Q1) 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”]

Answer) The goal of this project is to identify persons of interests based on financial and email data.To achieve this we need to find out the features which help in correctly identifying persons of interest(POI). POIs are people who are indicted, settled without admitting guilt or testified in exchange for immunity.The enron dataset consists of financial data and email data corresponding to the employees along with a label identifying them as a POI.We would look at the distinguishing trends in features of POI’s and non POI’s and understand if there is any significant trend present to predict given a new point.To facilitate this, machine learning algorithms are applied.

Goal and Role of ML: We employ various learning algorithms on the data, the algorithms learn the behavior of the features on pois and non pois, they find patterns and trends which point towards each of these classes. The dataset is split into training and test sets, the training set for the algorithm to learn the behavior, the test set to validate the models we create.

DATA EXPLORATION

On exploring the data, I found that there is information about 146 people, the data is presented through 21 features, I can also see that the data from all POI’s is not contained in the dataset, from the 35 POI’s there is information of only 18 POI’s.

OUTLIER INVESTIGATION

From scatterplots, I found extreme points in the dataset.I then found out the maximum value for salary, traced back to which key this extreme value was pointing to and found out that this was ‘TOTAL’ ,this information will not contribute towards our goal of prediction so I deleted this entry from the dataset.

MISSING DATA

There are features in the dataset which consist of more than 50% of its entries as ‘Nan’, these features with high missing values are not being considered for analysis.

Q2.)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”, “properly scale features”, “intelligently select feature”]

Feature Selection: A decision tree is fit to the entire data and feature importances are extracted, the most important feature is exercised_stock_options ,features with high importances are selected

from the output. There are however a few features which are given no importance.I would still consider these features for analysis as intuitively they seem important for identification.

From intuition, I can say that few features such as salary,expenses will be important features because they are clearly distinguishable features, higher management employees who take decisions will have higher salaries and expenses therefore these features can be considered as identifiers.The feature selection algorithm also supports my intuition as expenses received positive value.

Feature Ranking:

```
Feature Ranking:
1 feature to_messages (0.0)
2 feature deferral_payments (0.0)
3 feature expenses (0.0649654207133)
4 feature deferred_income (0.0529100529101)
5 feature long_term_incentive (0.108843537415)
6 feature restricted_stock_deferred (0.0)
7 feature shared_receipt_with_poi (0.0577200577201)
8 feature loan_advances (0.0)
9 feature from_messages (0.0481000481)
10 feature other (0.0300094315356)
11 feature director_fees (0.0)
12 feature bonus (0.109451528031)
13 feature total_stock_value (0.0)
14 feature from_poi_to_this_person (0.0873015873016)
15 feature from_this_person_to_poi (0.00734312416555)
16 feature restricted_stock (0.120480432567)
17 feature salary (0.0)
18 feature total_payments (0.112874779541)
19 feature exercised_stock_options (0.2)
```

Features selected based on scores and intuition:

```
poi
exercised_stock_options (0.2)
restricted_stock 0.120480432567
total_payments 0.112874779541
bonus 0.109451528031
from_poi_to_this_person 0.0873015873016
shared_receipt_with_poi 0.0577200577201
other 0.0300094315356
salary 0.0
expenses 0.0649654207133
```

I have selected the above features based on their scores and the importance of the financial details they give about an employee which could help in identifying a POI.

Exercised Stock Options is certainly the highest scored and logically also makes sense as an employee with more income or undocumented income would place money into circulation and hence exercise more stocks.

Restricted Stock are the shares issued to an employee as part of their pay, this also logically fits the feature which can identify a POI. More restricted stock implies that the employee could be accumulating more than needed but legally.

Total payments- Higher payments also denotes that there is a lot of money movement which could be suspicious.

Bonus is probably the most logical feature, the employees particularly higher management staff could be laundering the money from the scam in form of hefty bonuses without anyone questioning them.

Expenses- Expenses could also be a feature as high expenditure could be a form of money laundering, where the POI's invested in ventures which could cover their tracks.

From_poi_to this_person – More email traffic than required could be helpful in finding a POI. As more communication than ususal is only conducted in case of business deals.

Shared_receipt_with_poi—the person could be involved indirectly with a POI which can be traced with this feature

Other- There could be other financial data about the employee which is not being captured in the features stated above which could be obtained from this feature, also it is statistically significant.

The other features in the list are not selected as they are statistically insignificant from the feature importances extracted using the decision tree i.e they are scored very less or even 0.0 There are however a few features which are scored higher but not included such as long_term_incentive which do not give much information in identifying a POI or are redundant features such as restricted_stock_deferred and total_stock_value which are captured by other features related to stocks.

