Comparison of Predicting Stock Price Trend with Support Vector Machines and Random Forest

Xuelun Li
xl2678@columbia.edu

Zixiao Zhang
zz2500@columbia.edu

Abstract

Support Vector Machine has been proved to be a powerful predictive tool for stock predictions in the financial market. When combined with a non-linear kernel (e.g. RBF), it gains powerful ability to realize non-linear classification, so does Random Forest, which is also well known in dealing with non-linearity. Here, we make a comparison between these two models on the same problem to predict the stock price direction.

Key words: support vector machine, random forest, predict stock price, classification model

1. Introduction

For many years, the Efficient Market Hypothesis (EMH) is highly controversial and often disputed. It states that it is impossible to "beat the market" because stock market efficiency causes existing share prices to always incorporate and reflect all relevant information [1]. But we know that some investors, for example, Warren Buffet have consistently beaten the market over long periods of time, which is demonstrated impossible according to the EMH.

To test the predictability of stocks with the statistical learning model, we referred to the paper *Predicting Stock Price Direction using Support Vector Machines* [2]. In this paper, Saahil concludes in the long run the model has a better predictive ability than in the short run.

In our experiment, we also implement SVM with RBF kernel, plus a Random Forest for stock direction prediction, and compared their performance to predict the stock direction on a specific day. We conclude that it is better to use SVM to predict in the near future and it, in some sense, verify and reinforce the Efficient Market Hypothesis.

2. Related work

The use of prediction algorithms to determine future trends in stock market prices contradict EMH, where current stock prices fully reflect all the relevant information. But still, several algorithms have been implemented in stock prediction such as SVM, Neural Network, Linear Discriminant Analysis and Naive Bayesian Classifier. Some achievements have been made by these algorithms in the past a few years. In 2013, Zhen Hu, Jie Zhu, et al. introduced four company-specific and six macroeconomic factors and found that SVM is a powerful predictive tool for stock predictions in the financial market. And in 2016, Luckyson Khaidem, Snehanshu Saha, et al. using an ensemble learning method known as Random Forest to build predictive model and has produced really impressive results in predicting future direction of stock movement. [3]

3. Paper review

The paper we mainly referred was *Predicting Stock Price Direction using Support Vector Machines* [2]. Following is a brief summary of the paper.

3.1 Data

The data used by the paper was the historical NASDAQ-100 Technology Sector Index data as well as 34 stocks data out of the 39 in that index from 2007 to 2014, which can be obtained from Yahoo Finance and the CRSP stock database. The reason why they chose only one sector is that companies in one sector focus on a similar field, which to some extend normalizes the distribution of the whole dataset. After collecting all the data, they extracted features from the data and then partitioned the data as training data (from year 2007 to 2011) and testing data (from year 2012 to 2014).

3.2 Method

3.2.1 Feature selection

There are four features extracted from the raw dataset, which are index volatility, index momentum, stock volatility and stock momentum. Following are their definitions [2]:

Feature name	Description	Formula
σ_s	Stock price volatility. This is an average over the past n days of percent change in the given stock's price per day.	$\dfrac{\sum_{i=t-n+1}^{t} \frac{C_i - C_{i-1}}{C_{i-1}}}{n}$
Stock Momentum	This is an average of the given stock's momentum over the past n days. Each day is labeled 1 if closing price that day is higher than the day before, and -1 if the price is lower than the day before.	$\dfrac{\sum_{i=t-n+1}^{t} y}{n}$
σ_i	Index volatility. This is an average over the past n days of percent change in the index's price per day.	$\dfrac{\sum_{i=t-n+1}^{t} \frac{I_i - I_{i-1}}{I_{i-1}}}{n}$
Index Momentum	This is an average of the index's momentum over the past n days. Each day is labeled 1 if closing price that day is higher than the day before, and -1 if the price is lower than the day before.	$\dfrac{\sum_{i=t-n+1}^{t} d}{n}$

Table 1. Four features extracted

C_t and I_t are stock closing price and index closing price on t-th day respectively, y and d are both binary indicator variables (y, d ∈ {-1, 1}) represent the directional change of stock and index on a given day. The features then are fitted into a model to predict the direction of price change between day t and day t + m where m ∈ {1, 5, 10, 20, 90, 270}.

