Assignment 7 Report

Stock Trading Data

Mucun Tian

In this assignment, we are going to explore stock trading data. The goal of this assignment is to learn how to analyze time serial data and to predict future stock price and/or trend (rise or fall).

Datasets

Two data sets are used:

one is a "company list" dataset downloaded from NASDAQ Company List. This data set includes metadata (such as symbol, company name, sector, industry, etc.) for stocks.

the other one is the “ALPHA VANTAGE” stock dataset obtained from API https://www.alphavantage.co since I am not able to access Google Finance API. This dataset is the daily stock exchange price data including Date, Open, High, Low, Close, Volume.

I collected 189 stocks’ exchange data, each stock has about 500 data points in recent days.

Section 1

Question: What stocks do have the similar trending during a certain period?

Method: For each stock, we have five features: Close, High, Low, Open, Volume. So the values of these feature construct a feature vector for a stock in one day. We are going to combine several days' data together to form a long vector to represent a feature vector for a stock in several days. Then, across all stocks, a sphere K-Mean clustering algorithm is used to cluster stocks. Finally, stocks with similar trending are grouped together.

1. Randomly sampled 10 stocks to research. (Symbols: AMTD, AMZN, CNI, JBHT, NEA, POOL, PYPL, ROX, V, VCIT)

2. Time range is from 2017-11-01 to 2017-12-01, which means one month of exchange price data for a stock are aligned sequentially (e.x. Open1, High1, Low1, Close1, Volume1, Open2, High2, Low2, Close2, Volume2, ...) to form a feature vector for this stock.

3. Use skmeans and choose 5 clusters.

4. Results: cluster 1: AMTD, 2: AMZN, 3: JBHT, NEA, POOL, VCIT, 4: ROX, 5: CNI, PYPL, V.

The Sihouette metric is used to evaluate the quality of clustering.

The Sihouette is computed by:

$$s(i) = \frac{L(i) - A(i)}{Max\{L(i), A(i)\}}, -1 <= s(i) <= 1$$

- $L(i)$ is the smallest average distance of point i to all points of other clusters
- $A(i)$ is the average distance of i to other points in the cluster to which point i belongs

Basically, this metric is measure whether points in a cluster is well isolated from other clusters. The higher this value, the better is the cluster. Based on this metric, Cluster 3 has the maximal average Sihouette value, which means points in cluster 3 are best grouped.

