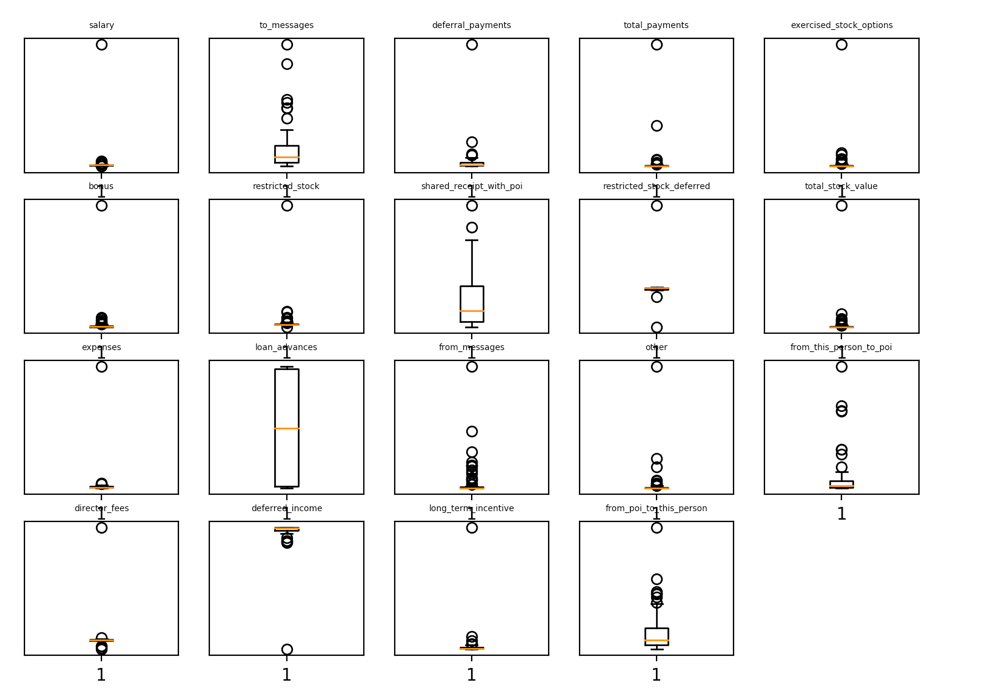
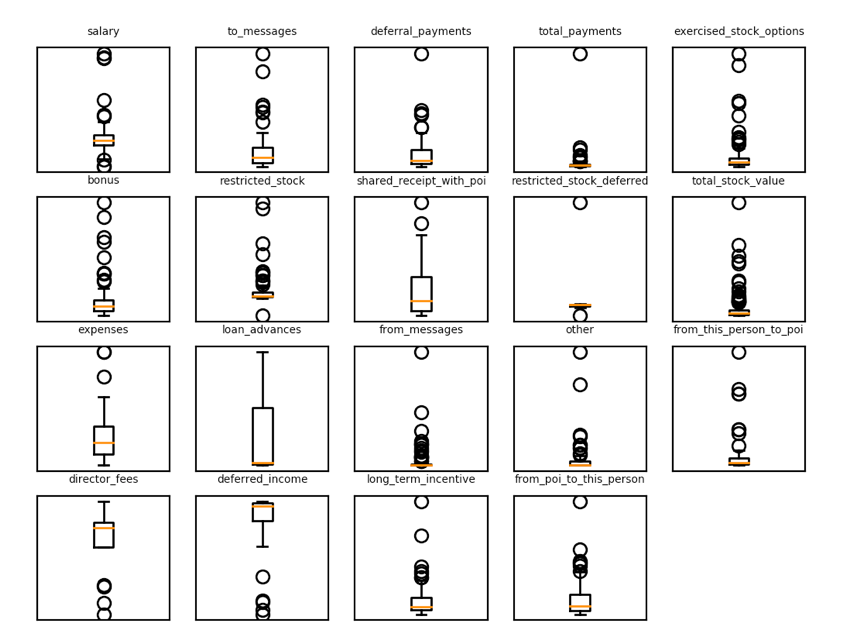
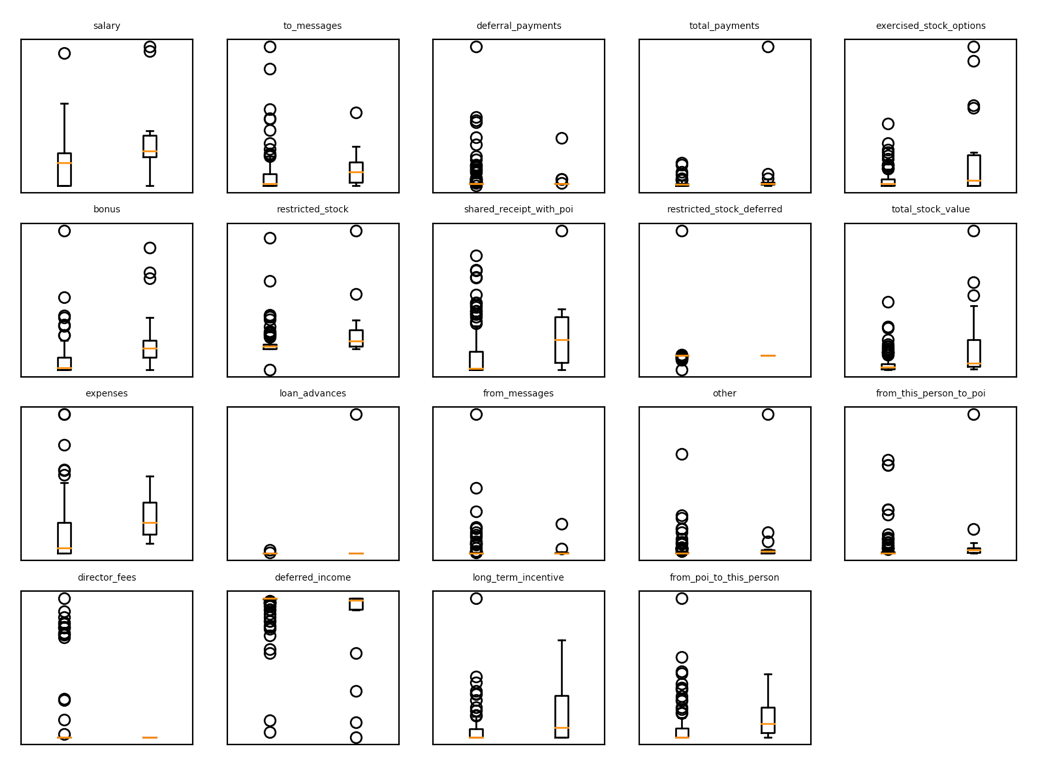
Introduction to Machine Learning – Final Project

