Enron Submission Free-Response Questions

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 purpose of this project is to choose and algorithm and fine tune it to get identification of Person of interest in Enron fraud. It is a supervised learning as we have the solution and we could check the error of the algorithm. Once we get a set of data, the levers that we have to get a good result are the features selection and definition, the classifier and the fine tune of the classifier.

The project result is the identification of who could be considered as person of interest in the investigation of Enron fraud. At the time of the fraud investigation some financial figures from employees were reveal as well as the emails in their accounts. The idea is to use the information related to financial compensations and emails exchange to define who is and who is not a person of interest for the investigation.

The key is the data quality and the data treatment before starting to submit the data set to the algorithm. We need to spend some time on data to understand what the matter is and to ensure the information we will pass to the algorithm will not induce bays, error or misleading results. We need to clean the data, identify the outliers and decide if we keep or remove them, define which features are useful and which ones we could create as combination of existing features. Finally we have to try different algorithms and fine tune them.

The initial data set contains 146 entries (thereof 18 Poi) and 21 features, and the proportion of NaN goes from 14% (total stock value and total payment) to 97% (loan advances). Regarding mails we have 24% missing emails address and 41% missing on exchange mails with poi.

The interpretation of NaN is different depending on the features. Financial features NaN means that the person have got 0 on that feature (see enron61702insiderpay.pdf). Those NaN will be considered as 0.

Regarding emails features, we could have use the option of imputing missing data but there will be difficult to define a figure to be accurate, mean does not mean much in this case as the emails amounts are not the real one but the result of a cleaning process. We will transform Nan in 0 and manage the entries depending on that.

The entry Total will be used to check the financial data and identify some mistakes in the financial data.

We identify some outlier in different features:

LAY KENNETH L is an outlier for total payment and total stock value as he was the founder, CEO and Chairman of Enron at that time. We not to be removed from our data set.

'KAMINSKI WINCENTY J' is an outlier as ‘from messages’ and 'SHAPIRO RICHARD S' and 'KEAN STEVEN J' are outlier in ‘to messages’, we will remove them from the data set.

The entries ‘TOTAL’, 'THE TRAVEL AGENCY IN THE PARK' , 'BELFER ROBERT', 'BHATNAGAR SANJAY', 'KAMINSKI WINCENTY J', 'SHAPIRO RICHARD S' and 'KEAN STEVEN J’ will be removed from the data set . New data set contain 139 entries and 21 features.

The total payment average is 2.2m$ with a Q3 on 1.9m$ (lower than the average) with a maximum value of 103.56m$. Total stock value has an average of 2.96m$ and a Q3 at 2.30 $ (lower than the average) with a maximum value of 49.1m$. That is thanks to our outlier K. Lay.

Regarding the amount of mails, we need to keep in mind that the emails has been cleaning but not depurate, that means there are mails not relevant for the investigation in the data set.

In a deeper analysis of the 18 poi, we found one outlier

DELAINEY DAVID W: more than 600 mails ‘from_this_person_to_poi’

And we found that 4 poi have no mails information. We decide to remove them from the data set as considered as noise value

FASTOW ANDREW S, KOPPER MICHAEL J , YEAGER F SCOTT and HIRKO JOSEPH

Final data set will have 14 poi with 100% of information related to mails exchanges

The final data set I will use contains 135 entries (thereof 14 Poi) and 21 features, and the proportion of NaN goes from 13% (total stock value) to 98% (loan advances). Regarding mails we have 24% missing emails address and 39% missing on exchange mails with poi.

The total payment average is 2.2m$ with a Q3 on 1.9m$ (lower than the average) with a maximum value of 103.56m$. Total stock value has an average of 2.70m$ and a Q3 at 2.25 $ (lower than the average) with a maximum value of 49.1m$. That is thanks to our outlier K. Lay.

For further details and coding see PML_1_data_adq.py and PML_2_data_chc.py

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”]

The features in the data fall into two types namely financial features and email features, on top we have the POI labels that is the one we need the algorithm predict properly

POI label: [‘poi’] (boolean, represented as integer)

financial features: ['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', 'director_fees'] (all units are in US dollars)

email features: ['to_messages', 'email_address', 'from_poi_to_this_person', 'from_messages', 'from_this_person_to_poi', 'shared_receipt_with_poi'] (units are number of emails messages; notable exception is ‘email_address’, which is a text string)

After removing the people in explained in question 1 I made some analysis of the features, their relations and correlations, and the information we have for the features, I take some assumptions to remove features

I remove all the features with more than 70% of Nan in the person that are Poi and at total data set : 'director_fees' (0%, 10%), 'restricted_stock_deferred' (0%, 89%), 'loan_advances' (93%, 98%), ‘deferral_payments' (71%, 73%)

I remove 'deferred_income' because that is a negative value that represent a discount in the payment and it is missing in 65% of the records

Then financial features remaining are: ['salary', 'bonus', 'expenses', 'other', 'total_payments', 'long_term_incentive', 'exercised_stock_options', 'restricted_stock', 'total_stock_value',]

Regarding the emails, this project use the dataset corresponding to the counting of mails and not gone to the details of the text in the emails. Taking into account that the total amount of emails for each person has been cleaned in a more or less proper way, I decided not to use that total amount but only the mails with relation with poi, that means send to or received from poi. On top, there are 40% of persons without mail.

The algorithm to determine feature importance will work only if the features we pass has a sense, it means if our data quality is bad, it could lead to a bias that provide result without any sense.

Machine can help a lot but we need to ensure data quality and common sense

Next step is create new features. Since the moment we decide to remove features we need to find a way to keep that information if we consider it is relevant, a good way is to combine those features in a way that we could ensure that.

In the case of payments and stock the new features we create will be focus in the extra payment the people get and its ratio over total money for the person:

"incentives" = bonus + long_term_incentive + exercised_stock_options

"total_money" = total_payments + total_stock_value;

"bonus_ratio" = bonus / total_money

"incentives_ratio" = incentives / total_money

I will pass all the features to different data selection algorithm to check if the result are in line with my analysis

Scaling: some algothim does not affected by scalation

For the other ones we will include the escalation in the pipe

Kmeans as unsupervised method

Algotime

As the objective is classify the entries in two groups (poi, non_poi) we need to sue classification instead of regression. SVM or K means are good examples, both of them need feature scaling

But scalin choice has to take into account what we are going to do with the data, we can scale financial figures among them or indiviudaly. If we want to use financial features and mails feautures to clusterize we need to scale them

Several way of scaling

Detailed analysis including and excluding outlier are available on
PML_3_features_fullview.ipynb

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”]

No scaling need for dec tree or linear regression

Sum and kmeans need it

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”]

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”]

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”]

My conclusion is that the machine learning could identify poi base on the information we give but the reason why some people has not been considered poi is more complex than the information we have. There were a lot of political implication and on top some people probably give information to the investigation on exchange of not being considered as poi. But that is my thought after reading the press and passing the data over machine learning getting once and again this person as poi being considered not poi