Figure 1 shows the clustering performance in terms of Sihouette Value

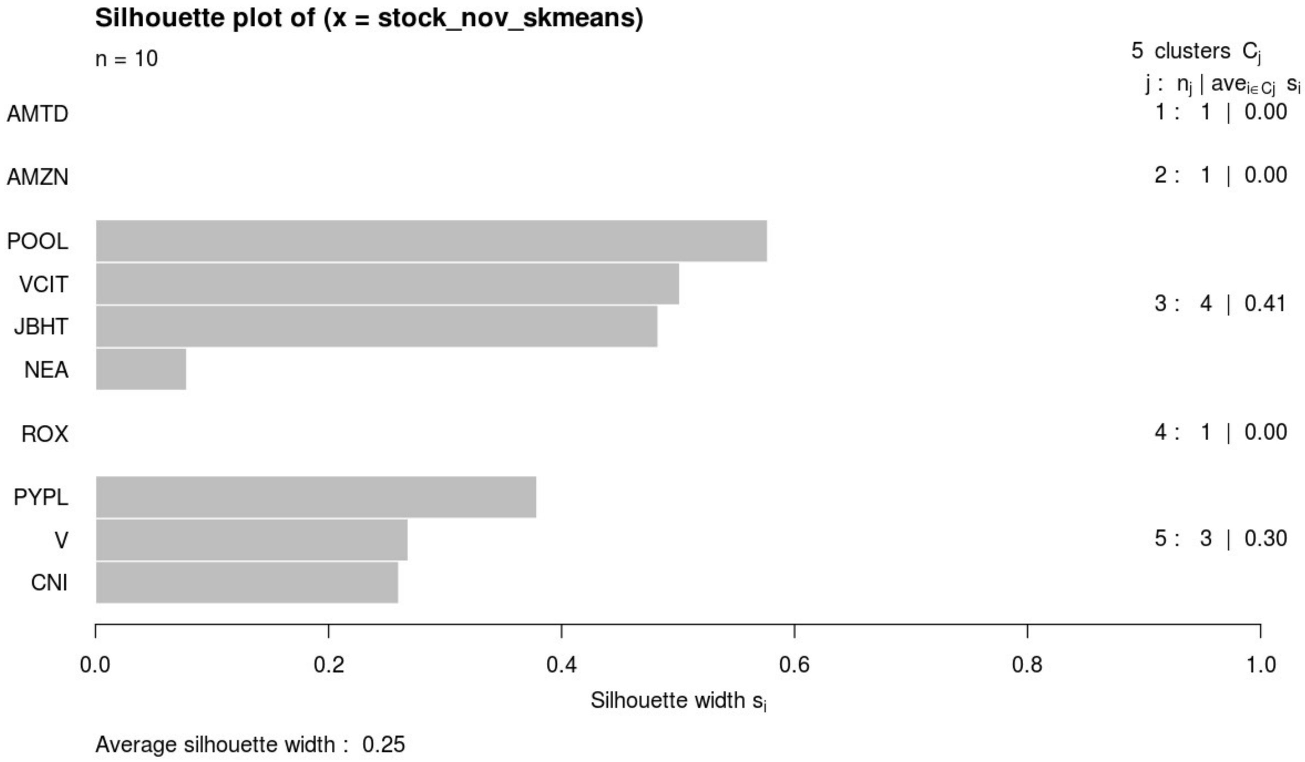


Figure 1

We can see that cluster 1 and cluster 3 and cluster 4 are well clustered.

Analysis Weekly

Do this skmeans analysis on a year period (2017)

1. create time window (weekly), each five days are grouped into one week.

2. in each week, cluster all sampled stocks using the same feature format as above, but the time is 5 days rather than one month.

3. select the cluster with the maximal average sihouette in each week.

4. use stocks in that optimal cluster to represent the most similar stocks in that week

So each week data, we have a best cluster with maximal average sihouette. Then, we rank this value over all weeks. The largest one is selected, and the price data of the stocks in the optimal week are plotted in Figure 2.

