Summary

About Enron

“**Enron Corporation** (former New York Stock Exchange ticker symbol **ENE**) was an American energy, commodities, and services company based in Houston, Texas. Before its bankruptcy on December 2, 2001, Enron employed approximately 20,000 staff and was one of the world's major electricity, natural gas, communications, andpulp and paper companies, with claimed revenues of nearly $111 billion during 2000.[1] *Fortune* named Enron "America's Most Innovative Company" for six consecutive years.

At the end of 2001, it was revealed that its reported financial condition was sustained substantially by an institutionalized, systematic, and creatively planned accounting fraud, known since as the Enron scandal. Enron has since become a well-known example of willful corporate fraud and corruption. The scandal also brought into question the accounting practices and activities of many corporations in the United States” – Wikipedia

The goal of this project is to identify possible persons of interest (**POI**), as we have a dataset of combined financial data of the employees, data from their emails we, and labels marking the already identified POIs, machine learning will be used to recognize the patterns of POIs and identify possible POIs other than the already identified persons.

By doing some data exploration the data can be described as having:

- total number of data points 145
- allocation across classes (POI/non-POI) 18/126
- number of features 21
- feature having missing values as per the below table (email data are shaded in green)

  The missing value here have different meaning depending on the type of data the feature belong to; so for financial data missing would mean 0 value, and for email data it would mean that it is not available, for example as per below table there are 34 person who don’t have email addresses.

-

Feature	Number of missing
salary	51
to_messages	59
deferral_payments	107
total_payments	21
long_term_incentive	80
loan_advances	142
Bonus	64
restricted_stock	36
restricted_stock_deferred	128
total_stock_value	20
shared_receipt_with_poi	59
from_poi_to_this_person	59
exercised_stock_options	44
from_messages	59
Other	53
from_this_person_to_poi	59
deferred_income	97
Expenses	51
email_address	34
Director_fees	129

There are some outliers that were found during data exploration of the dataset; we can categorize them into two groups

- one which we will remove from the dataset (an entry called Total instead of person name that contained total values for all employees across every feature)
- The other that is indeed an outlier but it is part of the pattern to identify POIs and so it will be kept as is. (salary, total payment, bonuses of POIs i.e. Kenneth Lay, Jeffrey Skilling)

Two features were created based on the intuition that the ratio between the restricted stocks and the exercised stocks might provide indication that this person is POI; their use and importance were left to be determined by SelectKBest algorithm with the number of features being determined by GridSearch

- (f1) the ratio between restricted_stock_deferred and exercised_stock_options
- (f2) the ratio between restricted_stock and exercised_stock_options

To select features, SelectKBest is used to select and score each feature; the parameters were chosen by GridSearch algorithm that did exhaustive search for number of features starting from 3 up to all features.

The effect of the features is tested on the final model by running it once with the selected features and another time adding the engineered features to the list.

Final model without the engineered features result

Accuracy: 0.85464 Precision: 0.48876 Recall: 0.38050 F1: 0.42789 F2: 0.39814

Total predictions: 14000 True positives: 761 False positives: 796 False negatives: 1239

True negatives: 11204

Final model including the engineered features result

Accuracy: 0.29393 Precision: 0.15601 Recall: 0.89400 F1: 0.26566 F2: 0.45938

Total predictions: 14000 True positives: 1788 False positives: 9673 False negatives: 212

True negatives: 2327

While the recall value is much increased to 0.894 up from 0.38 the precision is decreased to only 0.15

As a result the engineered features will not be used.

Features scores:

18.575703268041778	salary
0.21705893033950563	deferral payments
8.8667215371077805	total payments
7.2427303965360172	loan advances
21.060001707536578	bonus
0.064984311723709831	restricted stock deferred
11.595547659732164	deferred income
24.467654047526391	total stock value
6.234201140506757	expenses
25.097541528735491	exercised stock options
4.2049708583014187	other
10.072454529369448	long term incentive
9.3467007910514379	restricted stock
2.1076559432760891	director fees
1.6988243485808538	to messages
5.3449415231473347	from poi to this person
0.16416449823428589	from messages
2.4265081272428799	from this person to poi
8.7464855321290802	shared receipt with poi
0.044305772230916196	f1
0.5963579428007415	f2

Feature scaling is also applied since one of the algorithms used can use Euclidean metric (K neatest neighbors), and the method of scaling was min/max

Four algorithms were tried to predict POI, their details and GridSearch parameters used for parameter tuning are listed below, parameter tuning is important here because it determines the flexibility of the model and the units of freedom it has while fitting the model, thus having less chance of over-fitting

