Project 3 Identify Fraud from

Enron Email

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 dataset we use in this project is constructed by the email and financial data from senior manager of Enron. Some of them are related to the Enron Company Fraud.

The goal of this project is to find the PIO (Person of Interest) from the whole group. Features are extracted from both the email and financial data, and we can also construct new feature according to our understanding.

Some of the features are evaluated and selected. Different classifiers with verified algorithm will be built based on the selected features. The classifiers are used to label the each observation as PIO or not PIO. We will evaluate the performance of our classifier by precision, recall and F1 score.

During my exploration, I find the outlier “TOTAL” through visualization. Then I popped it out from the dataset.

1. **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”]**

Finally, i choosed these 4 features.

* salary
* exercised\_stock\_options
* bonus
* total\_stock\_value

At first, I delete the „email“ from the feature list. Because it is just the email addresses of the observations. That does not show any information for identification process.

I used *grid\_search* method to find the best parameter for my kBest selector.

The best parameter turns out to be *k=4.*

Then I feed the features data into the kBest selector again, to find the score of each feature. They are shown below:

|  |  |
| --- | --- |
| **salary** | 18.57570327 |
| **exercised\_stock\_options** | 25.09754153 |
| **bonus** | 21.06000171 |
| **total\_stock\_value** | 24.46765405 |
| deferred\_income | 11.59554766 |
| expenses | 6.23420114 |
| total\_payments | 8.86672154 |
| long\_term\_incentive | 10.07245453 |
| restricted\_stock | 9.34670079 |
| to\_messages | 1.69882435 |
| from\_poi\_to\_this\_person | 5.34494152 |
| from\_messages | 0.1641645 |
| from\_this\_person\_to\_poi | 2.42650813 |
| shared\_receipt\_with\_poi | 8.74648553 |
| stock\_to\_salary\_ratio | 0.11604758 |

I did not use feature scaling. Firstly, when I used scaling, the performance of my classifier didn’t get obviously better. Secondly, the unit of selected features are all USD dollars. It is meaningful unit in the real world. So the disparity may indicate the difference between the observations.

I created feature “**stock\_to\_salary\_ratio**”, because I didn’t know that, whether there is a relationship between the stock value and salary of each employee. I thought, maybe the Stock value is of more importance for POIs rather than non-POIs. But this ratio is little related to our investigation.

1. **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”]**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Accuray** | **Precision** | **Recall** | **F1** | **parameters** |
| **Naive Bayes** | **0.85053** | **0.42254** | **0.33000** | **0.37058** | **{'fselect\_\_k': 4}** |
| **KNeighbors** | **0.87613** | **0.55470** | **0.36000** | **0.43663** | **{'fselect\_\_k': 4,**  **'fselect\_\_score\_func': <function f\_classif at 0x000000000CE10F98>,**  **'knn\_\_algorithm': 'brute',**  **'knn\_\_leaf\_size': 1,**  **'knn\_\_n\_neighbors': 3,**  **'knn\_\_weights': 'distance'}** |
| **Decision tree** | **0.83000** | **0.34481** | **0.30550** | **0.32397** | **{'fselect\_\_k': 4,**  **'tree\_\_criterion': 'gini',**  **'tree\_\_max\_features': 2,**  **'tree\_\_min\_samples\_split': 3}** |

**I picked Kneighbors, because it has the best performance, no matter in which indicator.**

1. **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”]**
2. **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”]**
3. **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”]**