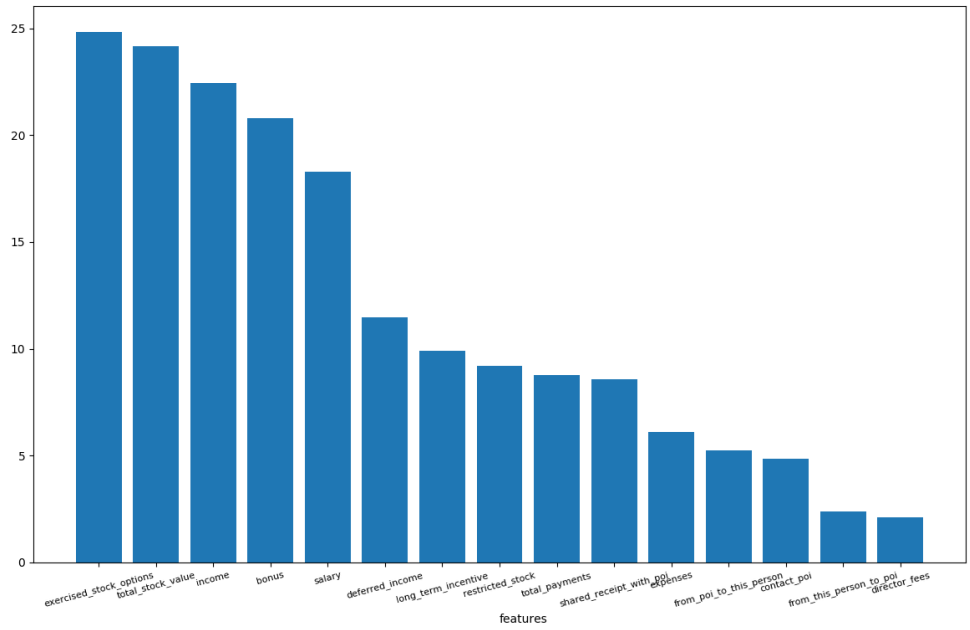
1. Summarize for us the goal of this project and how machine learning is useful in trying to accomplish it. As part of your answer, give some background on the dataset and how it can be used to answer the project question. Were there any outliers in the data when you got it, and how did you handle those?

   The goal of this project is to build an algorithm to identify Enron employee who was associated with fraud. Enron, which was one of largest company in US, was in scandal. Government started to investigate corporate fraud and decided to make private information of the company such as e-mail and financial data of the employee public.
   There were few outliers as “Total”, “The Travel Agency In The Park”, and “Lockhart Eugene E”. “Total” was not a person but a total for everyone in the list. “The Travel Agency In The Park” didn’t seemed to be a person and had most of the data missing. “Lockhart Eugene E” seemed to be a person but all the data was missing. I have taken out those three data from the dataset.
   Total number of data points:146
   allocation across classes (POI/non-POI):18/128

2. What features did you end up using in your POI identifier, and what selection process did you use to pick them? Did you have to do any scaling? Why or why not? As part of the assignment, you should attempt to engineer your own feature that does not come ready-made in the dataset -- explain what feature you tried to make, and the rationale behind it. (You do not necessarily have to use it in the final analysis, only engineer and test it.) In your feature selection step, if you used an algorithm like a decision tree, please also give the feature importances of the features that you use, and if you used an automated feature selection function like SelectKBest, please report the feature scores and reasons for your choice of parameter values.

   I have created few features as income, which is “salary” + “bonus”, and contact_poi, which is “from poi to this person” + “from this person to poi” to see if these could be helpful in explaining the data. From looking at the graph below, we could see that income has high value for select SelectKbest. I will SelectKbest method to decide which features will be used for building algorithm.



From looking at the graph we could notice sudden drop of value after salary. I will use the sudden drop as cut-off point and choose 'bonus', 'exercised_stock_options', 'total_stock_value', 'income', and ‘salary’ as featrues.

Exercise_stock_options	24.815079733218194
Total_stock_value	24.182898678566886
Income	22.455643114550313
bonus	20.792252047181538
salary	18.28968404340451

I have scaled all the features using min-max scaler. It is crucial to do feature scaling because all the features had different values. For example, income tend to be higher than contact_poi. In order to minimize such difference I have used min-max scaler.

3. What algorithm did you end up using? What other one(s) did you try? How did model performance differ between algorithms?

   I have decided to use decision tree as the algorithm since it gave the highest accuracy among algorithm that I have tested. I have used Gaussian Naïve Bayes(0.46), SVM(0.36), and Random Forest Classifier(0.435). However, decision tree had the highest accuracy of 0.513 among other algorithm.

4. What does it mean to tune the parameters of an algorithm, and what can happen if you don’t do this well? How did you tune the parameters of your particular algorithm? What parameters did you tune?

   Tuning parameters means making algorithm more accurate with given data. If we didn’t tune well, then we might have low accuracy with more bias. I have used

grid_search method to tune decision tree. From decision tree I have tested out max_depth, criterion, and min_samples_split. From the grid_search I have found out "min_samples_split=2, criterion= 'gini', max_depth= 20" had a highest accuracy.

5. What is validation, and what's a classic mistake you can make if you do it wrong? How did you validate your analysis?

   Validation is evaluating the performance of the algorithm on the given dataset by using the test data. The goal of validation is to have unbiased estimates of accuracy of the algorithm that has been built from the training set. We have divided the data into training set, which is for building algorithm, and test set, which is for evaluating the performance of the algorithm.
   Some of the mistake I could make is using same data to train the algorithm and test the performance of it. If such situation happens, then it might give a room for overfitting. Such overfitting algorithm might have a great performance on the test but might not be as great in real time.
   I have used cross validation method to check the performance of the algorithm. I have used 70% of dataset as training set while 30% dataset as test set. Since the data was skewed towards non-POI, I have used Stratified Shuffle Split method to prevent a situation where there might be no or small poi in training or test set. Thus, I have used training set to train and tune the algorithm, then used the test set to evaluate the accuracy of the test set.

6. Give at least 2 evaluation metrics and your average performance for each of them. Explain an interpretation of your metrics that says something human-understandable about your algorithm's performance.

   With a surprise, random forest classifier(0.857) had a higher accuracy compared to decision tree algorithm(0.816), which means random forest classifier predicted poi more than decision tree algorithm. Precision is how much algorithm had correctly predicted poi out of all instances where algorithm had declared someone is Poi.((T_p / (F_p+ T_p)) For precision, random forest classifier(0.398) had higher value than decision tree algorithm(0.309), which means random forest classifier had more accuracy on predicting actual poi among it had predicted as poi. Recall is how much algorithm had correctly predicted poi out of all the poi in the data. .((T_p / (F_n+ T_p)) For recall, random forest classifier(0.141) had lower value than decision tree algorithm(0.307), which means random forest classifier had less accuracy on predicting poi among people who are poi. From looking at the evaluation score I could notice that random forest classifier had a higher accuracy because it had high prediction value. However, from looking at the recall, it seems like random forest classifier does worse than decision tree algorithm in predicting poi, so I will choose decision tree algorithm which does a better job on predicting poi.