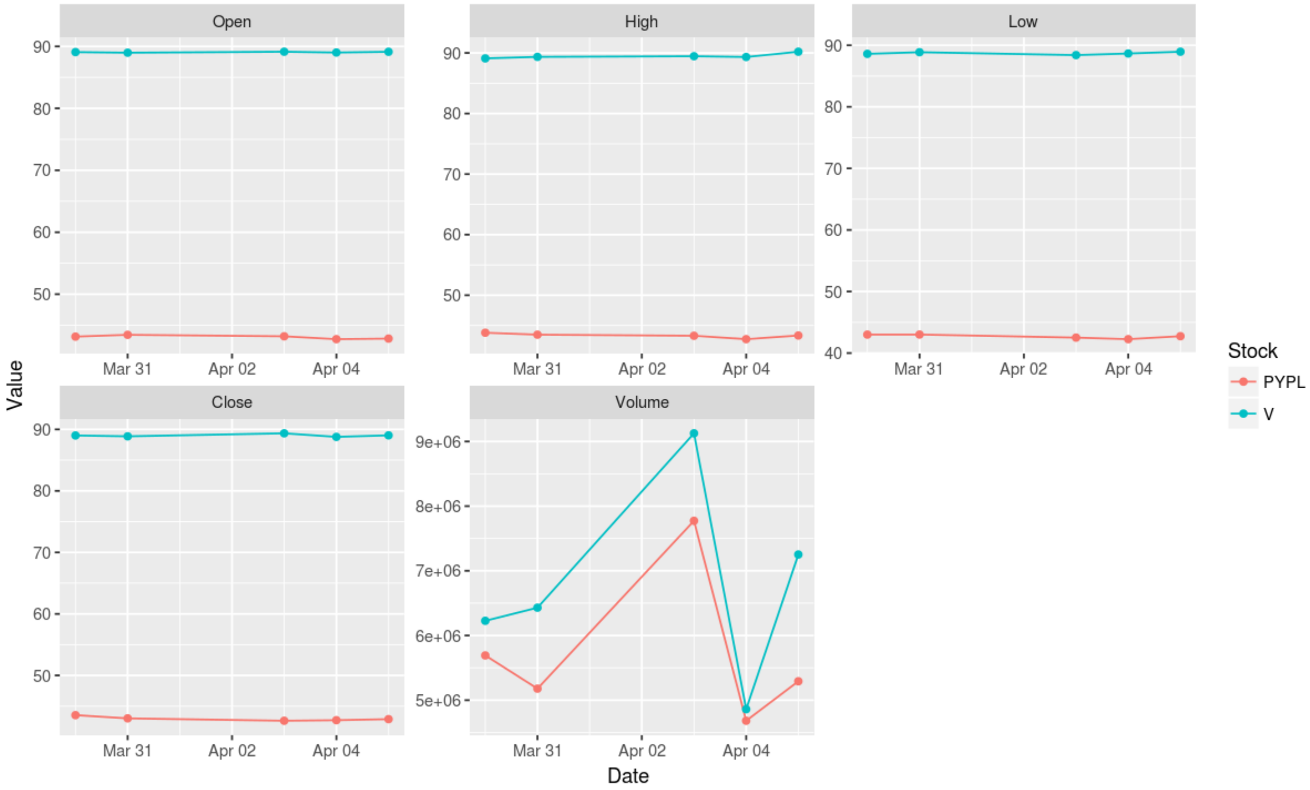


Figure 2

These two stocks are very stable! But they do close to each other in this week! Also the sKmean uses angle to evaluate the similarity, this makes the algorithm better capture the change of stocks rather than the amplitude.

Section 2

Question: What is the top 10 stocks which have the greatest increase or decrease during a certain period over all stocks?

Method: For a cetain period (Yearly, Monthly, Weekly), only considering the values of one of the features (like Close) at the beginning and last day, the relative difference of the values can be used to rank stocks.

The "Close Price" is selected and the greatest increase/decrease is computed by:

$$IncreaseRate = \frac{Close_{last} - Close_{first}}{Close_{first}}$$

Yearly

The top 10 stocks see the greatest increase in 2017 are: 'CCT', 'CTEK', 'DWDP', 'GLF', 'STRP', 'UQM', 'XXII', 'EVI', 'ISDR', 'BABA'.

The top 10 stocks see the greatest decrease in 2017 are: 'SE', 'LBY', 'GE', 'PDCE', 'CMCSA', 'VISI', 'BTN', 'BHB', 'BTI', 'BTX'.

Monthly

The same analysis is conducted for sampled 10 stocks and the top 2 stocks in each month are selected to plot.