Feature Engineering: Two new have been added to the dataset, these features are fraction of mails from a POI to an employee- fraction_from_poi and fraction of mails from an employee to POI. This seems as a good identifier as more traffic would indicate more communication and higher possibility of being a POI.The new features are also visualized to see if there are any patterns, there seems to be a cluster of POI's in the graph which might act as an identifier.The new features are also added to the final list and feature importances are extracted from the decision tree fit like before,though one of the features do not receive any importance I feel that these features might contribute so I would include these features in the final model.

Feature importances with the new features included

```
Feature Ranking:
1 feature exercised_stock_options (0.272206660442)
2 feature restricted_stock (0.0281712685074)
3 feature total_payments (0.0507936507937)
4 feature bonus (0.042328042328)
5 feature from_poi_to_this_person (0.0)
6 feature shared_receipt_with_poi (0.178159684927)
7 feature other (0.130612244898)
```

```
8 feature salary (0.042328042328)
9 feature expenses (0.119345984008)
(NEW)10 feature fraction_from_poi (0.0)
(NEW)11 feature fraction_to_poi (0.136054421769)
```

I would include the new features while selecting my classifier,once I have found out my best classifier I would exclude the new features and see the effect the new features have on the classification,I would look at the evaluation metrics.

Final Features List

```
poi
exercised_stock_options
restricted_stock
total_payments
bonus
from_poi_to_this_person
shared_receipt_with_poi
other
salary
expenses
fraction_from_poi
fraction_to_poi
```

Feature scaling is required, ranges with respect to each of the feature have been calculated, the ranges clearly show some of the features are recorded very high in magnitude which could dominate in the model which would be created later.So feature scaling will be applied while using the appropriate algorithm(algorithm which will be affected if data is not scaled).

Q3. 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"]

Gaussian Naive Bayes, Random Forests, AdaBoost, Decision Trees,K nearest neighbors and SVM algorithms were applied on the dataset. Decision Tree Classifier has performed well on all evaluation metrics compared to all the other algorithms(after tuning).The following table is a summary of the evaluation metrics of all the algorithms applied.

Number	Algorithm(classifier)	Accuracy	Precision	Recall
1	Random Forrest	0.857	0.392	0.121
2	AdaBoost	0.845	0.400	0.323
3	Decision Tree	0.831	0.361	0.347
4	K nearest neighbors	0.875	0.621	0.177
5	Gaussian Naïve Bayes	0.844	0.363	0.223

From the above results, I considered that AdaBoost, Decision Tree classifiers are the best,as all 3 evaluation metrics are good.All the functions called are using default settings,I would like to

change these parameters and deploy my models on the dataset.I would also like to scale the data and apply k nearest neighbors on the dataset to evaluate its performance on scaled data.

I would now tune the parameters of the algorithms to get better performance metrics and improve my models.

Q4. 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? (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 item: "tune the algorithm"]

Tuning parameters of an algorithm : The functions of different algorithms consider default settings and perform required analysis on a given dataset, the parameters are set to this default considering the best dataset on which the algorithm would provide optimal results, but sometimes depending on the nature of the data and dataset, these default parameters cannot perform the necessary tasks well and hence must be changed to other values.This modification of parameters is called tuning.Tuning parameters in a broad sense would refer to our requirement in the bias-variance tradeoff, if we want a robust algorithm or a highly accurate one which performs only on select datasets.Tuning parameters would push us to either of the sides in the tradeoff.

I tuned 3 classifiers and selected the best one out of the three classifiers.

1.AdaBoost

Parameters tuned were n_estimators(number of estimators), learning_rate , algorithm.There are 2 different algorithms which the classifier uses and different numerical values the parameters n_estimators and learning_rate can assume.To obtain the best combination a function called grid search is used.The function generates different combination of the parameters specified and fits these models on the dataset, it then searches for the best estimator (performs well on all evaluation metrics) and provides the model, this model when deployed on the dataset gives optimal results for these parameters.

Best parameters found for AdaBoost were

```
{'n_estimators': 30, 'learning_rate': 0.5, 'algorithm': 'SAMME'}
```

```
AdaBoostClassifier(algorithm='SAMME', base_estimator=None, learning_rate=0
.5,n_estimators=30, random_state=None)
```

2.Decision Tree

Parameters tuned

max_depth—maximum depth of the tree

Min_samples_split—minimum number of datapoints for further splits

Criterion—decision criteria on where to split data (gini or entropy)

Max_leaf_nodes—maximum nodes on leaf

The same procedure was followed here as in AdaBoost, gridsearch finds the best combination and the best combination is fit on the dataset to obtain optimal results.

