Responses

February 18, 2018

Free Response Questions

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 any POIs for the datasets available using machine learning. The dataset contains compensation information as well as emails from Enron, the various people working for or involved with the company. Enron, in late 1990s and prior to it declaring bankruptsy, was one the largest companies in the US. It started out as an energy company (natural gas) but eventually become a conglomerate with divisions including commodity trading, fiber optic networking, and owning natural resources like water. Enron filed for bankruptsy in 2001. The massive fraud committed was discovered due to discrepancies in the company's accounting practices.

The data contains a part of the enron corpus which is "a large database of over 600,000 emails generated by 158 employees[1] of the Enron Corporation and acquired by the Federal Energy Regulatory Commission during its investigation after the company's collapse." The data set it includes many compensation categories including salary, bonus, expenses as well as email related categories, this looks like it was obtained through the lawsuit that was filed for investors.

In my initial inspection of the data (I decided after struggling, to put the dataset in a dataframe to view the information more easily). Some issues I noted are: many missing values, each feature has a different amount of avaialble data: there are any where between 4 and 146 data points depending on the feature being reviewed, there was a total row as well as a travel agency that needed to be removed. The minimal amount of datapoints made me realize I had to remove outliers very carefully and only when necessary. The reason I decided to remove the travel agency is even though it seems to be sketchy, since it is a company run by the chairman's sister, it did not participate in the fraud. There are some large values (like Kenneth Lay's compensation) but those are legitimate data points. I also have some concern about some of the data points that contain people who are not employees

To work through the data furtherthere is some minor wrangling and cleaning that will need to be done. There may need to be even further changes to the dataset as the project progresses.

`What features did you end up using in your POI identifier, and what selection process did you u`

I decided to use SelectKBest to find the best features but made some changes to the features before using the algorithm. After completing a review I decide to excluded Loan Advances and the Director Fees both had too few data points to be useful also these features are captured in the total payments. Also because all of these features were Not POI features I was concerned it could create a distortion running it in SelectKBest.

I next created a few features to ease my review of the data. I made an additional POI feature that turned the POI boolean into a categorical variable. I also created a percentage of emails to POIs and from POIs for each person in the data set. I decided that this would replace the features to emails, from emails, emails to poi, emails from poi as it showed more clearly something that would. After looking at the data I decided to create a feature which combines the salary and the total stock value to look at the total compensation. After working through the total compensation I realized at it double counted some of the stock value so I thought maybe it's better to compare divide total payments by total stock value. That would be a better feature. The reason this variable could be useful is that I wanted to look at the relationship between payments and total stock values by POIs vs Non POIs.

The features that were selected by SELECTKBest were 'salary', 'bonus', 'total_stock_value', 'exercised_stock_options', 'percentage_emails_to_poi'. I decided to limit the amount of features in the algorithm to 5 because I was concerned that too many features would cause overfitting. The pvalues of these features are really low which is a good sign.

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 started with the algorithm provided Naive Bayes. I actually had pretty decent performance on the first try. I then tried a decision tree classifier. The decision tree did not have as good of a performance as the other. Then I did the Random Forest Classifier which also did not a good perforamnce at all. The last algorithm I tried was the SVM and that also did not have a very good performance. Some tuning and parameters will need to change to see if I am able to get a higher score. Since my Naive Bayes had decent performance I will leave it as is but will definitely tune the other three to see if I can improve performance.

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 parameters means to adjusting parameters within the algorithm before you fit the model. If you do not tune properly the performance may deteriorate. Each algorithm was tuned based that had under a .3 performance. I started with the decision tree classifier I used the minimum samples split and changed the I increased the parameter from 2 to 10. While I limited the variables I only removed a few samples from the data. The original minimum samples split of 2 would have the tree continue to split until there are only two samples left which I believe would cause over fitting. Increase it to 10 most likely create a simpler boundary and it seems to have helped. I used trial by error to find what minimum samples split value worked. I also switched the criterion from from gini to entropy. Entropy is a measure of impurity in samples, it is used to help the decision tree find a new split point. This also seems to have helped improve the perforamce. For the random forest algorithm I tried to do some of the same changes to see if it would bring the same success. While it did improve the precision it did not do a good job improving the recall. I then added n_estimator and started to change that to see if that improved the performance. The support vector machine gave me the most difficulty in tuning it's parameters. The run time of the algorithm made it very difficult to test different parameters. I decided that changing the kernel from RBF to poly. This helped improve the score but I could not get it higher than .2. Due to performance issues I excluded it from my final submission. I would definitely not use this model.

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”]

Validation is the process of seperating your data into a testing and training set to assist you in evaluating your machine learning alogorithm. A classic mistake that you can make is that you measure the performance of the training set and not the testing set. I used some evaluation metrics and compared my testing set to my training set using these performance evaluators.

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”]

I will give my metrics for the best performing model. I used the accuracy, recall score, and precision to guage the performance of the model. My average precision for my top 3 performing models was .53 and my average recall was .54. The recall score is the percentage of POIs in the dataset that were identified correctly as POIs. The precision the percentage of POI predictions that are POIs.