ECE 562 Machine Learning, Fall 2019
Instructor: Zafer Aydın
Computer Homework 3

Introduction

Download the template codes for Udacity's Intro to Machine Learning course by running the following command in a terminal

git clone https://github.com/udacity/ud120-projects.git

Submit your scripts as q1.py and/or a text document that includes your answers to the questions below through Canvas.

The original Enron fraud data is a big, messy and totally fascinating story about corporate malfeasance of nearly every imaginable type. The Enron email and financial datasets are also big, messy treasure troves of information, which become much more useful once you know your way around them a bit. In Udacity course the email and finance data are combined into a single dataset called Enron email + finance (E+F).

In this homework you will explore the aggregated Enron email + finance (E+F) dataset that could be loaded by running `explore_enron_data.py` under `datasets_questions` folder.

Assignment

1. How many data points (people) are in the dataset?

2. For each person, how many features are available?

3. The "poi" feature records whether the person is a person of interest, according to definition in Udacity course. How many POIs are there in the E+F dataset? In other words, count the number of entries in the dictionary where data[person_name]["poi"] == True

4. A list of all POI names is available (in ../final_project/poi_names.txt) and associated email addresses (in ../final_project/poi_email_addresses.py).

How many POI's were there total? (Use the names file, not the email addresses, since many folks have more than one address and a few didn't work for Enron, so their emails are not available.)

5. Like any dict of dicts, individual people/features can be accessed like so:

```
enron_data["LASTNAME FIRSTNAME"]["feature_name"]
```

or, sometimes

```
enron_data["LASTNAME FIRSTNAME MIDDLEINITIAL"]["feature_name"]
```

What is the total value of the stock belonging to James Prentice?

6. Like any dict of dicts, individual people/features can be accessed like so:

```
enron_data["LASTNAME FIRSTNAME"]["feature_name"]
```

How many email messages do we have from Wesley Colwell to persons of interest?

7. Like any dict of dicts, individual people/features can be accessed like so:

```
enron_data["LASTNAME FIRSTNAME"]["feature_name"]
```

or

```
enron_data["LASTNAME FIRSTNAME MIDDLEINITIAL"]["feature_name"]
```

What's the value of stock options exercised by Jeffrey K Skilling?

8. Of these three individuals (Kenneth L Lay, Jeffrey K Skilling and Andrew S Fastow), who took home the most money (largest value of "total_payments" feature)? How much money did that person get?

9. How many folks in this dataset have a quantified salary? What about a known email address?

10. As you saw a little while ago, not every POI has an entry in the dataset (e.g. Michael Krautz). That's because the dataset was created using the financial data you can find in final_project/enron61702insiderpay.pdf, which is missing some POI's (those absences propagated through to the final dataset). On the other hand, for many of these "missing" POI's, we do have emails.

While it would be straightforward to add these POI's and their email information to the E+F dataset, and just put "NaN" for their financial information, this could introduce a subtle problem. You will walk through that here.

How many people in the E+F dataset (as it currently exists) have "NaN" for their total payments? What percentage of people in the dataset as a whole is this?

11. How many POIs in the E+F dataset have "NaN" for their total payments? What percentage of POI's as a whole is this?

12. If a machine learning algorithm were to use total_payments as a feature, would you expect it to associate a "NaN" value with POIs or non-POIs?

13. If you added in, say, 10 more data points which were all POI's, and put "NaN" for the total payments for those folks, the numbers you just calculated would change. What is the new number of people of the dataset? What is the new number of folks with "NaN" for total payments?

14. What is the new number of POI's in the dataset? What is the new number of POI's with NaN for total_payments?

15. Once the new data points are added, do you think a supervised classification algorithm might interpret "NaN" for total_payments as a clue that someone is a POI?