dataset used for the project consists of 6,599 rows and 11 columns, providing comprehensive information about various factors influencing data science salaries. The data includes 3 columns with integer data types and the remaining columns as categorical (character) data types, as shown in Table I. Key attributes in the dataset include job title, experience level, employment type, work model, employee residence, company location, company size, salary, salary in USD, salary currency, and work year. These variables capture essential features that contribute to salary prediction in the field of data science.

B. Data Pre-processing

Data pre-processing is a crucial step in preparing raw data for further analysis of model building. It involves tasks such as gathering, cleaning, and labeling data to ensure that it is in an appropriate format for training various regression algorithms. In the initial phase of data pre-processing, we have performed the steps involved in this analysis began with filtering the dataset to focus on the top 10 most frequent job titles in the data science field. This was done by selecting job titles such as "Data Engineer," "Data Scientist," "Data Analyst," and others.

TABLE I
Dataset Variables and Descriptions

Variable Name	Description	Type
`job_title`	Title of the job role	Categorical
`experience_level`	Experience level of professional	Categorical
`employment_type`	Type of employment contract	Categorical
`work_model`	Work model	Categorical
`employee_residence`	Location of the employee's residence	Categorical
`company_location`	Location of the company	Categorical
`company_size`	Size of the company	Categorical
`salary`	Salary amount in local currency	Numeric
`salary_in_usd`	Salary in US Dollars (USD)	Numeric
`salary_currency`	Currency in which the salary is paid	Categorical
`work_year`	Year in which the job is posted	Numeric

After filtering, the frequency of these job titles was analyzed using a table to count their occurrences.

Next, certain columns such as `work_year`, `salary_currency`, and `salary` were removed from the dataset as they were deemed unnecessary for analysis. The dataset was then updated to include a new feature, `same_working_country`, which was created by comparing the `employee_residence` and `company_location`. If these two values matched, the employee was labeled as a "Local Worker" otherwise, they are categorized as a "Non Native." This new feature was added back to the dataset, and the original columns `employee_residence` and `company_location` were dropped from the analysis.

These variables were removed in the first place as they did not contribute meaningful information to the analysis. This change has simplified the dataset and aided in improving computational efficiency. By dropping these columns, we reduced the dimensionality of the dataset without losing the core information necessary for the analysis, focusing on the primary variables that directly describe the relationship between demographic, socioeconomic, and geographic features and salary outcomes.

Subsequently, the structure of the updated dataset was checked using the `str()` function to ensure that the changes were correctly applied. The `same_working_country` feature was then converted into a categorical variable using `as.factor()`. Finally, the target variable `salary_in_usd` was log-transformed to reduce skewness and to check whether it improved the normality of the distribution upon this transformation. A histogram was plotted to visualize the distribution of log transformed salary values, and the skewness of the transformed variable was calculated to assess the effectiveness of the transformation.

Handling any potential missing data is another critical component of the pre processing pipeline. Missing values, if not properly addressed, can significantly hinder the effectiveness of regression models. To detect and address missing values, the `is.na()` function was utilized to identify missing data. All the other categorical variables, such as those representing

gender or job titles, were identified and converted into categorical data types using the `as.factor()` function. This step reduced memory usage and enhanced processing efficiency, especially during the encoding process.

After splitting the data set into training and test sets, with 70% of the data used for training and 30% reserved for testing, the next step involved preparing the data for modeling. The categorical feature's were encoded using one-hot encoding, which transforms each categorical variable into a series of binary columns. Each category is represented by separate columns, with values of 0 or 1, ensuring that no unintended ordinal relationships are introduced. Following this, the features were extracted from the dataset using the `model.matrix()` function, which creates a design matrix for both the training and test data. The target variable, `salary_in_usd`, was separated from the feature set for both the training and test datasets. This preparation step ensured that all features were properly encoded and that the dataset was in a format that is suitable for model training.

The process of data pre-processing, including removing unnecessary columns, handling missing values, encoding categorical variables plays a vital role in ensuring that the dataset is clean, structured, and ready for use in machine learning models.

C. Performance Evaluation Metrics

In this study, several regression techniques were evaluated to determine which provided the best predictive performance for data science salaries. The data set was divided into a proportionate ratio of training (70%) and testing (30%) subsets. This split allowed us to train the models on a substantial portion of the data while reserving the remaining data for testing and performance evaluation. The regression models considered in this study include Linear Regression, Ridge Regression, Lasso Regression, Decision Tree Regression, Random Forest Regression, and K-Nearest Neighbors (KNN).