1. Goal of this project – To identify who is the person of interest (POI) in Enron
2. Exploration of the datsaset:

* In this dataset, there are 146 people and there are two major categories of features, namely financial feature and email features. Financial features are all numerical values, including salary, deferral\_payments, total\_payments, loan\_advances, bonus, restricted\_stock\_deferred, deferred\_income, total\_stock\_value, expenses, exercised\_stock\_options, other, long\_term\_incentive, restricted\_stock and director\_fees. Similarly email features are also numerical values, including to\_messages, email\_address, from\_poi\_to\_this\_person, from\_messages, from\_this\_person\_to\_poi and shared\_receipt\_with\_poi. In sum, there are 19 features mentioned above regardless of the feature poi and the feature email\_address which stores the email address of the person.
* As to the target feature “poi”, in this dataset, there are 18 people who are POI and 126 who aren’t. So, it is an unbalanced dataset.
* Carefully, there is an outlier in this dataset, that is “TOTAL”, whose features represent the total sum of the values of those feature. From the box plot of each feature, it can be evidently seen. It should be removed in advance.
* 
* After removal, the boxplot would look like this one.



1. Feature Selection

* In order to know which feature can be used to identify the POI, the box plots of each feature which show the comparison between the POI and non-POI can be helpful.
* 
* From this plot, it can be speculated, the features, “salary”, “bonus”, “expenses”, “shared\_receipt\_with\_poi”, “from\_poi\_to\_this\_person” and exercised\_stock\_options” may be helpful in classifying POI and nonPOI. The reason is that the spreading of the values of these features between two groups may be slightly more different than that of other features.
* Another way to choose features is to use SelectKBest in sklearn, which will rank the importance of each feature according to the contribution to the accuracy. After running SelectKBest (input: all features), the scores are in the chart below. It can be suggested that “salary”, “bonus”, “expenses”, “restricted\_stock”, “load\_advances”, “deferred\_income”, “total\_payment”, “exercised\_stock\_options”, “long\_term\_incentive” and “shared\_receipt\_with\_poi” (10 features, which scored above 5) are helpful in classification. Also, these results support the feature selection by box plot.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| salary | 18.57 | bonus | 21.06 | expenses | 6.23 |
| restricted\_stock | 9.34 | loan\_advances | 7.24 | deferred\_income | 11.59 |
| director\_fees | 2.10 | total\_payments | 8.86 | restricted\_stock\_deferred | 0.06 |
| deferral\_payments | 0.21 | exercised\_stock\_options | 25.09 | long\_term\_incentive | 10.07 |
| other | 4.20 |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| from\_messages | 0.16 | to\_messages | 1.69 | shared\_receipt\_with\_poi | 8.76 |
| from\_poi\_to\_this\_person | 5.34 | from\_this\_person\_to\_poi | 2.42 |  |  |

1. Classification algorithm

* After feature selection, 10 features are used to build up the model for identifying POI. Here, I tried Naïve Bayes, Decision Tree, Adaboost Decision Tree, and Logistic Regression. I got the two best f1 score, which are 0.32 and 0.33, by using Naïve Bayes and Adaboost respectively.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Algorithm | Parameter | Accuracy | Precision | Recall | F1 |
| Naïve Bayes | N/A | .71 | .24 | .53 | .32 |
| Decision Tree | min\_samples\_split: 2  min\_samples\_leaf: 10 | .84 | .27 | .10 | .15 |
| Adaboost | n\_estimators: 55  learning\_rate: 0.1 | .86 | .46 | .25 | .33 |
| Logistic Regression | C: 5 | .78 | .17 | .16 | .16 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

* Also, I used PCA for all features, and get back 10 features from PCA for classification. With PCA, the two best f1 scores I got are from Naïve Bayes and Decision Tree.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Algorithm | Parameter | Accuracy | Precision | Recall | F1 |
| Naïve Bayes | N/A | .71 | .24 | .52 | .32 |
| Decision Tree | min\_samples\_split: 2  min\_samples\_leaf: 1 | .79 | .22 | .21 | .22 |
| Adaboost | n\_estimators: 25  learning\_rate: 0.04 | .87 | .63 | .10 | .18 |
| Logistic Regression | C: 5 | .78 | .17 | .16 | .16 |