Project C: Use Tweeter Data to predict the return of Stocks

Author: Weihao(Vincent) Li

Objects:

We are given four stock: IBM, AMZN, VZ, MSFT, TSLA and a month of tweeter data. With python and Spark, we are trying to discover the relationship between stock and tweeter data.

Preparation:

1. Install Spark and spark.
2. Acquired the data of full month online.
3. Chose the dictionary that is used for sentimental analysis.

D1. Data Exploration:

1. The whole data is three month, which is a small set of observation in terms of the evolution of the stock price. I discover that 11 days of the data are not the day before trading day. So, I decided not to analysis them for them simplicity (In practice, I need to find a good way to average two day weekends data to predict price on Monday).
2. Given 20 observations, I will not choose more than 3 features to avoid overfitting problems.
3. Instead of testing on the full month data, I chose to start with the daily tweeter data which is 1.65 G
4. Loading the twitter data, I discover that several field of interest they are "Text", "Favorite", "language", "Followers count" and" friends count"
5. After reading Tweeter's documentation, these data are created when people tweeted, so favorite count will always be 0. I assuming that people from region that do not speak English will have less impact on the stock price (Since analysis in united states will read mostly English-written test)
6. Followers count is an interesting feature, however, based on the type of tweeter - sender, follower count will have positive or negative effect, we cannot just take it into consideration without knowing whose is the senders and how is their reputation.
7. So, considering the time limited, I decided to choose analysis the "Text" field.

D2. Data Programing

1. The tweeter data is in .json.bz2 format, which can be read directly by spark.
2. I choose hash-table to record the features in a tweeter text. Hash-table is more intuitive to me since in the map-recue procedure, what we do is exactly mearing the dictionaries.

3. DATA, I use SQL in spark to select the English-tweeter and select the file of “text”. Noticing that text always contains URL which will have symbol like : “VZ”,”TSLA”(I found this in D1). I use regular expression package to remove the character other than English words.
4. In each of the rows (RDD in spark), I will check if the tick name is in the “Test” field.
5. If the tweeter mentions the ticker, I will give 1 “M” pointes to the ticker. And then check if it is positive(P+1)/negative(N+1) by the dictionary.
6. Scan all 30 days’ file and summary the dictionary on each day (10 + hours)

D3. Data Analysis

1. Use stock data on the 20 trading days and feature of each days before these 20 trading days to run a linear regression.
2. Parameters: I choose mention time M and positive count P. Since there are collinearity between P and N(negative count)
3. Run regression for all 5 tickers and get the betas.
4. Output the data as a python list.

D4. Data Insight.

1. Run through one day tweeter data and plug into the linear model to get the stock price.

PS: Among above steps, D1, D2 and D3 are in the jupyter file D4 is in the .py script.

Summary:

1. The linear model is good in the trading set, since there two few of observations.

Constrain:

1. Testing time is too long on local machine: It will be great if we can run on AWS.
2. Data set is too small. A lot of features are very interesting, however to avoid the overfitting problem, I cannot include in the data set.
3. Dec is a very special month. We have reason to believe that because December has Christmas and New year, the stock is influenced by much more variables that we need to take into considerations. We must analysis the whole year data to avoid this periodic problem.
4. I will not surprise if the model shows a large MSE on the new data.
5. Natural language analysis is terrible. In my project, I analysis the text on a single word. However, “Not great” and “Don’t buy” are very common. If we want to analysis the sentimental better, we should use some professional package rather than based on a dictionary.