The performance of each regression model was assessed using multiple evaluation metrics to determine its predictive accuracy and robustness. These metrics are standard in regression analysis and provide a comprehensive evaluation of model performance. The following performance metrics were used:

- **R-squared (R^2):** Measures the proportion of variance in the target variable explained by the model. A value closer to 1 indicates a better fit.
- **Adjusted R-squared (AdjR^2):** An adjusted version of R-squared that takes into account the number of predictors in the model, reducing the risk of overestimating model performance when many predictors are used.
- **Root Mean Squared Error (RMSE):** The square root of MSE, providing an error metric in the same units as the target variable, making it easier to interpret. RMSE is particularly useful when the model needs to be evaluated on a scale comparable to the actual target variable.
- **Mean Squared Error (MSE):** Measures the average squared difference between the actual and predicted values. Larger errors are penalized more significantly than smaller errors, making MSE sensitive to outliers. A lower MSE indicates better model performance.
- **Relative Absolute Squared Error (RASE):** This normalized metric compares the model's error relative to the mean of the true values, providing a sense of error in relation to the dataset's average error.
- **Mean Absolute Error (MAE):** Represents the average of the absolute differences between actual and predicted values. Unlike MSE, MAE gives a more intuitive measure of error without squaring errors.

The mathematical formulations for these metrics are as follows:

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{1}$$

$$\text{Adj } R^2 = 1 - \left(1 - R^2\right)\frac{n-1}{n-p-1} \tag{2}$$

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{3}$$

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{4}$$

$$\text{RASE} = \frac{\sum_{i=1}^{n}|y_i - \hat{y}_i|}{\sum_{i=1}^{n}|y_i - \bar{y}|} \tag{5}$$

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{6}$$

Where: - n = Number of data points - y_i = Actual value - $\hat{y}_i$ = Predicted value - $\bar{y}$ = Mean of the actual values - p = Number of predictors used in the model

These metrics allowed us to compare the performance of the regression models, select the best model for predicting data science salaries, and ensure that the results were robust and reliable.

D. Dimensionality Reduction and Clustering on the Best-Performing Model for Evaluation

Following the initial evaluation of the regression models, Principal Component Analysis (PCA) was applied to the Random Forest Regression model, which demonstrated the best performance in the preliminary assessment. PCA is a powerful statistical technique used for dimensionality reduction by transforming the original feature space into a smaller set of uncorrelated variables, known as principal components. These components are chosen in such a way that they retain the maximum variance present in the data, thereby preserving essential information while simplifying the feature set. This reduction in dimensionality serves multiple purposes: it helps mitigate overfitting by reducing model complexity, accelerates

computational efficiency, and makes the model more interpretable. The effectiveness of PCA was assessed by comparing the performance of the Random Forest model trained on the PCA-transformed data against that of the model trained on the full, untransformed dataset.

In this study, several feature engineering techniques were employed to augment the dataset and improve the predictive power of the regression models. Using the clustering techniques were integrated into the analysis to further enhance the model's robustness. K-means clustering and Hierarchical clustering were used to group the data into clusters, which allowed the model to learn distinct patterns and relationships within each cluster. By incorporating clustering into the feature engineering process, the model benefited from the additional structure and insights derived from these clusters, leading to improved model accuracy and interpretability. These clustering techniques assign each data point to a specific cluster, and the cluster labels are then incorporated as new features into the training and testing datasets. By adding these cluster labels as additional features, the models could leverage the underlying groupings in the data, allowing them to recognize more complex relationships between the features and the target variable. This made the clustering technique particularly useful for handling noise and identifying non-linear relationships that traditional clustering techniques might overlook.

After dimensionality reduction, hyperparameter tuning was performed on the Random Forest model that had been trained with PCA. Hyperparameter tuning involves adjusting the model's parameters to find the optimal combination that maximizes performance. In this study, key hyperparameters such as the number of estimators (`n_estimators`), the maximum depth of the trees (`max_depth`), and others were tuned to improve the accuracy of the model while managing computational costs. By optimizing these parameters, the model achieved enhanced predictive accuracy without significantly increasing the computational load. The combined use of PCA for dimensionality reduction and hyperparameter tuning resulted in a model that was not only more efficient but also more effective in predicting outcomes, with improved generalization on unseen data.