3.2.2 Model selection

The classification model they implemented was SVM model with radial basis kernel. Originally, the SVM functions to create an optimal decision boundary that maximizes the distance from any points to it and best splits the data into two classes. And the kernel is used to map the features into high dimensions in order to make it possible for linearly inseparable dataset. Compared with ANN, SVM is more tractable and interpretable, which is an important property when people want to dive into and investigate the decision steps of a model.

3.2.3 Model training

To predict the trend on the m day in the future, they used the four features discussed above, with different combination of n1 and n2 as parameter n for index and stock respectively. Since the calculation of the volatility and momentum needs to use the information of max(n1, n2) days earlier, only the days after the d=max(n1, n2) -th date can be predicted. Moreover, the direction of the last m days is not able to be predicted as there is no extra price information after the last m days to for calculating the output labels of them. So, there are 2014 trading days from 2007 to 2014 in total, and the dataset can only be created with feature vectors containing (2014-d-m) trading days, each combination of n1, n2 and m for each model. And the output trend vector y is the price direction corresponding to different m.

To simplify, they selected m from set {1, 5, 10, 20, 90, 270} which represents one day, one week, two weeks, one month, one quarter and one year of trading. n1 and n2 can only be chosen from {5, 10, 20, 90, 270}. With totally 150 combinations of (m, n1, n2), they constructed 150 SVM models for each one of them and compared the performances of their mean prediction accuracy.

3.3 Result

Figure 1 shows the mean prediction accuracy of SVM model they trained against different selection of m, where mean prediction accuracy here is the mean of the mean accuracies for all 25 combinations of n1, n2 with a fixed m. The most striking part is that the mean accuracy increases when m = 5, 10, 20 but then decrease when m = 90 and m = 270.

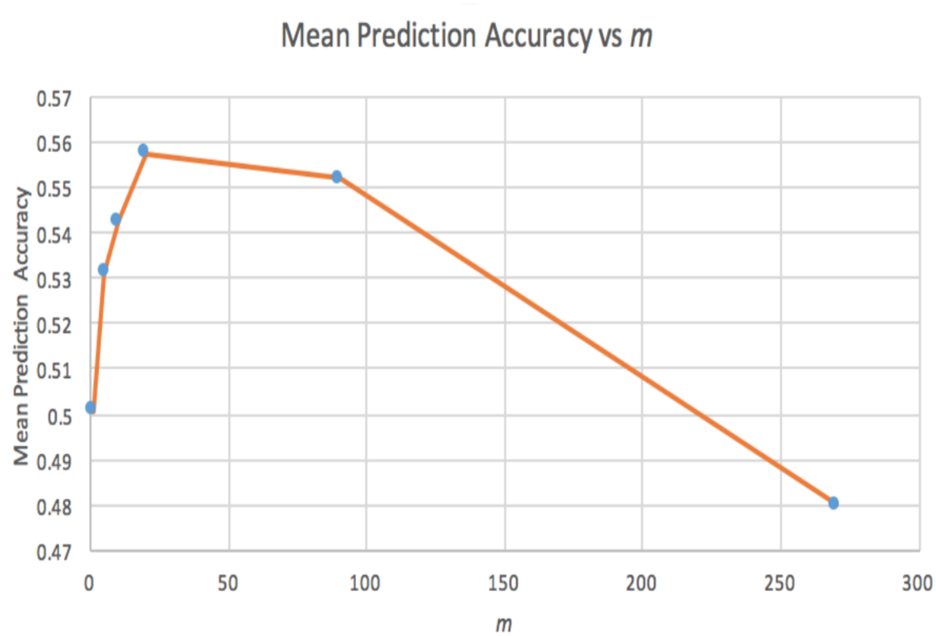


Figure 1: The mean prediction accuracy for each parameter m

By comparing the mean accuracy of different models, they conclude the model cannot predict next day's stock price direction better than random guessing (since the accuracy is almost 50%), which strongly reinforces the Efficient Markets Hypothesis. Additionally, varying n1 and n2 has little effect on the result when m is small, however, the importance of n1 and n2 increase as m increases. When m=20, the result reaches the highest mean prediction accuracy to above 0.55. These results show that when the model is going to forecast farther into the future, the historical data and extracted features become more influential. On the other hand, the standard deviation of accuracy becomes larger as m increases, for example, the model can predict price direction for some stocks with greater than 80% accuracy, but for some others it can only give no more than 30% accuracy.

