Enron Submission Free-Response Questions

A critical part of machine learning is making sense of your analysis process and communicating it to others. The questions below will help us understand your decision-making process and allow us to give feedback on your project. Please answer each question; your answers should be about 1-2 paragraphs per question. If you find yourself writing much more than that, take a step back and see if you can simplify your response!

When your evaluator looks at your responses, he or she will use a specific list of rubric items to assess your answers. Here is the link to that rubric: [**Link**] Each question has one or more specific rubric items associated with it, so before you submit an answer, take a look at that part of the rubric. If your response does not meet expectations for all rubric points, you will be asked to revise and resubmit your project. Make sure that your responses are detailed enough that the evaluator will be able to understand the steps you took and your thought processes as you went through the data analysis.

Once you've submitted your responses, your coach will take a look and may ask a few more focused follow-up questions on one or more of your answers.

We can't wait to see what you've put together for this project!

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 build a model which can identify a fraud or a person's of interest identifier based on email and financial data.This dataset has been combined with a hand-generated list of person's of interest in the fraud case, which means individuals who were indicted, reached a settlement, or plea deal with the government, or testified in exchange for prosecution immunity.Using machine learning we can create a model using available features in enron data to learn and identify patterns from the data.

   The dataset has 146 data points ,14 financial features ,6 email features and 1 labeled feature(poi)

   Outliers are data points which have significantly differ values than the other data points that can be due to data entry errors,sensor malfunction and natural outlier.

In enron financial dataset we find four outliers using exploratory data analysis(EDA):

1. TOTAL
2. LOCKHART EUGENE E
3. THE TRAVEL AGENCY IN THE PARK
4. WROBEL BRUCE

After removing outliers from enron dataset we left with 142 financial data points
Yes there are many features in enron dataset with missing value for example restricted stock deferrals,bonus etc.

Name of features	Number of missing values
salary	51
bonus	64
long_term_incentive	80
deferral_income	97
deferral_payments	107
loan_advances	142
others	53
expenses	51
director_fees	129
total_payments	21
exercised_stock_options	44
restricted_stock	36
restricted_stock_deferral	128
total_stock_value	20
poi	0
from_poi_to_this_person	60

to_messages	60
from_messages	60
from_this_person_to_poi	60
shared_receipt_with_poi	0

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”]

   I tried many feature combinations but the best feature combination is :**poi,bonus, exercised_stock_options,fraction_to_poi,loan_advances.**Using this feature combination i am getting **87.63%** accuracy on the test set.
   Yes,I used **MinMax** scaling to convert all the features to a uniform scale. The motivation to use this scaling was to preserve zero entries in sparse data as is this dataset.

   Note: Not all algorithms require feature scaling. For example, the Decision Tree doesn't require scaling because it doesn't rely on the Euclidean distance between data points when making decisions.

   I created two new features: **fraction_from_poi**, which represents the ratio of the messages from POI to this person divided with all the messages sent to this person, and **fraction_to_poi**, which is the ratio from this person to POI divided with all messages from this person.

   Univariate feature selection works by selecting the best features based on univariate statistical tests. I used the univariate feature selection, SelectKBest, which selects the K highest scoring features based on a scoring function that returns univariate scores.

   Because our data is sparse I chose chi2 as the scoring function which is recommended for sparse data as it will deal with the data without making it dense, which scored the features in the following order.

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”]

I tried four supervised machine learning algorithms including GaussianNB, DecisionTreeClassifier, RandomForest and KNeighborNearest for accuracy ,recall and precision.But i am getting best performance through KNeighborNearest so i am end up with using KNeighborNearest algorithm to create my machine learning algorithm because by which we are achieving best accuracy ,precision and recall.

I tried these algorithms with different numbers of features. For example for best accuracy we are using poi , exercised_stock_options,bonus,fraction_to_poi,loan_advances.

Algorithms	Accuracy	Precision	Recall	F1-score
KNeighborNearestClassifier	0.87638	0.66781	0.39100	0.49322
RandomForestClassifier	0.84508	0.49472	0.32800	0.39447
DecisionTreeClassifier	0.79069	0.32074	0.32250	0.32162
GaussianNB	0.75192	0.22197	0.24450	0.23269

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 usually a trial-and-error process by which we can change some parameters (for example, the number of neighbors in KNNalgorithm), run the algorithm on the data again, then compare its performance on many validation sets in order to determine which set of parameters results in the most accurate model.
   Parameter tuning is an essential part of controlling the behavior of a machine learning model. If we don't tune our parameters correctly, our model produces suboptimal results, as they don't minimize the loss function. This means our model makes more errors.Scikit-learn has the GridsearchCV method which exhaustively generates candidates from a grid of parameter values and provides a dictionary with the best performing ones:

   I used GridsearchCV to calculate the best possible parameters for the KNN from the following grid of possible choices:

   params_grid = [{'algorithm':['auto', 'ball_tree', 'kd_tree', 'brute'], 'n_neighbors': [2,3,....10]}]

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"]

Training a model on a dataset and testing a model on the same dataset can create an illusion that our model is performing very well but it is actually the case of overfitting and our model will fail to generalize the unknown data points which it has not seen yet.
So to avoid the overfitting in the machine learning model we separate data in two different parts one is called training dataset and other is testing dataset .

When evaluating different parameters, there is still a risk of overfitting on the test set because the parameters can be tweaked until the estimator performs optimally. To solve this problem, yet another part of the dataset can be held out as a so-called: validation set training proceeds on the training set, after which evaluation is done on the validation set and final evaluation can be done on the test set.

A more sufficient validation method is called **cross-validation**. In the basic approach, called k fold cross validation, the training set is split into k smaller sets. A model is then trained using k-1 of the folds as training data and the resulting model is validated on the remaining 1 fold. The performance measure reported by k-fold cross-validation is then the average of the values computed in the loop.
This approach can be computationally expensive, but does not waste too much data.

Since our dataset is small I used cross-validation by calling the cross val score helper function from Scikit-learn with a number of 5 k folds.

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”]

Accuracy tells us how many times the model made correct predictions in the entire dataset. It does not give us any class-specific information like which class boundaries were learned well, where the model was more confused, etc.Precision-Recall is a useful measure of success of prediction when the classes are very imbalanced. In information retrieval, precision is a measure of result relevancy, while recall is a measure of how many truly relevant results are returned.

I used precision-recall evaluation metrics in this project to measure the model's performance.

Precision is defined as the number of true positives divided by the sum of the number of true positives and number of false positives.

Recall is defined as the number of true positives divided by the sum of true positives and false negatives.

Precision = tp/(tp+fp)
Recall = tp/(tp+fn)

Our model has a precision of .66781 means when it predicts 66% poi’s were actually labeled as poi’s.

Our model has a recall of .39100 means 39% of poi in the dataset were correctly identified.