Identifying Fraud from Enron Email

Data Analyst Nanodegree Project #5

Shalva Usubov

Project Overview

In 2000, Enron was one of the largest companies in the United States. By 2002, it had collapsed into bankruptcy due to widespread corporate fraud. In the resulting Federal investigation, there was a significant amount of typically confidential information entered into public record, including tens of thousands of emails and detailed financial data for top executives.

In this project, you will play detective, and put your new skills to use by building a person of interest identifier based on financial and email data made public as a result of the Enron scandal. To assist you in your detective work, we've combined this data with a hand-generated list of persons of interest in the fraud case, which means individuals who were indicted, reached a settlement, or plea deal with the government, or testified in exchange for prosecution immunity.

Introduction

Most important characteristics:

- 146 executives in Enron Dataset.
- 21 features ('salary', 'to_messages', 'deferral_payments', 'total_payments', 'exercised_stock_options', 'bonus', 'restricted_stock', 'shared_receipt_with_poi', 'restricted_stock_deferred', 'total_stock_value', 'expenses', 'loan_advances', 'from_messages', 'other', 'from_this_person_to_poi', 'poi', 'director_fees', 'deferred_income', 'long_term_incentive', 'email_address', 'from_poi_to_this_person')
- 18 out of 35 POI’s.

- “loan_advances” feature is almost empty only 4 of 146 executives have non ‘NaN value.’
- “director_fees” feature is almost empty only 17 of 146 executives have non ‘NaN value.’

The Enron Data

The first step of this project is examining the data for any outliers or obvious mistakes. There were few outliers:

1. “**TOTAL**” - that outlier arose from a spreadsheet summary line that got transcribed into the dataset. Was easy found by ploting “salary” and “bonus” features.
2. “**THE TRAVEL AGENCY IN THE PARK**” - that key does not looks like a executives, more looks like a company name.
3. “**LOCKHART EUGENE E**” - Only for this person all features is empty and he is not a POI executive, so I decide remove it.

Feature Processing

After I cleaned the data from outliers I start pick the most important features for our prediction model.

Two new features were created and tested for this project:

- “fraction_from_poi_email” - the fraction of all emails to a person that were sent from a person of interest
- “fraction_to_poi_email” - the fraction of all emails that a person sent that were addressed to persons of interest

The hypothesis behind these features was that there might be stronger email connections between POIs than between POIs and non-POIs, and a scatterplot of these two features suggests that there might be some truth to that hypothesis.

In order to find the most effective features for classification I used DecisionTree Classifier for find the most important features by use method “clf.feature_importances_”. I started form full list of features and then I removed all not important features based on score value and human intuition.

Example of testing output:

```
DecisionTree accuracy: 0.837209302326
precision = 0.333333333333
recall = 0.4
Feature Ranking:
1 feature salary (0.305039787798)
2 feature bonus (0.226348364279)
3 feature fraction_from_poi_email (0.106100795756)
4 feature fraction_to_poi_email (0.0925434718538)
5 feature deferral_payments (0.079575596817)
6 feature total_payments (0.0663129973475)
7 feature loan_advances (0.0663129973475)
8 feature restricted_stock_deferred (0.0577659888005)
9 feature deferred_income (0.0)
10 feature total_stock_value (0.0)
11 feature expenses (0.0)
12 feature exercised_stock_options (0.0)
13 feature long_term_incentive (0.0)
14 feature shared_receipt_with_poi (0.0)
15 feature restricted_stock (0.0)
16 feature director_fees (0.0)
```

After this step I picked 8 features with biggest score and model accuracy **0.837**:

```
features_list = ['salary', 'bonus', 'fraction_from_poi_email', 'fraction_to_poi_email',
'deferral_payments', 'total_payments', 'loan_advances',
'restricted_stock_deferred']
```

Precision and recall still “low”, so I decide manually pick features which give me the best precision and recall. For this particular dataset we should not use accuracy for

evaluating algorithm/model because there a few POI's in dataset and the right evaluator is precision and recall.

Finally I picked the following features: "fraction_from_poi_email", "fraction_to_poi_email", "shared_receipt_with_poi"

Scaling

I not used feature scaling for two algorithms that I used "Decision Tree" and "GaussianNB". Both algorithms ignore the relationship between features. Features in the model are scaled automatically depending on their assigned coefficients which renders feature scaling useless.

Algorithm Selection and Tuning

A decision tree classifier was selected as the algorithm for the POI identifier; Naive Bayes was also attempted.

The decision tree was selected over Naive Bayes because an examination of the Naive Bayes predictions over decision tree have smaller precision and recall, so I decided focus on tuning decision tree classifier for get higher precision and recall metrics.

After the algorithm and features were selected, the min_samples_split parameter of the decision tree was tuned by hand. The results of that tuning can be found in Table 1 below.

min_samples_split	precision	recall
2	0.5	0.25
10	0.5	0.25
20	0.43	0.75
25	0.42	0.75
average	0.46	0.5

Table 1: Precision and recall when tuning the min_samples_split parameter.
Since the best precision and recall were both found when min_samples_split=25, this was the clear choice for the value to set for this parameter.

The final model I chose had the highest overall performance of all 3 parameters (accuracy, precision, recall).

Accuracy: 0.89900 Precision: 0.56721 Recall: 0.38400

Analysis Validation and Performance

This process was validated using 3-fold cross-validation. The precision and recall are somewhat different for each of the 3 folds, so these numbers were averaged and the average precision and recall were used as the final metric to quantify the performance of the algorithm.

As listed in Table 1, the average precision was 0.46 and the average recall was 0.5.

Discussion and Conclusions

The precision can be interpreted as the likelihood that a person who is identified as a POI is actually a true POI; the fact that this is 0.46 means that using this identifier to flag POI's would result in 54% of the positive flags being false alarms.

Recall measures how likely it is that, given that there's a POI in the test set, this identifier would flag him or her; 50% of the time it would catch that person, and 50% of the time it wouldn't.

Possible path to improvement is digging into the emails data more. As example we have missed 17 emails of POI's executives, fill these data can improve performance of algorithm.