Figure 2 shows when m=90, the prediction accuracy with different combinations of n1 and n2.

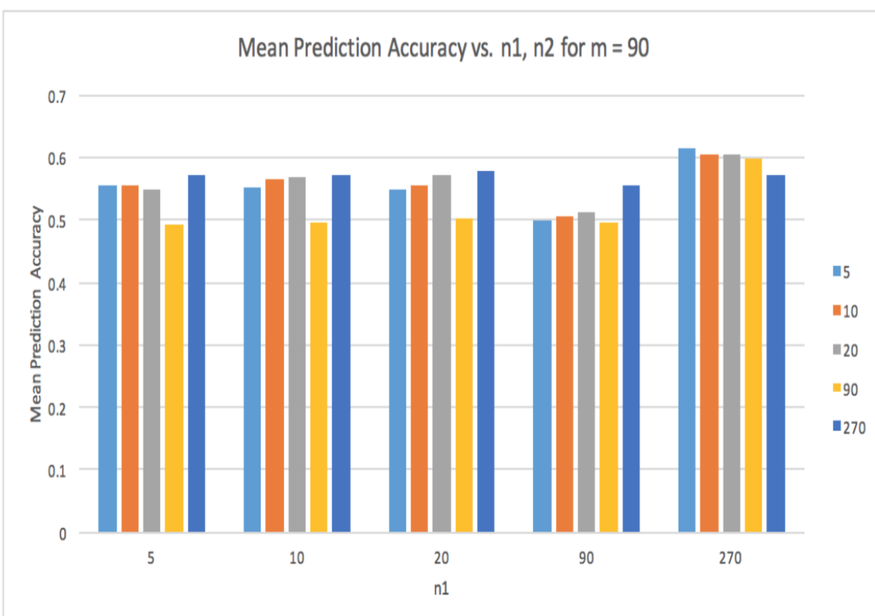


Figure 2: The mean prediction accuracy for each combination of n1, n2 when m=90

4. Experiments and result analysis

4.1 Data

Although we are trying to use the same dataset as in the paper, some of them are unavailable at present. Finally, the data here we used are 28 from the 39 stocks with the same period from 2007 to 2014, which were obtained from Yahoo Finance using R package quantmod.

4.2 Reproducing result and analysis

Here we used the same data preprocessing methods and filled the missing data using interpolated values. We also constructed 150 SVM models with radial basis kernel with default parameters (cost = 1, sigma = 0.25). More details about results are shown in Appendix 1. The result we get is similar to the result of the previous paper. As shown in Figure 3, the mean prediction accuracy here we get have the same trend as in Figure 1. When m=1, the model performs with the accuracy no better than 0.5 and then the mean accuracy increases to as high as 0.65 when m=90. Though it reached the highest point when m=90, we can still have an intuitive guess the real summit is between m=20 and m=90, which is also indicated by Figure 1.

Besides, it can also be interpreted from Appendix 1 that when m is small, the choice of n2 is more effective to accuracy than n1. Conversely, as m increases, n1 shows greater influence than n2.