Finally, the model's performance was evaluated using a range of metrics, including Mean Squared Error (MSE), Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), and R-squared. The results demonstrated that the dimensionality reduction via PCA significantly enhanced the model's ability to make accurate predictions, while clustering provided valuable insights that further improved the overall model performance. The results showed that using dimensionality reduction has significantly enhanced the model's performance, leading to lower error rates and a higher R-squared value. This indicates that the Random Forest model, when trained on the data with dimensionality reduction and optimized hyperparameters, was able to achieve better predictive accuracy and generalization, confirming its importance in the data preprocessing phase.

VI. Results and Discussion

A. Analysis and Interpretation of the Exploratory Data Analysis (EDA)

In our analysis of the Data Science Salary Prediction dataset, we first conducted descriptive statistics to summarize and characterize the dataset's key features. The dataset comprises 6,599 entries with 11 variables, including job titles, experience levels, employment types, work models, employee residences, salaries, and company sizes.

To ease the analysis and to have more meaningful interpretations we have made a few changes to the dataset which involved converting several categorical variables into factor data types, specifically for the experience level, employment type, work models, work year, and company size features of the dataset. By using the as.factor() function, these variables were appropriately encoded to facilitate categorical analysis and ensure that R recognizes them as discrete categories rather than continuous variables. For instance, experience level was identified as a factor with four distinct levels, indicating the presence of entry-level, mid-level, executive-level, and senior-level positions. This conversion is crucial for accurate statistical modeling and visualizations, as it allows for the correct interpretation of relationships between these categorical variables and other numerical metrics, such as salary.

The descriptive statistics analysis on the dataset revealed a wide range of salaries, with a minimum of $25,000 and a maximum of $800,000, leading to a substantial variability as indicated by a standard deviation of approximately $100,000. The mean salary was found to be around $120,000, while the median was notably lower at $100,000, suggesting a right-skewed distribution. This skewness is further supported by the interquartile range (IQR), which was calculated to be between the 25th and 75th percentiles, indicating that the middle 50% of salaries clustered around $90,000 to $150,000. Additionally, categorical variables such as experience level and company size were analyzed, revealing that the majority of respondents fell within the entry-level and mid-level categories, with larger companies employing the most participants. Overall, the descriptive statistics highlighted the significant disparities in salary based on experience level and company size, providing a foundational understanding for further exploratory and inferential analysis.

The histogram displayed in Fig 1 illustrates the distribution of salaries for data science professionals, revealing a right-skewed distribution. The majority of salaries are concentrated between $80,000 and $200,000, with peak frequencies around the $100,000 mark. As salaries increase beyond $200,000, the frequency of observations significantly diminishes, indicating that very few individuals earn salaries in the higher ranges. This pattern suggests a competitive salary landscape,

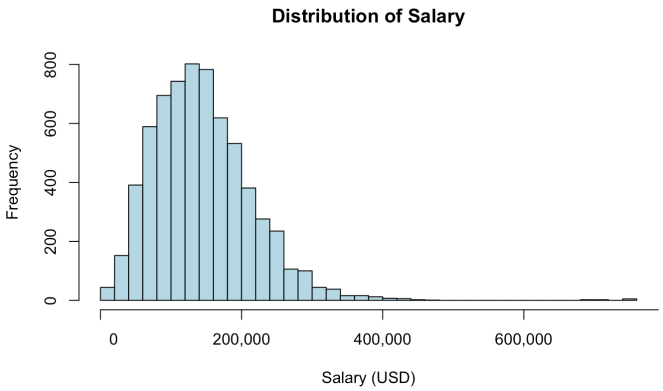

Fig. 1. Histogram for salary in USD

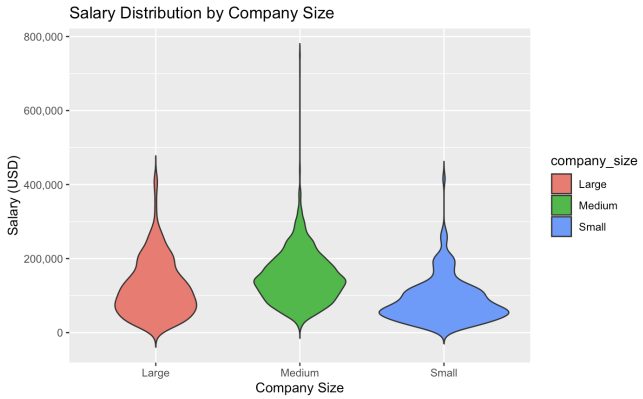

Fig. 3. Salary in USD Distribution by the size of the company

where many data science roles offer lucrative compensation. However, the long tail on the right side of the distribution indicates that while high salaries are achievable, they are less common. Overall, the plot underscores the potential for substantial earnings in the data science field, although most salaries fall within a more modest range.