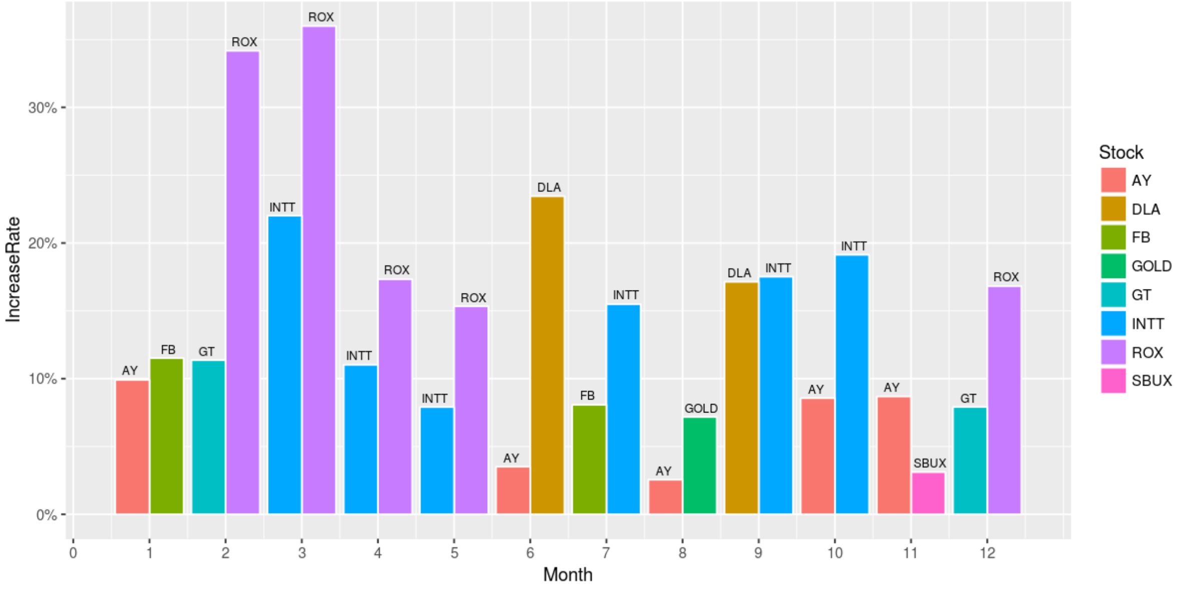


Figure 3 Top 2 Increased Stocks in Each Month in 2017

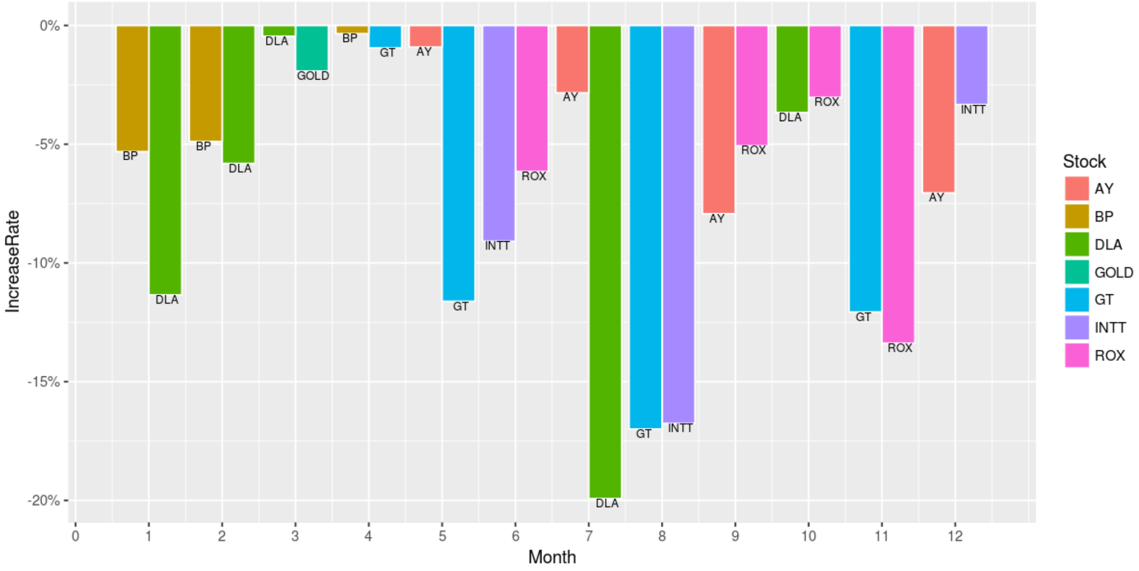


Figure 4 Top 2 Decreased Stocks in Each Month in 2017

Weekly

The Top 1 increased/decreased stocks in each week are plotted as following:

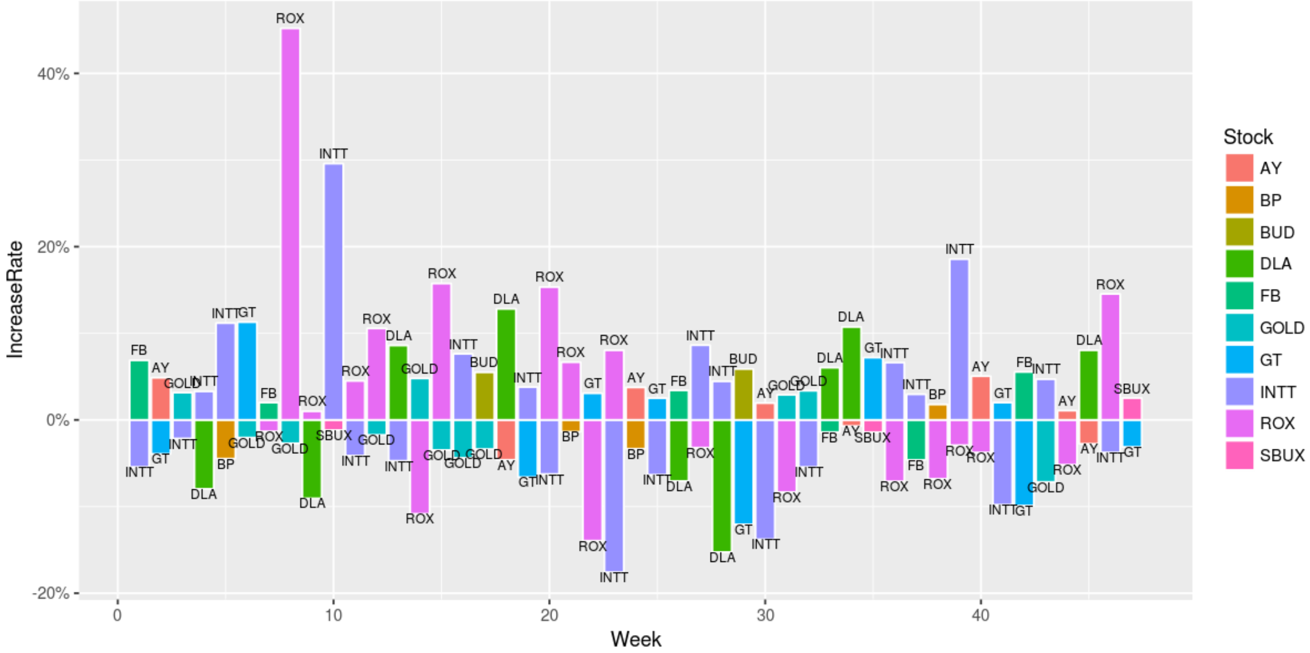


Figure 5 Top 1 Increased and Decreased Stocks in Each Week in 2017

Figure 5 shows that ROX and INTT are fluctuating heavily over all the year.

