Investigating Fraud in Enron Dataset

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? [relevant rubric items: “data exploration”, “outlier investigation”]

The goal of this project is to use machine learning methods to construct a predictive model for identifying potential parties of fraud. These are termed “persons of interest” (POI). The Enron data set is comprised of email and financial data (E + F data set) collected and released as part of the US federal investigation into financial fraud.

The data set is comprised of:

- 146 records
- 18 of these points is labeled as a POI and 128 as non-POI
- Each point/person is associated with 21 features (14 financial, 6 email, 1 labeled)
- financial features: ['salary', 'deferral_payments', 'total_payments', 'loan_advances',
- 'bonus', 'restricted_stock_deferred', 'deferred_income', 'total_stock_value', 'expenses',
- 'exercised_stock_options', 'other', 'long_term_incentive', 'restricted_stock',
- 'director_fees'] (Units = USD)
- email features: ['to_messages', 'email_address', 'from_poi_to_this_person',
- 'from_messages', 'from_this_person_to_poi', 'poi', 'shared_receipt_with_poi'] (units =
- number of emails messages; except ‘email_address’, which is a text string)
- POI label: [‘poi’]

Different metrics are needed to evaluate the classifiers to decide the performance. The two selected for use in this project are Precision and Recall.

- Precision is the number of correct positive classifications divided by the total number of positive labels assigned. For this case, it is the fraction of persons of interest predicted by the algorithm that are truly persons of interest.

Precision = true positives / (true positives + false positives)

- Recall is the number of correct positive classifications divided by the number of positive instances that should have been identified. For this case, it is the fraction of the total number of persons of interesting the data that the classifier identifies.

Recall = true positives / (true positives + false negatives)

Precision is also known as positive predictive value while recall is called the sensitivity of the classifier. A combined measured of precision and recall is the F1 score.

F1 Score=2 (precision x recall) / (precision + recall)

Exploratory data analysis revealed few outliers

Salary :
LAY KENNETH L
SKILLING JEFFREY K
TOTAL
FREVERT MARK A
Bonus :
LAVORATO JOHN J
LAY KENNETH L
BELDEN TIMOTHY N
SKILLING JEFFREY K
TOTAL
ALLEN PHILLIP K

Only Total was not required and it is removed from the dataset.

Outlier:

TOTAL: Extreme outlier for numerical features, consisting of spreadsheet summary.

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. [relevant rubric items: “create new features”, “intelligently select features”, “properly scale features”]

In the dataset we have the number of emails sent to POI and received from POI. But having it as number will not give an equal comparison if some person sends or receives more emails. So we are creating the two new features: - fraction to POI and fraction from POI.

Comparing the results for the final chosen algorithm with and without our new engineered features, we get the following results:

	Precision	Recall	F1 Score	Accuracy
With Features	0.31994	0.3105	0.31515	0.82007
Without New Features	0.31823	0.3055	0.31173	0.82013

There is a slight improvement with the new features addition

3. What algorithm did you end up using? What other one(s) did you try? How did model performance differ between algorithms? [relevant rubric item: “pick an algorithm”]

Multiple Classifiers are tried for prediction. And the decision tree looked promising based on the below results.

Algorithm	Precision	Recall	F1 Score	Accuracy
Gaussian NB	0.23578	0.4	0.29668	0.74713
AdaBoost	0.39985	0.2705	0.3227	0.84886
Random Forest	0.51629	0.103	0.17174	0.86753
Decision Tree	0.31994	0.3105	0.31515	0.82007

I checked the feature importance for the DecisionTree and used SelectKBest to choose the features to keep. Additionally, I used GridSearchCV in combination with SelectKBest to find the optimal number of features to use. This will run through a number of k values and choose the one that yields the highest value according to a designated performance metric.

According to the grid search performed with SelectKBest with the number of features ranging from 1 to 21 (the number of features minus one), the **optimal number of features for the decision tree classifier is 19**. I can look at the scores assigned to the top performing features using the scores attribute of SelectKBest.

feature	score
exercised_stock_options	25.09754
total_stock_value	24.46765
bonus	21.06
salary	18.5757
fraction_to_poi	16.64171
deferred_income	11.59555

long_term_incentive	10.07246
restricted_stock	9.346701
total_payments	8.866722
shared_receipt_with_poi	8.746486
loan_advances	7.24273
expenses	6.234201
from_poi_to_this_person	5.344942
other	4.204971
fraction_from_poi	3.210762
from_this_person_to_poi	2.426508
director_fees	2.107656
to_messages	1.698824
deferral_payments	0.217059

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? (Some algorithms do not have parameters that you need to tune -- if this is the case for the one you picked, identify and briefly explain how you would have done it for the model that was not your final choice or a different model that does utilize parameter tuning, e.g. a decision tree classifier). [relevant rubric items: "discuss parameter tuning", "tune the algorithm"]

Tuning is important for the algorithm to use optimal parameters to get accurate predictions based on different means and also reduce timing for huge datasets.

There are numerous parameters that can be changed, each will have an effect in the way how algorithm makes decisions as it parses the features and creates the 'tree'. For example, **criterion** can either be set as 'gini' or 'entropy' which determines how the algorithm measures the quality of a split of a node, or in other words, which branch of the tree to take to arrive at the correct classification. ('gini' utilizies the gini impurity while 'entropy' maximizes the [information gain] at each branching(https://en.wikipedia.org/wiki/Information_gain_ratio). I will put both options in my parameter grid and let grid search determine which is the best. The other three parameters I will tune are **min_samples_split**, **max_depth**, and **max_features**. Grid Search will be directed by cross-validation with 10 splits of the data and the scoring criteria is specified as Recall.

Based on the best parameters given by GridSearchCV, I am classified the Decision Tree algorithm.

Result from GridsearchCV:

	Select K Best	Criterion	max depth	max features	max samples spit
Optimal	19	entropy	15	None	20

5. What is validation, and what's a classic mistake you can make if you do it wrong? How did you validate your analysis? [relevant rubric items: "discuss validation", "validation strategy"]

Validation in machine learning consists of evaluating a model using data that was not touched during the training process. A classic mistake is to ignore this rule, hence obtaining overly optimistic results due to overfitting the training data, but very poor performance on unseen data.

Validation is process of determining how the model performs. We have to break the data set into a test and training set. Training set has to be fit and prediction is done on the test set. I used the Cross Validation to split the data into Train and Test for better results.

As we have less observation to train and test the algorithms, in order to extract the most information from the data, the selected strategy to validate our model is Stratified Shuffle Split Cross-Validation.

This strategy effectively uses a series of train/validation/test set splits. In the inner loop, the score is approximately maximized by fitting a model to each training set, and then directly maximized in selecting (hyper) parameters over the validation set. In the outer loop, generalization error is estimated by averaging test set scores over several dataset splits. All sets are picked randomly, but keeping the same proportion of class labels.

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. [relevant rubric item: "usage of evaluation metrics"]

For classification algorithms, some of the most common evaluation metrics are accuracy, precision, recall and the f1 score.

- Accuracy shows the ratio between right classifications and the total number of predicted labels.
- Precision is the ratio of right classifications over all observations with a given predicted label. For example, the ratio of true POI's over all predicted POI's.
- Recall is the ratio of right classifications over all observations that are truly of a given class. For example, the ratio of observations correctly labeled POI over all true POI's.
- F1 is a way of balance precision and recall, and is given by the following formula:

$$F1 = 2 * (precision * recall) / (precision + recall)$$

Scores for the final selected model :

Algorithm	Accuracy	Precision	Recall	F1 Score
Decision Tree	0.8504	0.44179	0.463	0.45215