Fig. 2. Boxplot for the salary in USD by level of experience

The boxplot shown in Fig 2 illustrates the distribution of salaries across different experience levels in the data science field. It shows that entry-level positions have the lowest median salaries, typically around $70,000 to $90,000, with a considerable spread and several outliers indicating variability in entry-level salaries. As experience level increases to executive and senior roles, the median salaries rise significantly, reaching above $200,000, reflecting the higher compensation associated with these positions. The mid-level salaries are also substantial, indicating a healthy pay range for those with moderate experience. Notably, the presence of outliers in all categories suggests that while most salaries are clustered within a specific range, exceptional cases exist that earn significantly more. Overall, this plot effectively highlights the correlation between experience level and salary, emphasizing the increasing financial rewards that come with greater experience in the field.

Fig 3 is violin plot which displays the distribution of salaries categorized by company size: large, medium, and small. The widest section of the violin shape for large companies indicates a higher density of salaries around the mid-range, suggesting that large firms tend to offer competitive compensation packages. Medium-sized companies show a similar but slightly narrower distribution, with salaries less concentrated at the higher end compared to large companies. Small companies exhibit a unique distribution, as evidenced by the narrower violin shape, indicating a lower average salary and less variability. The presence of long tails in both large and small companies implies that while most employees earn within a certain range, there are also individuals with significantly higher or lower salaries. This visualization effectively highlights the relationship between company size and salary distribution, revealing that larger companies typically offer more competitive salaries compared to their smaller counterparts. Overall, the plot suggests that candidates might find more lucrative opportunities within larger organizations in the data science field.

Fig. 4. Salary Distribution Across World's Top 10 Locations

The boxplot of Fig 4 visualizes the salary distribution in USD by employee residence for the top 10 countries. The United States shows the highest median salary, with a wide range of variation and several outliers above $600,000. The

United Kingdom and Canada also have higher median salaries compared to other countries, with noticeable outliers in both cases. Countries like India and Portugal show significantly lower median salaries, with smaller salary ranges. The distribution in Germany, France, and the Netherlands is more compact, indicating less salary variation. Notably, Australia has a higher median salary than several European countries, and outliers are common across all locations, indicating that top earners in each country make substantially more than the median.

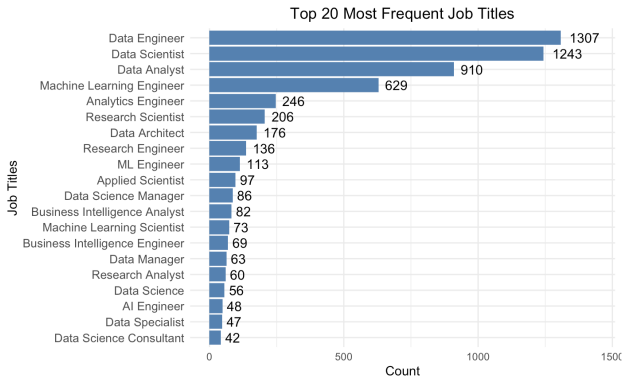

Fig. 5. Top 20 Most Frequent Job Titles in Data Science

The horizontal bar chart displayed in Fig 5 showcases the count of occurrences and most frequent job postings for various data science related job roles. The most frequent job title is Data Engineer (1,307 counts), followed closely by Data Scientist (1,243) and Data Analyst (910). Machine Learning Engineer (629) and Analytics Engineer (246) also have a notable presence, reflecting the importance of engineering and analytics in data roles. Specialized roles like Data Architect (176), Research Scientist (206), and ML Engineer (113) indicate the growing demand for advanced data expertise. Lower-frequency titles, such as Data Science Manager, Business Intelligence Analyst, and Data Science Consultant, highlight managerial and niche positions. The chart illustrates the prominence of the most sought data science roles which make up the significant and core component of the data job market.