The metrics used to evaluate the models are (precision, recall, accuracy, f1) in this project's context recall and f1 (f1 combines precision and recall) are the important metrics, because precision is the ratio of the POI correctly identified as POI and the sum of POI correctly identified as POI and POI incorrectly identified as non-POI

On the other hand, recall is the ratio of the POI correctly identified as POI and the sum of POI correctly identified as POI and non-POI incorrectly identified as POI. Because of that recall is of more importance in this context

It is important to validate the models on data different than the data used in fitting, to assess how the model will generalize on new data. Since the number of positive labels is low and so the errors will not accurately represent an assessment of the model, Cross-Validation will be used by implementing StratifiedShuffleSplit with 500 iterations and test size of 0.3

The highest results obtained were from Naïve Bayes classifier

The selected model: is Naïve Bayes classifier with features selected by SelectKBest with K value of 5

nb_pipe.set_params(selector__k= 5, selector__score_func= f_classif) which is evaluated as below

Accuracy: 0.85464 Precision: 0.48876 Recall: 0.38050 F1: 0.42789 F2: 0.39814

Below are the results of gridSearch on the four models and the results for each score used by gridSearch

Scoring used as parameter to GridSearch

Score: Precision Weighted

	Accuracy	Precision	Recall	F1	F2
AdaBoost	0.83553	0.34297	0.25500	0.29252	0.26879
AdaBoost 1000 Iterations	0.84833	0.39897	0.27150	0.32312	0.29003
KNN	0.85680	0.44061	0.27450	0.33826	0.29689
KNN 1000 Iterations	0.85680	0.44061	0.27450	0.33826	0.29689
KNN 1000 Iterations no engineered features	0.85680	0.44061	0.27450	0.33826	0.29689
Naïve Bayes	0.84267	0.38235	0.29250	0.33144	0.30693
NB 1000 iterations	0.85200	0.43134	0.34550	0.38368	0.35982
NB 1000 Iterations no engineered features	0.85047	0.42226	0.33000	0.37047	0.34508
RandomForests	0.87087	0.53122	0.26800	0.35626	0.29748

Score: Recall Weighted

	Accuracy	Precision	Recall	F1	F2
AdaBoost	0.84933	0.25655	0.06850	0.10813	0.08027
AdaBoost 1000 Iterations	0.85067	0.28495	0.07950	0.12432	0.09290
KNN	0.87707	0.66525	0.15700	0.25405	0.18532
KNN 1000 Iterations	0.87707	0.66525	0.15700	0.25405	0.18532
KNN 1000 Iterations engineered features	0.87707	0.66525	0.15700	0.25405	0.18532
Naïve Bayes	0.84160	0.36798	0.26200	0.30607	0.27801
NB 1000 iterations	0.85200	0.43134	0.34550	0.38368	0.35982
NB 1000 Iterations no engineered features	0.85047	0.42226	0.33000	0.37047	0.34508
RandomForests	0.87100	0.54057	0.21650	0.30918	0.24599

Score: Accuracy

	Accuracy	Precision	Recall	F1	F2
AdaBoost	0.84980	0.26617	0.07200	0.11334	0.08430
AdaBoost 1000 Iterations	0.85273	0.31306	0.08750	0.13677	0.10223
KNN	0.87707	0.66525	0.15700	0.25405	0.18532
KNN 1000 Iterations	0.87707	0.66525	0.15700	0.25405	0.18532
KNN 1000 Iterations no engineered features	0.87707	0.66525	0.15700	0.25405	0.18532
Naïve Bayes	0.85047	0.42226	0.33000	0.37047	0.34508
NB 1000 iterations	0.85047	0.42226	0.33000	0.37047	0.34508
NB 1000 Iterations no engineered features	0.85047	0.42226	0.33000	0.37047	0.34508
RandomForests	0.87220	0.55155	0.22200	0.31658	0.25213

Score: F1 Weighted

	Accuracy	Precision	Recall	F1	F2
AdaBoost	0.83467	0.34395	0.26450	0.29904	0.27731
AdaBoost 1000 Iterations	0.84360	0.37868	0.27000	0.31524	0.28644
KNN	0.85680	0.44061	0.27450	0.33826	0.29689
KNN 1000 Iterations	0.85680	0.44061	0.27450	0.33826	0.29689
KNN 1000 Iterations no engineered features	0.85680	0.44061	0.27450	0.33826	0.29689
Naïve Bayes	0.85200	0.43134	0.34550	0.38368	0.35982
NB 1000 iterations	0.85047	0.42226	0.33000	0.37047	0.34508
NB 1000 Iterations no engineered features	0.85047	0.42226	0.33000	0.37047	0.34508
RandomForests	0.86820	0.51179	0.24950	0.33546	0.27799