Section 3

Question: What is the probability of rise or fall after continuously rising or falling for a certain amount of days? Consider it over all stocks, stocks in each sector separately.

Method: For a certain number of days, finding all the cases of continuously rising (falling) in previous days, and rising or falling in the last day, then the probability of rise or fall can be computed by Conditional Probability.

$$P(R_{(n+1)th} | R_n) = \frac{P(R_{(n+1)th}, R_n)}{P(R_n)} = \frac{P(R_{n+1})}{P(R_n)} = \frac{n(R_{n+1})}{n(R_n)}$$

Where

- $P(R_{(n+1)th})$ means the probability of rise of a stock on $(n+1)th$ day,
- $P(R_n)$ means the probability of continously rise of a stock on next n days,
- $n(*)$ just the frequency of each corresponding case, since the denominators for both probabilities above are same - just the frequency of all cases (rise, fall, or not change) in next $n+1$ days

Over All Stocks

All cases in each stock are aggregated together to compute final probabilities.

Using all stocks, the probabilities are shown below. Based on this data, the maximal number of days of continuously increasing or decreasing for a stock that we found is only three.

Days	P(Rise | Rise)	P(Fall | Fall)	P(NotRise | Rise)	P(NotFall | Fall)
1	0.3543873	0.3320021	0.6456127	0.6679979
2	0.3275782	0.3033948	0.6724218	0.6966052
3	0.2415505	0.2023495	0.7584495	0.7976505

Table 1

For each row:

The P(Rise | Rise) means the probability of rise for next day, given continuously rising day days;

The P(Fall | Fall) means the probability of fall for next day, given continuously falling day days;

The P(NotRise | Rise) means the probability of not rise for next day, given continuously rising day days;

The P(NotFall | Fall) means the probability of not fall for next day, given continuously falling day days;

We can see the highest probability is the one that given continuously falling for 3 days, the next day does not fall! We can also compare these probabilities in each row. For example, from first row we can see that, given 1 day, the "falling for one day and not falling for next day" has the biggest probability; “the falling for one day and continuously falling for next day” has the smallest probability.

Over Each Sector

Do the same computation but using data in each sector. Table 2 is an example of three sectors’ given 3 days condition.

Sector	Days	P(Rise | Rise)	P(Fall | Fall)	P(NotRise | Rise)	P(NotFall | Fall)
Basic Industries	3	0.2449664	0.1992481	0.7550336	0.8007519
Technology	3	0.2371429	0.1901141	0.7628571	0.8098859
Transportation	3	0.2465278	0.1813725	0.7534722	0.8186275

Table 2

Roughly, we can see these probabilities are similar across different sectors.

Section 4

Objective: Predict next day's trend (rise or fall) using previous days' price and volume variables. Different variables will be tried.

Method: The training data format is processed in the same way as did in Section 1. For close price (either one of price features can be used), whether or not rise in the last day with respect to previous one day is used as label. Several previous days of Open, High, Low, Close, Volume, Sector, and Industry data are used as features. Then a binary classifier (like logistic classifier) is used to classifier using all stocks' data.

Parameter: each 5 days data are used to predict the change at sixth day. A moving window of five days is used to collect each 5 days data.

Data split: Training data are all data before 2017-11-1 (about 1 year), and Testing data are all the data from 2017-11-01 to 2017-12-08

Features: 1. only exchange features: Open, High, Low, Close, Volume.

2: Sector, Industry features (one-hot encodings) in addition to exchange features.

Model: 1. Logistic Model 2. Random Forest.

Four Combinations of two type of features and model are used.