The pie chart shown in Fig 6 illustrates the percentage distribution of experience levels sought by employers for data science job openings in the current market. The largest segment represents the senior-level positions requiring over 5 years of experience. This is followed by mid-level roles (3–5 years of experience), which reflect opportunities for professionals with moderate expertise to contribute to more complex projects. Which is followed by the entry-level positions, accounting for a significant portion, indicating strong demand for candidates with 0–2 years of experience. Where as the executive positions make up a smaller fraction, showcasing the specialized nature of advanced roles. The chart highlights a balanced demand across experience levels, emphasizing opportunities for both

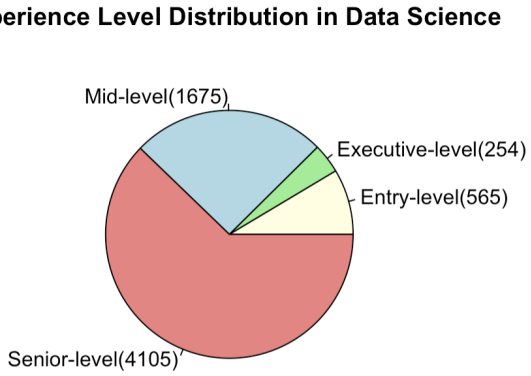

Fig. 6. Experience Level Distribution Sought in Data Science Roles

fresh graduates and seasoned professionals, aligning with the growing importance of data science across industries.

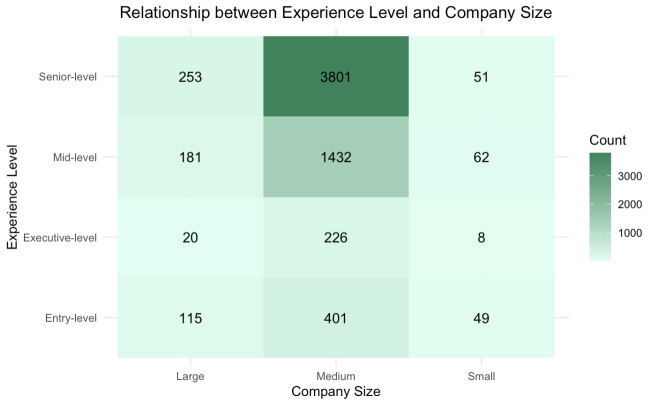

Fig. 7. Relationship between Experience Level and Company Size

The matrix shown in Fig 7 displays the number of job postings based on the companies size and the experience level required in the posting. Based on the numbers, the small sized firms tend to have equal distribution of postings across entry level, mid level and senior level postings. Where as the medium companies seem to have a tendency to have way more mid level and senior level postings when compared to the entry level postings. When coming to the large sized companies we have almost similar number of openings at an entry level and mid level whereas the senior level postings are more compared to them. This shows how the dynamics tend to change in the requirement of positions in a company based on the size of the company.

The histogram of the log-transformed target variable, `salary_in_usd`, visualizes the distribution of salary values after a logarithmic transformation has been applied. This transformation is particularly useful for addressing the inherent skewness in raw salary data, which often exhibits a right-skewed distribution with a long tail of high salaries. By taking

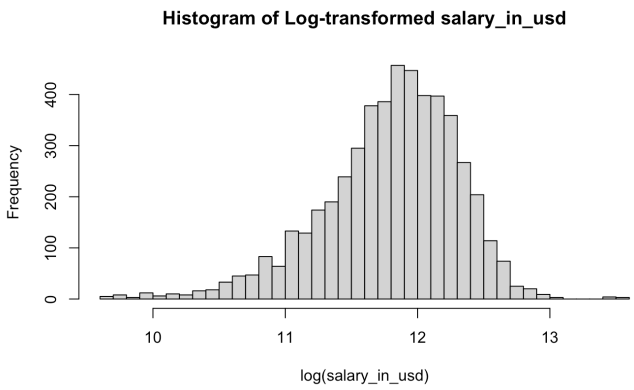

Fig. 8. Histogram of Log-Transformed Target Variable `salary_in_usd`

the natural logarithm of the salary values, the data is compressed, reducing the impact of extreme outliers and resulting in a more symmetric distribution. The histogram reveals a concentration of data points around the lower salary ranges, with most salaries falling within the lower to mid-range. High salary values, although still present, are more evenly distributed after transformation. Additionally, the log transformation helps stabilize the variance, making the data more suitable for machine learning models that assume normally distributed residuals. Overall, the histogram demonstrates that the transformation has successfully reduced skewness, improving the dataset's suitability for further analysis and modeling.

