Documentation for Project :

Identify Fraud from Enron Dataset

The goal of the project is to identify with the help of machine learning which people in the Enron company were involved in the fraud which resulted in the bankruptcy of the company in the year 2001. We have used the Enron dataset on which machine learning is applied. Through machine learning, we are enabled to find common patterns followed by people involved in the fraud which separate them from the ones which are not involved in the fraud.

Data Exploration:

Total number of data points: 146

Number of POIs: 18

Number of Non-POIs: 127

Number of features used: 4 (Selected through SelectKBest algorithm)

There are many features with missing values. The top ones with the number of missing values are:

'loan_advances' : 142

'director_fees' : 129

'restricted_stock_deferred' : 128

'deferral_payments' : 107

'deffered_income' : 97

Outlier investigation:

One outlier was found in the dataset which was named 'TOTAL', which consisted of the total of all data points' features in its features. It was removed from the dictionary of the dataset using the 'pop' function of dictionaries.

New Features:

A new feature 'poi_messages' which is equal to the sum of 'from_poi_to_this_person' and 'from_this_person_to_poi' is created. This is relevant because this gives an overview of the

total correspondence happening between this person and (other)persons of interest. This feature was not included in the final features selected, but it did have an impact on the selection of the best features through GridSearchCV. The performance of the algorithm including the feature and without including is given at the end, where the metrics for measuring the efficiency of the algorithm are given.

Feature Selection:

The features were selected using the SelectKBest algorithm in from SciKit-Learn and GridSearchCV was used to find the optimum solution. The options given in the parameter grid were 4,5,6,7,8, out of which 4 was selected. The 4 features selected along with their scores are:

'salary': 9.324

'bonus': 13.029

'from_poi_to_this_person': 5.0389

'shared_receipt_with_poi': 7.7926

Features Scaling:

The features were scaled using the MinMaxScaler available in SciKit-Learn. Scaling was employed because otherwise features with large numerical values are given undue weight. Scaling balances the weight of the features.

Algorithm:

Gaussian Naive Bayes algorithm and Decision Tree algorithm were picked for the project. Gaussian Naive Bayes was used in the final analysis.

Parameter Tuning:

Parameter tuning is required to fit our data to our algorithm. Tuning is necessary if we want to optimize our algorithm according to the data that we have. There are different parameters in each algorithm which need to be altered to fit it to the dataset we have so that we could have accurate predictions.

The algorithm that we have used in the final analysis does not require parameter tuning, but for the other algorithm, tuning is performed using GridSearchCV.

Validation:

Validation is used to assess the performance of the algorithm decided. This can be done by dividing the training data into chunks and then using a smaller chunk as the testing data. We

can tune the parameters of our algorithm with its help. Not doing validation could result in overfitting of our model.

We used Grid Search Cross Validation (GridSearchCV) as our validation strategy, to tune the parameters.

Evaluation Metrics:

Performance of the Gaussian Naive Bayes algorithm:

With new feature(**used in final analysis**):

Precision: 0.41703

This means that out of all the people classified as persons of interest by our classifier, 41.703 % were classified correctly.

Recall: 0.338

This means that our algorithm was able to correctly classify 33.8 % people who were actually persons of interest.

Without new feature:

Precision: 0.40235

Recall: 0.308

Performance of the Decision Tree algorithm:

Precision: 0.31575

Recall: 0.2065