**Data Analyst Salary Prediction**

**Problem Definition:**

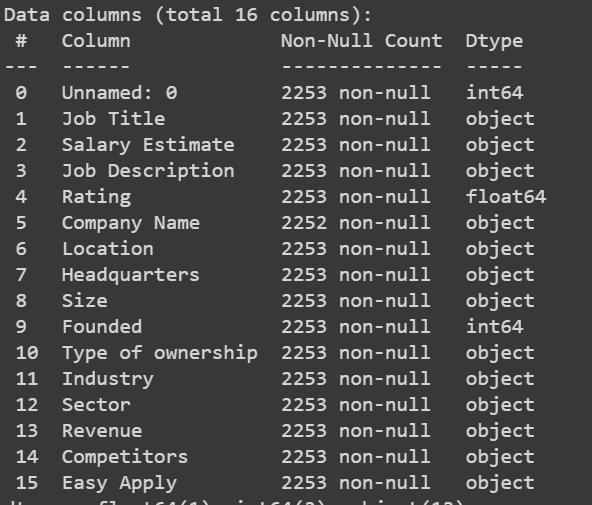
* The present work market is competitive, yet there is still a great demand for qualified data analysts. The increased need is caused by the necessity for data-driven decision making across numerous industries.
* To address this problem, we are building a predictive model that can predict the salary based on the industry, location, company revenue. This model will use Salary Estimate, Location, Company Rating, Job Description and more to identify the best jobs by salary and company rating.
* The model will be trained on a dataset of more than 2000 records of Data analyst positions and will be able to predict salary based on industry, location, company revenue

**Data Sources:**

Kaggle dataset:

<https://www.kaggle.com/datasets/andrewmvd/data-analyst-jobs>

**Data Description:**

* The dataset we are going to use is DataAnalyst.csv. This dataset contains 2253 records and 16 columns
* The dataset has the following information:

**Variables in data set**

Text

Description automatically generated

This dataset does not have null values but has (-1). Replaced them with np.nan

Text

Description automatically generated

**Exploratory Data Analysis:**

Used describe() command to basic statistics such as count, unique, top, and frequency for the object columns.

A screenshot of a computer

Description automatically generated

Due to the presence of categorical data, exploratory data analysis required some initial data preprocessing activities such as converting object data types to numerical data types to enable for better understanding of the data.

**Data Cleaning**

* Due to the large number of missing values in the salary-related columns, imputation techniques such as mean or median could not be applied. Therefore, the missing values were dropped from the dataset to avoid bias in the analysis.

A screenshot of a computer

Description automatically generated with medium confidence

**Data type conversion**

* The data contained several object data types which were not compatible with the data analysis techniques used in this project
* The object data types were converted to integer or float data types as per the requirements of the analysis techniques.
* The conversion process involved using various techniques such as label encoding, one-hot encoding, and manual mapping

Text

Description automatically generated

**Visualizations:**

Chart, histogram

Description automatically generated

This graph shows that majority of companies offering jobs are having ratings ranging between 3 to 4.5

Chart, funnel chart

Description automatically generated

This graph illustrates that most of the jobs postings are in CA followed by TX

Chart, bar chart

Description automatically generated

This graph shows highest and lowest salaries offered by companies across different states

Chart

Description automatically generated

This graph shows number of job openings across different job positions

Chart, scatter chart

Description automatically generated

1. The first graph shows that the companies with rating between 3 to 4.5 are offering the salaries ranging from 40k to 60k USD (Lowest salary)
2. The second graph shows that the companies with rating between 3 to 4are offering the salaries ranging from 70k to 100k USD (Highest Salary)
3. The third graph shows that the companies with rating between 3 to 4.5 are offering the salaries ranging from 80k to 110k USD (Average salary)

**Encoding and feature engineering:**

* Encoding involves transforming categorical data into numerical data, which can be achieved through techniques such as one-hot encoding and label encoding, while mapping involves converting values of one data type to another data type, such as converting string values to numeric values, or scaling values to a specific range for standardization or normalization purposes.

A screenshot of a computer

Description automatically generated with medium confidence

To handle categorical variables in the dataset, we used the get\_dummies function for one-hot encoding. This helped in transforming the categorical variables into numerical variables, making it easier for the machine learning model to understand the data.

Graphical user interface, text

Description automatically generated

**Data Reduction:**

* In the data reduction step, Principal Component Analysis (PCA) is used to reduce the dimensionality of the data, handling multicollinearity, and improving computing efficiency.

Text

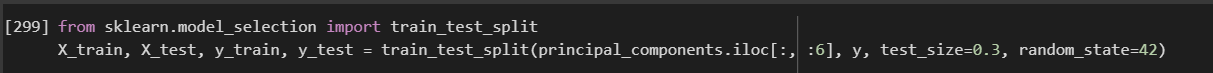
Description automatically generated

* After applying Principal Component Analysis (PCA) on our data, the explained variance ratio was obtained for each principal component. This helped us in identifying the most significant components in the dataset. we considered the top components for further analysis while the rest were dropped.

Text

Description automatically generated

* After applying PCA, then we split our dataset into train and test sets to ensure that your model is not overfitting to the data



**Model Building**

Using scikit-learn, we built a pipeline to train and evaluate various machine learning models

Text

Description automatically generated

Text

Description automatically generated

**Model Training:**

* We trained different models like linear regression, KNN, Decision Tree, Random Forest , SVM, Boosted Tree, MPL regressor
* The outputs are as follows

Text

Description automatically generated

Text

Description automatically generated with low confidence

**Tuning Hyperparameters:**

* After models are trained, we select the best 3 models Random Forest, Decision Tree, Boosted tree and we did grid search for each model and used it on test data and the results are as follows  
  **Random Forest:**

Text

Description automatically generated

* **Decision Tree:**

Text

Description automatically generated

* **Boosted tree:**

Text

Description automatically generated

**Gains chart:**

* We plotted Gains chart for the 2 best performing models which are random forest and Boosted tree

Chart, line chart

Description automatically generated

**Conclusion**

Despite using a variety of methods, including feature engineering, encoding, and PCA, the model's R2 score was determined to be low. Some of the possible reasons are:

1. The performance of the model can be significantly impacted by the level of the training data. To increase the model's accuracy, it is crucial to carefully review the data and deal with any problems that might be present, such as cleaning the data or deleting outliers.
2. It's possible that the model lacks the data necessary to understand the complicated relationships between the features and the target variable. The performance of the model might be enhanced by extending the sample size or gathering more data.
3. Other potential reason for the low R2 score of the model is the presence of missing values in the data. Specifically, out of 2253 records, there were around 400 null values in critical columns such as rating, founded and size. Due to the large number of missing values, it was not possible to impute them without significantly affecting the accuracy of the salary prediction. As a result, all columns with missing values were removed, which may have reduced the amount of information available to the model and potentially affected its performance.

**Summary**

To summarize, we used a data analyst jobs dataset to predict salaries for data analysts. Despite the use of numerous techniques such as feature engineering and PCA, the machine learning model's performance was unsatisfactory, with an R2 score of only 0.14. The inclusion of missing values in essential columns such as rating, founded, and size was one of the primary causes of the bad performance. To avoid any detrimental impact on salary projection accuracy, we chose to eliminate these columns with missing values. However, this may have curtailed the number of features available to the model, contributing to its low performance.