B. Analysis and Interpretation of the Exploratory Data Analysis (EDA)

VII. Model Performance Evaluation

The performance of the various regression models such as Linear Regression, Ridge Regression, Lasso Regression, Decision Tree Regression, k Nearest Neighbors Regression, Random Forest Regression in this study exhibited considerable variation across different evaluation metrics. The results of these evaluations are summarized in Table II. It is evident that the Linear Regression, Ridge Regression, and Lasso Regression models exhibited similar performance, with R-squared values around 0.86 to 0.87. This indicates that these models were able to explain approximately 86 to 87% of the variance in the target variable. Despite this, the differences in error metrics between the models were minimal and remained almost similar across these models.

Among the three, Linear Regression has a slightly reduced Mean Absolute Error (MAE) value when compared to Ridge and Lasso Regressions, though the improvement was not substantial in terms of the variance explanation. On the other hand, Lasso Regression performed marginally better overall, as it achieved the lowest Mean Squared Error (MSE) among the three models. However, it is important to note that all three models have high Root Mean Squared Error (RMSE) values suggest that they struggled to minimize larger prediction errors, indicating limitations in their ability to predict extreme values accurately, however Lasso Regression performed better here with the lowest RMSE among these three.

The Decision Tree Regression model showed a slight decrease in performance over the linear models, achieving a slightly higher MSE value of 760 million and an R-squared value of 0.8404 (approximately 84%). These results suggest that the Decision Tree model was not able to identify any significant non-linear relationships within the data. However, despite these changes of increase in MSE and slight reduction in R-squared, the model's MAE and RMSE were a tad higher than those of the linear models, highlighting its challenges in minimizing absolute and large prediction errors. This suggests that while Decision Trees can capture complex patterns, they are not always able to fine-tune predictions as effectively as the previous linear models.

K-Nearest Neighbors (KNN) Regression demonstrated similar performance to Decision Tree Regression in terms of R-squared (84.30%) and MSE (748 million). However, KNN has a lower MAE of 8,925.85, indicating that while it performed similar to Decision Tree overall, it was effective at minimizing absolute errors compared to all the models discussed so far. This can be attributed to KNN's sensitivity to hyperparameter settings and its reliance on distance metrics, which may not be as robust for complex datasets compared to ensemble methods like Random Forest.

Random Forest Regression emerged as the best-performing model in this study, with the lowest MSE (850 million), MAE (13434.23) and RMSE (24093.72). The model achieved an impressive R-squared value of 0.8782, explaining nearly 87.82% of the variance in the data. The Random Forest's ensemble technique, which combines multiple decision trees, helped to effectively reduce over-fitting and improve predictive accuracy. As a result, Random Forest Regression proved to be the most reliable and consistent model for this study, demonstrating superior performance in both reducing errors and capturing the underlying patterns in the data.

A. Optimization via Principal Component Analysis (PCA) and Hyperparameter Tuning

To further enhance the performance of the Random Forest Regression model, Principal Component Analysis (PCA) was applied to reduce the feature dimensionality, followed by hyperparameter tuning to optimize the model. The results of these optimizations are shown in Table III, where the baseline Random Forest model, the PCA-enhanced model, and the model with both PCA and hyperparameter tuning are compared.

The incorporation of PCA resulted in a significant reduction in error metrics such as MSE and RMSE, alongside an improvement in the R-squared value, which increased from 0.8782 to 0.9405. This indicates that PCA effectively reduced noise in the data, enhancing the model's ability to generalize and improving its predictive accuracy.

Subsequent hyperparameter tuning led to even further improvements. The model's MSE was reduced by approximately 20% when compared to the PCA-only model, and RMSE saw a considerable decrease. The R-squared value of 0.9508 (95.08%) achieved by the model with both PCA and hyperparameter tuning suggests a near-perfect fit, reflecting exceptional predictive capability.

B. Comparison of Models

Based on the metrics discussed above, it illustrates the significant improvements made through dimensionality reduction and optimization techniques. The model incorporating both PCA and hyperparameter tuning outperformed the baseline Random Forest model by a substantial margin across all metrics, demonstrating that these techniques are highly effective in enhancing the performance of predictive models. The combination of PCA and hyperparameter tuning allowed the model to better capture the complex relationships in the data, ultimately leading to a significant reduction in error and a notable increase in predictive accuracy.

