Identify Fraud from Enron Email Dataset

1. Goal of this project is to Identify the fraud people from Enron Company's email dataset by using Supervised Machine learning method. Enron company email dataset is huge, it has a total 145 data points. The Class POI has 18 data points and Non-POI has 127 data points. 21 features in the dataset. A few outliers, I have identified them using a scatter plot, I have plotted one of the features of the dataset, Salary against Person and I found that the person name 'Total' does not seem like a real person and I have immediately removed it considering as outlier

2. Initially I have chosen all the features in the dataset except 'email address' because it is a string and does not seem useful in classification and I have used PCA to pick the most important features, I did scaling for all the features because there is a mix of financial and email features. I have scaled all the features in the range 0 to 1.

   I was interested in learning about the total income earned by each person , so I created a new feature 'total income' which is the sum of salary ,bonus and stock value .Including this new feature in the final algorithm reduced the recall value a bit, that was not much a difference but the new feature did not give any benefit , so I removed it in the final analysis and continued with the other features in the dataset.

3. I have used Naive Bayes Algorithm for this project. I have also tried Support Vector Machine, logistic regression classifier, Decision Tree, Random Forest and Ada boost Algorithms. Logistic Regression Classifier has got the highest accuracy of 0.86, but its Recall is less than 0.3, other classifiers also have accuracy in the range 0.82 to 0.84 ,however Recall values are not above 0.3, with Naïve Bayes I got accuracy of 0.84, precision 0.386 and Recall 0.393, so I have chosen Naïve Bayes Algorithm

4. Tuning of an Algorithm means getting the best performance out of a model by using appropriate parameters of that model for that dataset, if we don't do this well, it can result in bad results, I did not tune parameters in this project as I have used Naïve Bayes algorithm which did not have parameters to tune. But initially I have tested Support Vector Machine. I have tuned its parameters: C, gamma and Kernel using GridSearchCV, I have taken list of random values for C, gamma and Kernel and passed them to the GridSearchCV along with Classifier and then I have validated this model by calculating Accuracy, precision and Recall, I tried different values and observed for the improvement in validation results

5. Validation means checking the performance of the model, if we don't validate well, we may end up overfitting the data. I have validated my data by splitting the whole data into 70% training data and 30% testing data using Stratified shuffle split, I have used this method because the poi classes are imbalanced. By fitting the training data into the model, I have predicted values using testing data and compared the actual values with predicted values, which helped my model from overfitting

6. The evaluation metrics chosen are Precision and Recall. The average Precision for Naïve Bayes algorithm on this data is 0.386, which means predicted POIs are 38.6% true out of all the predicted POI true values. and Average Recall is 0.393, which means predicted POIs are 39.3% true out of all the POI true values in the dataset