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 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 coach evaluates 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 to the rubric

Each question has one or more specific rubric items associated with it, so before you submit an answer, take a look at that the rubric. If your response does not meet expectations, you will be asked to resubmit.

Once you've submitted your responses, your coach will take a look and 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"]

   Introduction: The Enron scandal, revealed in October 2001, eventually led to the bankruptcy of the Enron Corporation, an American energy company based in Houston, Texas, and the de facto dissolution of Arthur Andersen, which was one of the five largest audit and accountancypartnerships in the world.

Background of dataset: This dataset includes Enron email and financial data of a number of people and whether they are poi(person of interest). Total number of data point is 146 (len(data_dict)), among which the number of poi is 18 and non-poi is 128. There are 21 features used for every person in the dataset.

Goal: I try to use Enron email and financial data to predict whether a person is involved in the fraud (poi, person of interest). I used part of data as training sample to do machine learning, and then use the other part of data as testing sample to evaluate the validation.

Outliers: There is one outlier that obviously needs to be removed, the total line in the spreadsheet. It is not a information for one person, but a summary data. Therefore it should be removed. However, other outliers such as Ken Lay and Jeff Skilling, who made much more money than others, should be kept still, as they are actual person involved, and the relationship between high payment with poi should be further explored.

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 doesn't 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.) If you used an algorithm like a decision tree, please also give the feature importances of the features that you use. [relevant rubric items: "create new features", "properly scale features", "intelligently select feature"]

I use all the features except 'email)address' as identifier to start up.

The last two features, 'fraction_from_poi' and 'fraction_to_poi', are new features I created. 'fraction_from_poi' means the fraction of all emails to a person that were sent from a person of interest, and 'fraction_to_poi' means the fraction of all emails that a person sent that were addressed to persons of interest. The reason why I create these two features is that I assume that poi should have had more frequent email communication with

poi. The fraction of poi email may be a meaningful feature for predicting poi or not. I have tried to created more features from financial data, but no one is as straight forward or meaning full as them.

I deploy univariate feature selection and select five most important features. Before I decided to use 5 as k value, I had tried 2-10 for k, and when k=5, the performance is best. The final five most important features are salary, exercised_stock_options , bonus, total_stock_value, and fraction_to_poi.

I do not need to do scaling, as my algorithms are GaussianNB and Decision Tree, neither of which requires scaling.

In terms of decision tree algorithm, the feature importances of the features I use are:

[0.2808295 0.15751729 0.23774822 0.05180316 0.27210182], respectively for the five features above.

3. What algorithm did you end up using? What other one(s) did you try? [relevant rubric item: "pick an algorithm"]

I use GaussianNB and decision tree(default parameters).

	Accuracy	Precision	Recall	F1	F2
GaussionNB	0.33700	0.14879	0.84150	0.25287	0.43576
Decision Tree	0.81973	0.31724	0.30550	0.31126	0.0778

GaussianNB has lower precision and higher recall than decision tree, but the precision and recall for CaussianNB seem biased. Therefore I decide to choose decision tree as my 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? (Some algorithms don't 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 if you used, say, a decision tree classifier). [relevant rubric item: "tune the algorithm"]

In decision tree algorithm, I try to tune the min_samples_split, which means the minimum number of samples required to split an internal node.

Here are results:

min_samples_split	Precision	Recall
2	0.317	0.306
3	0.332	0.289
4	0.318	0.282
5	0.331	0.290

Best value for min_samples_split should be 2, with both precision and recall are higher than 0.3. Usually such a low value for this parameter may result in overfitting problem. However, considering the number of data is only 146, and most targets are not poi, high value for the parameter may lead to all the test samples are predicted as non-poi. Therefore min_samples_split could be 2.

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 item: "validation strategy"]

   Validation is for assessing how the results of a statistical analysis will generalize to an independent data set. It is mainly used in settings where the goal is prediction, and one wants to estimate how accurately a predictive model will perform in practice.

   If I use all of my data for training, it may result in overfitting problem. Therefore I split my data into training part and testing part. And I also use k fold to further improve validation. I have tried different k from 2-10, and when k equals to 6, the performance is best.

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"]

   According to question no.4, my best result after parameter tuning is precision 0.317, recall 0.306.

   Precision means, given a person who is considered to be a poi, there is a 31.7% possibility that this person is truly a poi; Recall means, if one person

is a poi, there is a 30.6% possibility that this person can be predict corrected as a poi.