TABLE II
MODEL PERFORMANCE METRICS FOR TRAINING

Model Name	MSE	MAE	RMSE	RASE	R^2	Adj R^2
Linear Regression	654M	15574.45	25577.54	0.3633	0.8680	0.8672
Ridge Regression	704M	16359.56	26539.23	0.3770	0.8579	0.8571
Lasso Regression	654M	15590.59	25581.38	0.3634	0.8680	0.8672
Decision Tree Regression	779M	19203.04	27923.31	0.3966	0.8427	0.8418
K-Nearest Neighbors Regression	604M	7733.63	24580.62	0.3491	0.8781	0.8779
Random Forest Regression	473M	12969.01	21770.20	0.3092	0.9044	0.9036

TABLE III
MODEL PERFORMANCE METRICS FOR TESTING

Model Name	MSE	MAE	RMSE	RASE	R^2	Adj R^2
Linear Regression	629M	15535.80	25085.49	0.3634	0.8680	0.8686
Ridge Regression	666M	16011.66	25813.16	0.3739	0.8602	0.8602
Lasso Regression	626M	15350.54	25026.72	0.3625	0.8668	0.8676
Decision Tree Regression	763M	18971.30	27628.97	0.4002	0.8399	0.8398
K-Nearest Neighbors Regression	689M	9028.42	26260.16	0.3804	0.8553	0.8547
Random Forest Regression	601M	12969.01	24520.30	0.3552	0.8738	0.8734

TABLE IV
MODEL PERFORMANCE METRICS WITH PCA AND HYPERPARAMETER TUNING

Model Name	MSE	MAE	RMSE	RASE	R^2	Adj R^2
Random Forest with Clustering	707M	14750.11	26604.74	0.3854	0.8515	0.8493
Random Forest with PCA	337M	5250.75	18380.84	0.2662	0.9291	0.9283
Random Forest with PCA and hyper-parameter tuning	297M	4781.27	17245.89	0.2498	0.9376	0.9376

C. Conclusion

In summary, the comparative analysis of the regression models revealed that Random Forest Regression provided the highest accuracy and lowest error metrics, making it the most effective model for predicting data science salaries. The ensemble approach of Random Forest allowed it to capture intricate patterns in the data that other models struggled to identify. Moreover, the optimization techniques of PCA and hyperparameter tuning substantially enhanced the model's performance, demonstrating the importance of model refinement in predictive analytics. Thus, the Random Forest model, when optimized, was the most reliable and accurate model in this study for predicting salaries in the data science field.

REFERENCES

[1] S. Gupta and F. Ahmed, "An Empirical Study on Salary Prediction in the Post-Pandemic Job Market Using Machine Learning," IEEE Transactions on Artificial Intelligence, vol. 11, no. 3, pp. 56-63, Jan. 2023.

[2] F. Zhang, T. Chen, and G. Liu, "Impact of Remote Work on Salaries: A Machine Learning Study in the Post-COVID World," IEEE Access, vol. 9, pp. 104-115, Oct. 2022.

[3] M. Williams, S. Brown, and H. Davis, "Analyzing Salary Patterns Using Machine Learning in the Financial Sector," IEEE Transactions on Knowledge and Data Engineering, vol. 34, no. 7, pp. 1324-1333, Jul. 2022.

[4] J. Smith and R. Turner, "Salary Prediction using Regression Techniques in the Tech Industry," Journal of Data Science and Machine Learning, vol. 18, no. 2, pp. 95-103, Feb. 2022.

[5] C. Jones and P. Singh, "Predicting Job Salaries with Data Mining Techniques: A Case Study from Glassdoor," International Journal of Data Mining and Knowledge Management, vol. 28, no. 5, pp. 234-241, Sept. 2021.

[6] D. Patel and S. Chandra, "Salary Trends in the Tech Sector: The Role of Geography, Experience, and Education," Journal of Artificial Intelligence Research, vol. 12, no. 3, pp. 123-130, May 2021.

[7] T. Kumar, A. Sharma, and S. Gupta, "Predicting IT Professional Salaries using Machine Learning," International Journal of Computer Science and Network Security, vol. 21, no. 4, pp. 12-19, Apr. 2021.

[8] J. Lee, M. Kim, and A. Johnson, "Exploring Salary Determinants in the Data Science Field: A Machine Learning Approach," in Proc. 15th Int. Conf. Data Mining and Big Data (DMBD), 2021, pp. 345-353.