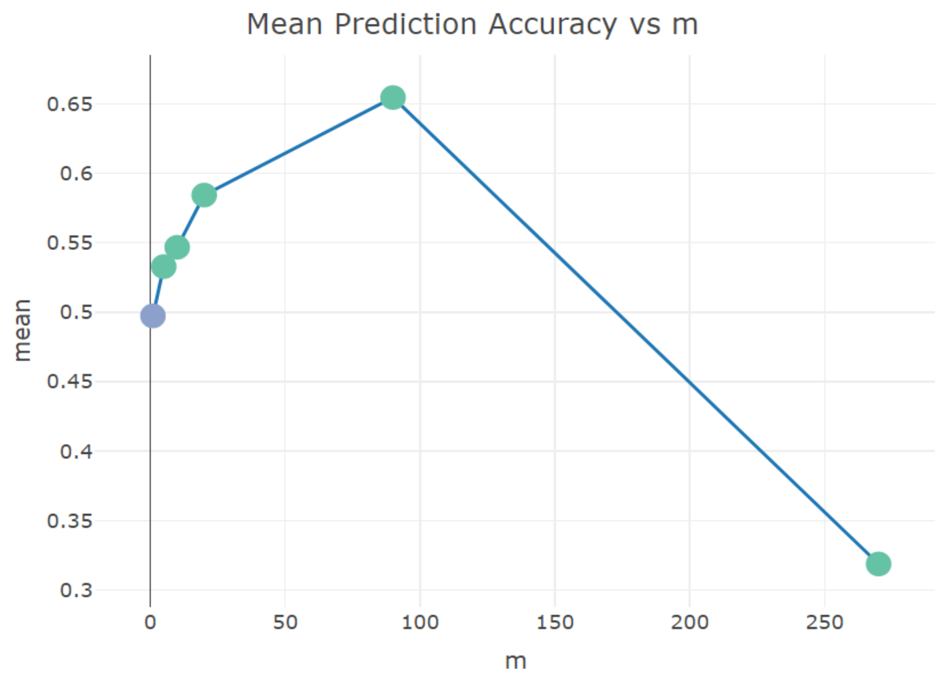


Figure 3 The mean prediction accuracy for each parameter m

We also compared the mean accuracy of different combinations of n1 and n2 when m=90, with the original paper (shown in figure 2). And our result (shown in figure 4) is also similar to the reported result in overall trend.

Along with the previous result shown in figure 3, all these reproducing results do have some difference compared with the original ones. For example, in figure 3, the highest accuracy of our result is above the highest one in the original result and the lowest is also under the lowest original result, which means the variation is relatively large; what's more, the total mean accuracy we obtained when m=90 are almost all higher than the original ones, except when n1=90. And We think the main reason leading to this discrepancy is the missing stock data, since the average accuracy was taken over the number of the stocks, and in our experiment that number is 28 whereas in the paper it is 34. Also, the missing data may compensate for the decreasing of accuracy when n1 = 90 in figure 4.

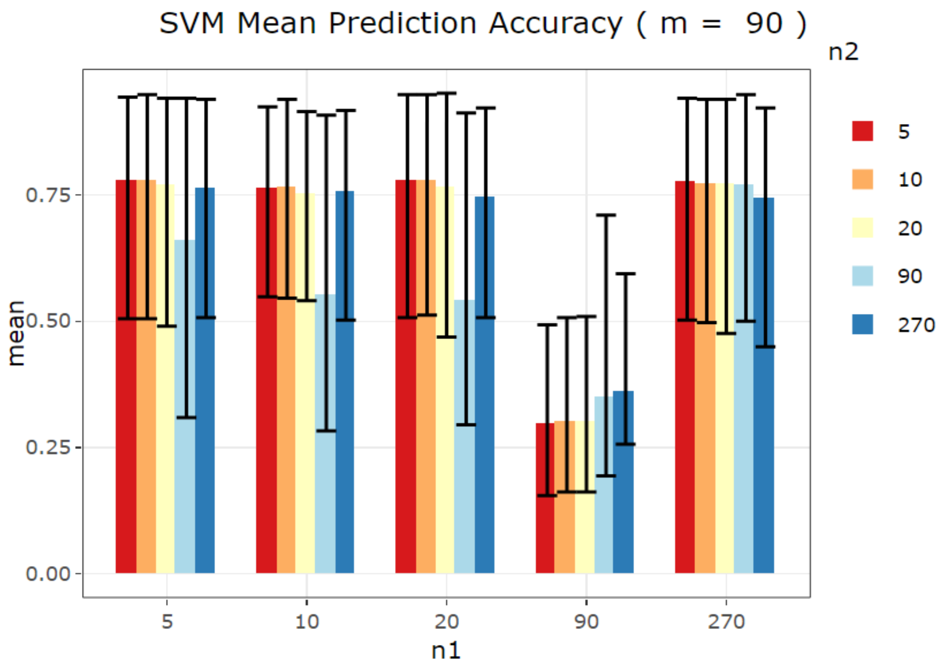


Figure 4 The mean prediction accuracy for each combination of n1, n2 when m=90

4.3 Method comparison

In our experiment, we realized another popular classification model, Random Forest, we grew 100 trees and assess the importance of predictors to predict the stock direction on a given day. To compare the prediction accuracy, we also constructed 150 models and found their mean. More details are shown in the Appendix 2.