Best parameters:

{'min_samples_split': 20, 'max_leaf_nodes': 15, 'criterion': 'entropy', 'max_depth': 3}

Model :

DecisionTreeClassifier(class_weight=None, criterion='entropy', max_depth=3,

max_features=None, max_leaf_nodes=15, min_impurity_split=1e-07,

min_samples_leaf=1, min_samples_split=20,

min_weight_fraction_leaf=0.0, presort=False, random_state=None,

splitter='best')

3.k nearest neighbours

I wanted to observe the affect of scaling on the features and the result of the classifier.I first employed scaling of the features using the min max scaler and then tuned the parameters.Pipeline in sklearn was used to carry this out in sequence.The pipeline function fits and transforms the features, then tunes parameters as did above and applies on the transformed dataset.

Parameters tuned:

1) n_neighbors – number of neighbors

2) weights – the weight to be given to surrounding points (uniform-equal or distance-farther points gets less weights)

3) algorithm—type of algorithm

Best Parameters:

{'knn__algorithm': 'auto', 'knn__weights': 'uniform', 'knn__n_neighbors': 5}

Model:

Pipeline(steps=[('scaler', MinMaxScaler(copy=True, feature_range=(0,1))),('knn', KNeighborsClassifier(algorithm='auto', leaf_size=30, metric='minkowski',

metric_params=None, n_jobs=1, n_neighbors=5, p=2,

weights='uniform'))])

Impact of factor scaling: Factor scaling reduced the performance of the algorithm on the dataset than contributing to it.So it can be concluded that factor scaling cannot be applied to the features in this dataset.

After tuning these models and observing the evaluation metrics all have performed better than the default parameters assumed by the functions when fit previously. Decision Tree classifier has performed well compared to the other 2 algorithms.The results will be discussed later.

Q5. 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 item: "validation strategy"]

The model created which learns a particular dataset must be employed on an useen data set and its performance must be evaluated.The process of checking the performance of model on a new data set is called validation.To achieve this, the original dataset itself is split into training and test sets, where the model is first applied on the training set, so that it learns all patterns,trends and behaviors among the features in the dataset.The model is then fit on the unseen test set and the performance of the model is measured based on the evaluation metrics.

A classic mistake would be applying the model on dataset and then checking its performance on a completely different dataset.Also splitting of the data would be crucial while performing validation,if all data points of one class are in the training set and all points of another class are in the test set,the model would perform poorly as it has learnt trends of only a single class.

Validation is necessary as we need to understand how our model functions on a dataset, if a new datapoint is introduced,the model must be robust enough to identify the class or value of it correctly.Validation also tells us if we are overfitting by giving near perfect results on training set and large error rate on the test set

I performed cross validation to validate the models I created. Cross validation is a type of k folds validation method but in this method,datapoints from one fold can be sent to another fold which is not possible in k folds validation.So if all points of a single class are in one bin, in cross validation they can be mixed so that the model can learn all classes in the dataset.This concept is called stratified shuffle split, where splits are made in the data and datapoints are shuffled among the splits.

Q6. 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"]

The results of the tuned algorithms are as follows

Serial no	Algorithm(classifier)	Accuracy	Precision	Recall
1	AdaBoost	0.873	0.54	0.29

2	K nearest neighbors	0.85	0.22	0.04
3	Decision Tree	0.86	0.47	0.49

The Decision Tree classifier performs well on all 3 evaluation metrics and thus I have opted for this model to identify persons of interest from the enron fraud dataset.

Final Model

DecisionTreeClassifier(class_weight=None, criterion='entropy', max_depth=3,

max_features=None, max_leaf_nodes=15, min_impurity_split=1e-07,

min_samples_leaf=1, min_samples_split=20,

min_weight_fraction_leaf=0.0, presort=False, random_state=None,

splitter='best')

Evaluation metrics

Precision- 0.47619-If identified by the model as a POI, there is a probability of 47.61% the employee is a POI

Recall-0.49—49% probability of identifying a POI given the employee is actually a POI

F1-0.483—this measure is a weighted average of precision and recall.A value of 1 is a good indicator and 0 is bad.

F2- 0.487- this measure weigh recall more than precision.Values reaching 1 are good and those closer to 0 are bad indicators

Total Predictions- 15000

True positives-983 POIs were identified as POIs by the model

False positives-1083 non POI's were tagged as POI's

True negatives-11917 non POI's were tagged as non POI's

False negatives-1017 non POI's were tagged as POI's

Impact of new features on classification

The evaluation metrics precision and recall reduced without the addition of the new features which clearly implies that the new features are important and contribute to identifying the POI's effectively.

	ACCURACY	PRECISION	RECALL
With new features	0.86	0.47	0.49
Without new features	0.82	0.30	0.24