Logistic Model

The following is the ROC plot for Logistic Model with two type of features

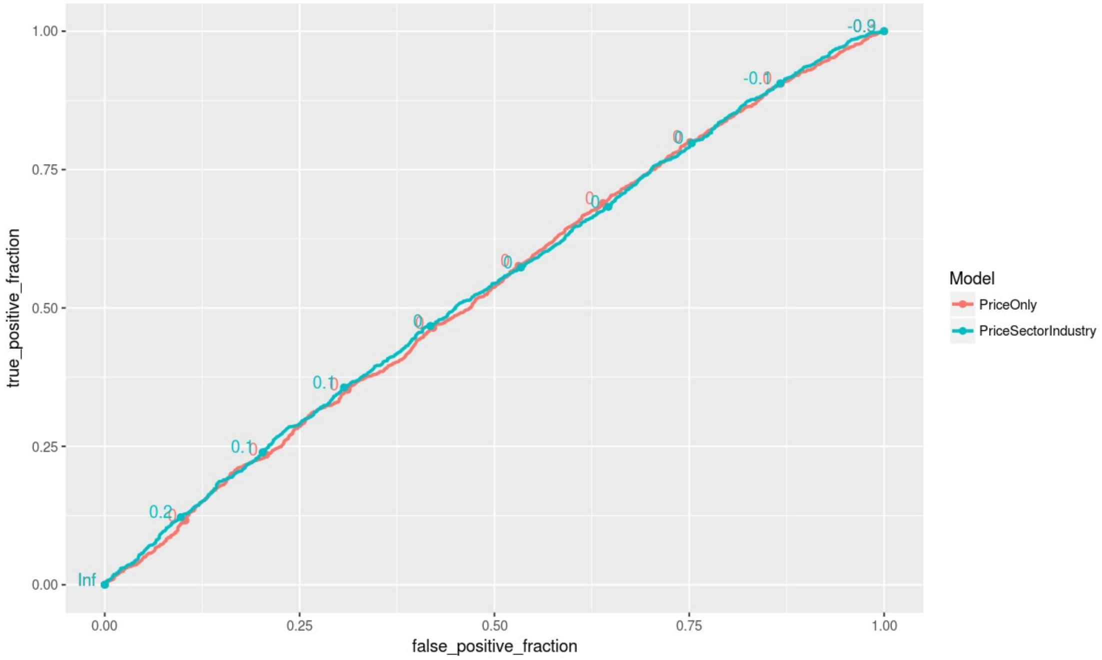


Figure 6

Figure 6 shows that the model almost did random performance, which means using this method is not able to predict the trend of stock price or the stock price is not predictable under this setting.

Model	Best Accuracy	Threshold
PriceOnly	0.5185632	0
Price Sector Industry	0.5214561	-0.1