4.4 Prediction accuracy

Random forest is a multitude of decision trees whose output is the mode of the outputs from the individual trees and the ensemble characteristic may grant it with some good performance. Here, we also constructed 150 models with different combinations of m, n1 and n2. The prediction accuracy when m=90 is shown as below, where we can see that the performance is not as good as SVM.

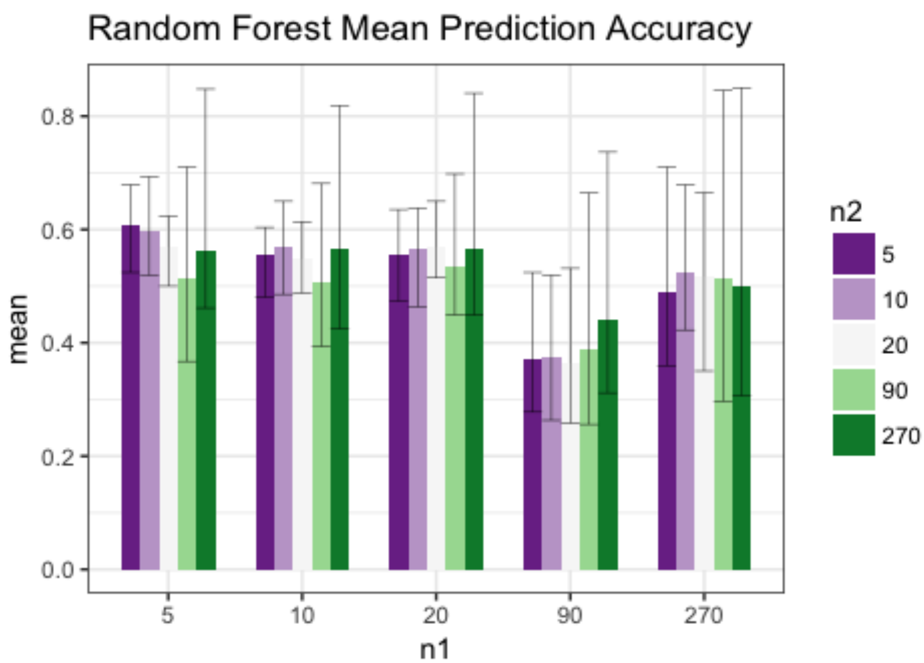


Figure 5 The mean prediction accuracy for each combination of n1, n2 when m = 90 using RF

To make better comparison between all the models we have trained, we take subtraction from the prediction accuracy of SVM by that of RF. The differences are shown in Figure 6. We can interpret from the graphs that, to predict only one day ahead, there is only small difference between these two models. However, when predicting the short term like 5 days later till 90 days later, SVM outperforms the RF model, especially when m=20, all the subtracted values are above zero regardless of n1 and n2; and when m = 90, some combinations lead to a difference in accuracy of 0.2. On the other hand, we can also notice that, to predict for the long term, RF seems to have a better performance than SVM for any of the combination of n1 and n2.

By what we have obtained from all the prediction accuracies, there appears another interesting phenomenon: when m is small, n2 has more effect on accuracy than n1. However, as m increases, n1 has greater influence than n2 on the results.

Above all, the result mainly suggests that we may use SVM to predict for the short run but use Random Forest to predict for the long-term. And for the short run, we may consider more about n2 as an important factor on accuracy than n1, and n1 as an important factor when m is large, where we need to predict for the long run.

