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 identify Fraud from the emails that were taken from Enron. Most emails wouldn't contain an individual's financial information (salary, bonus) and the POIs that we have were already determined by looking at the financial information. To identify other POIs the main features to look at would be the to_emails and from_emails, but only running four features would not give us enough data to find other potential POIs, so all of the features that were provided were used in the classifiers.

The Classifier information that I found was the following:

Total Number of data points: 146

Number of POIs:18

Number of Non-POIs:128

Number of features: 20

The major outlier of the project was the 'TOTAL' for the Salaries, the payments and the loans, once that was removed it appeared that there were more outliers but when examining the data versus what the graph showed these outliers appeared to be POIs so they were not removed.

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”]

Looking at the data just because someone is emailing a POI or receiving an email from a POI doesn’t necessarily mean that they are part of the scandal. To narrow down potential other persons of interests I thought to look at the percentage of emails that were sent and received and compare that to the total number of emails. If most of a persons emails are being sent to or from a POI then the recipient/sender would have a higher chance of being involved in the scandal. Only using those features would ignore most of the data presented, so all of the provided features were still implemented into the classifiers in order to not artificially skew the data or create outliers that were otherwise not present.

I wrote two new features: percent_received_from_poi and percent_sent_to_poi to determine who the POIs were emailing and who was emailing the POIs.

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 ended up using Select K Best to determine the best features to use out of all features available including the features that I wrote. I used several classifiers to determine which classifier yielded the greatest results and out of the classifiers used (Naïve Bayes, SVM, K Nearest Neighbors and Linear Regression) the classifier that has the best performance was Naïve Bayes. The results of the classifiers were as follows:

   I ended up choosing the top four features to run the k_best function on because when looking at the data there were four features that had much higher scores than the others so it seemed logical to select those four features to fit and test.

Naive Bayes Classifier (without Tuning)

Mean of accuracy: 0.829

Mean of precision: 0.332

Mean of recall: 0.32

Mean of f1 score: 0.316919191919

K Nearest Neighbors Classifier (without Tuning)

Mean of accuracy: 0.872

Mean of precision: 0.467

Mean of recall: 0.16

Mean of f1 score: 0.230952380952

Mean of accuracy: 0.829

Mean of precision: 0.332

Mean of recall: 0.32

Mean of f1 score: 0.316919191919

K Nearest Neighbors Classifier (without Tuning)

Mean of accuracy: 0.872

Mean of precision: 0.467

Mean of recall: 0.16

Mean of f1 score: 0.230952380952

K Nearest Neighbors Classifier (with Tuning)

Mean of accuracy: 0.86

Mean of precision: 0.4

Mean of recall: 0.16

Mean of f1 score: 0.214682539683

SVC Classifier (without Tuning)

Mean of accuracy: 0.87

Mean of precision: 0.0

Mean of recall: 0.0

Mean of f1 score: 0.0

SVC Classifier (with Tuning)

Mean of accuracy: 0.874

Mean of precision: 0.25

Mean of recall: 0.06

Mean of f1 score: 0.0952380952381

Decision Tree Classifier (without Tuning)

Mean of accuracy: 0.806

Mean of precision: 0.321

Mean of recall: 0.44

Mean of f1 score: 0.366613386613

Decision Tree Classifier (with Tuning)

Mean of accuracy: 0.836

Mean of precision: 0.145

Mean of recall: 0.12

Mean of f1 score: 0.127777777778

Logistic Regression Classifier (without Tuning)

Mean of accuracy: 0.876

Mean of precision: 0.2

Mean of recall: 0.04

Mean of f1 score: 0.0666666666667

Logistic Regression Classifier (with Tuning)

Mean of accuracy: 0.875

Mean of precision: 0.217

Mean of recall: 0.094

Mean of f1 score: 0.128571428571

Best features selected and Scores:

{'bonus': 21.06000170753657, 'exercised_stock_options': 25.09754152873549, 'salary': 18.575703268041785, 'total_stock_value': 24.4676540475264}

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 the parameters of an algorithm means to change some of the variables that the data is compared against, number of features, how the data is analyzed and where to draw the fit line, such as SVM's kernel variable, linear was used in class but changing it to rbf could lead

to different scores. Tuning is also done to make sure that some variables carry more weight than others and to change their weights if the results seem to be grossly exaggerated one way or another (For example a classifier giving a score of 100%) Over tuning an algorithm can create false information. For example if you have a line of data and normal fitting isn't giving you the results you wanted, you can change the 'fit' parameters and force the data to fall in line with what you are looking for.

Feature scaling was used with the K Nearest Neighbor Classifier, the Decision Tree Classifier, the SVM classifier and the Regression Classifier. These four algorithms were scaled because the initial scores of these algorithms using the base parameters yielded precision and / or recall scores greater than 0.3. Part of this project was to tune or scale the algorithms to have a precision and recall score of 0.3, and to avoid over tuning which could skew the results.

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"]

   Simply put, validation is assessing that your algorithm is doing what you want it to do. In this analysis validation was performed using Precision and Recall. One mistake can be made if the validation is done wrong is all your data becomes incorrect. In the Eigenfaces example if the validation is done incorrectly the machine will match up the wrong faces with the wrong people and report that Tony Blair is George Bush. If this algorithm were put into practice to identify people then no one would be identified correctly.

   The final validation that was performed on the data was done with StratifiedShuffleSplit based on the advice that I received from the Udacity mentor to use provided validation algorithims.

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"]

   The metrics used were Precision and Recall, is the fraction of relevant instances among the retrieved instances, while **recall** is the fraction of the total amount of relevant instances that were actually retrieved. The lower that precision and recall scores means the less false positives and false negatives that you have. In this case the Precision and Recall are accurately measuring that a POI on our known POI list is being identified as a POI when we run other features through our classifiers, and known non-pois aren't being identified as POIs.

Resources used:

Sklearn.com Documentation

Stack Overflow

Udacity Mentor Knowledge Base

WGU Instructor Conference