Table 3 The best accuracy achieved by Logistic Model

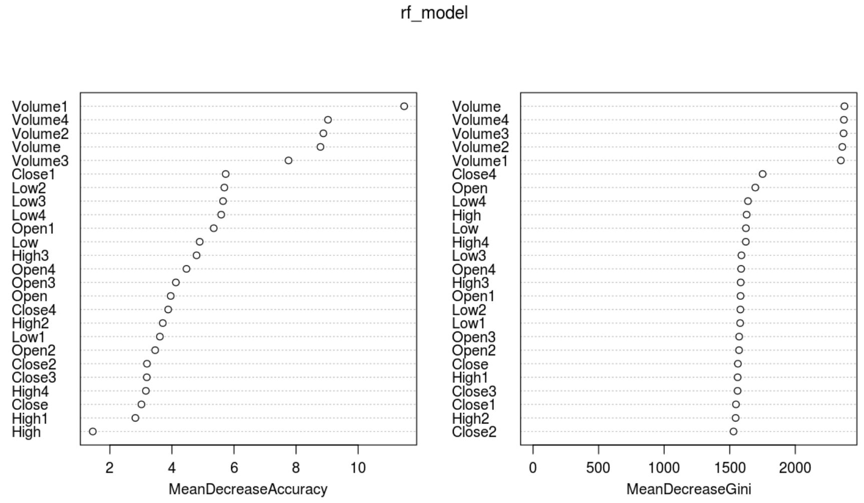
Random Forest

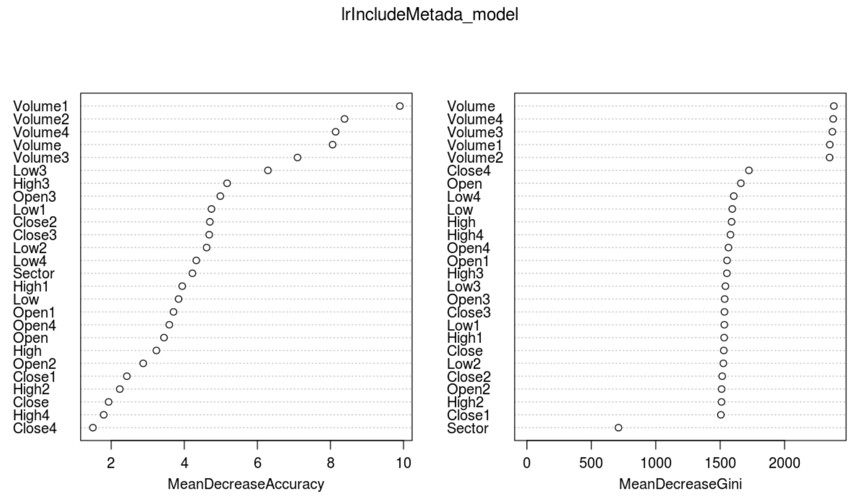
Parameter: 200 maximal trees

Model	Accuracy
PriceOnly	0.5113308
Price + Sector	0.5139826

The importance of variables are shown below. The higher value means the more important a variable is.

Figure 7 shows that volume variables are the most import variables for prediction accuracy.





Section 5

Objective: Predict a day's price using several previous days' features

Method: The data format is the same as section 5, but a regression model is used to predict the price on the last day

Data split: Training data are all data before 2017-11-1 (about 1 year), and Testing data are all the data from 2017-11-01 to 2017-12-08

Features: 1. only exchange features: Open, High, Low, Close, Volume.

2: Sector, Industry features (one-hot encodings) in addition to exchange features.

Model: 1. Lasso Linear Model, 2. Random Forest

Lasso Linear Model

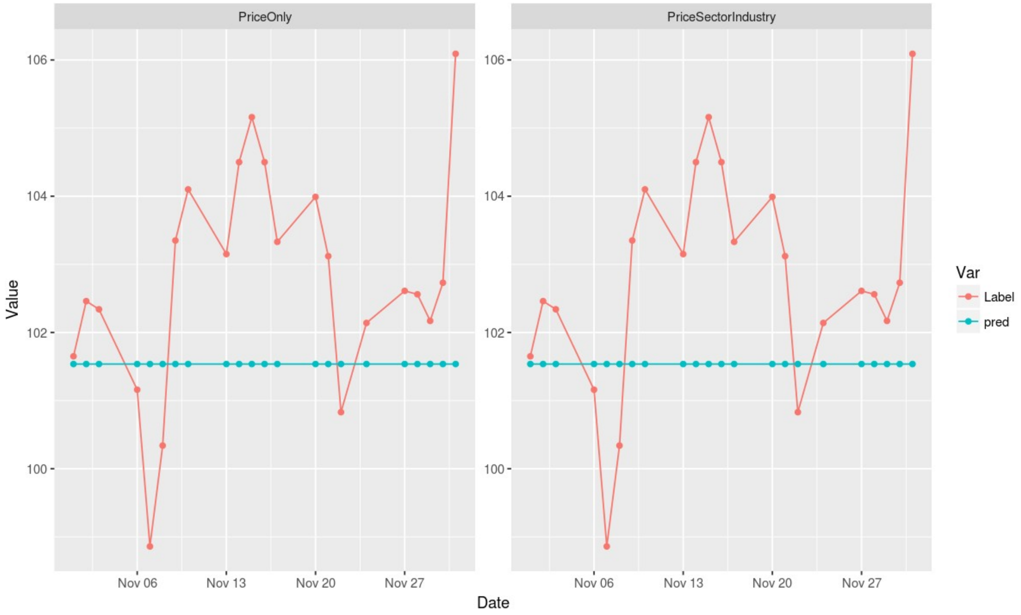
Whether or not including sector and industry information for each stock, we get the same results. This probably is because lasso regularization automatically excludes these two features.

Table 4 shows the best lambda based on RMSE of training data.

Model	Best Lambda	Best RMSE On Training	RMSE On Testing
Lasso	304.0338	304.0956	358.4687
Lasso + Metadata	304.0338	304.0956	358.4687

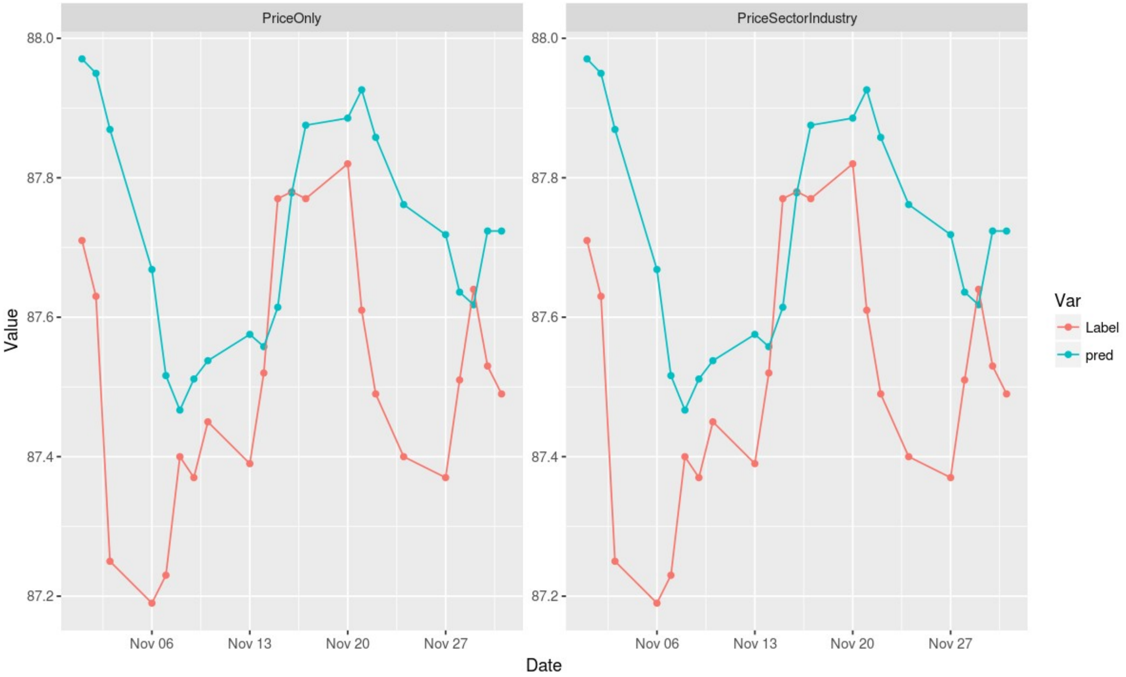
Table 4

After training, we do prediction for each stock in test data. The best predicted stock (CELG) based on RMSE (9.526057) is plotted below.



From this chart, we can see that the model probably predict everything to the same value. So selecting the best lambda based on RMSE of all training data may not be a proper way.

A cross validation is used to solve this problem. After cross validation, we do prediction for each stock in test data. The best predicted stock (VCIT) based on RMSE (0.2669452) is plotted below.



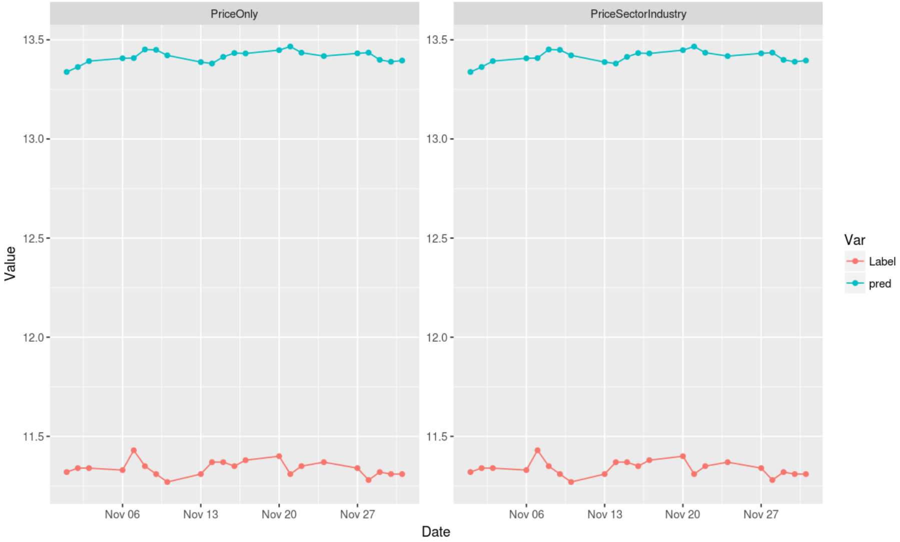
Random Forest

Parameter: in order to save time, we set node size to 20, max nodes to 256, tree number to 200.

Model	RMSE On Test data
PriceOnly	5.975770
Price + Sector	6.100396

Table 5

The best predicted stock (DNP) based on RMSE (0.05353517, 0.04580475 for each model respectively) is plotted below.


PriceOnly
PriceSectorIndustry
Value
Date
Nov 06
Nov 13
Nov 20
Nov 27
Var
Label
pred