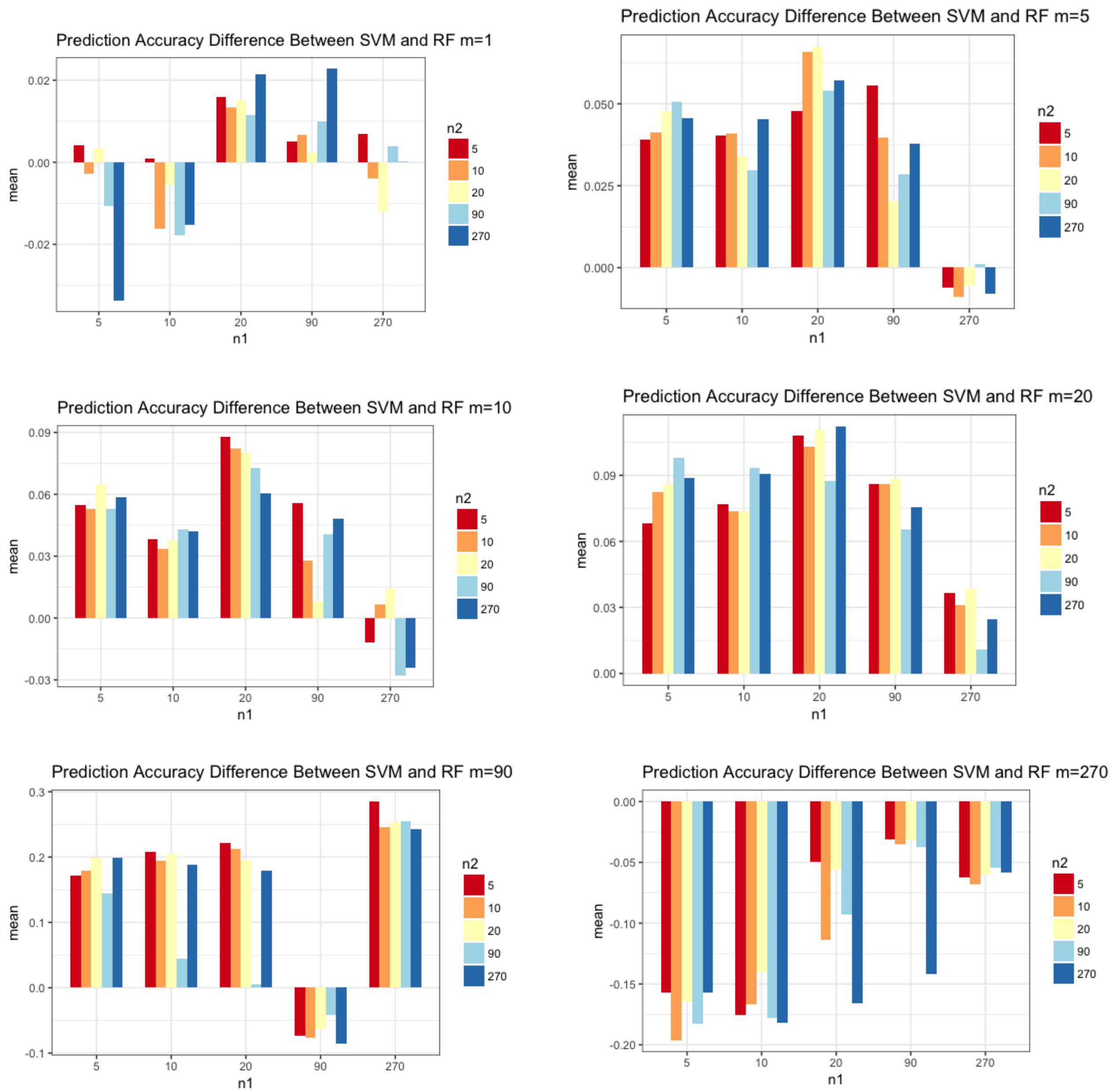


Figure 6: Comparison of prediction accuracy of SVM and Random Forest

4.5 Performance analysis

An ROC curve is the most commonly used way to visualize the performance of a binary classifier, and AUC, area under ROC curve, is a good way to summarize its performance in a single number. Here, by 10-fold cross-validation on the best model we selected (m=90, n1=20, n2=5) through all the data from 2007 to 2014 on the best model we select, we generate the ROC curve with AUC equal to 0.571, which is shown as Figure 7.

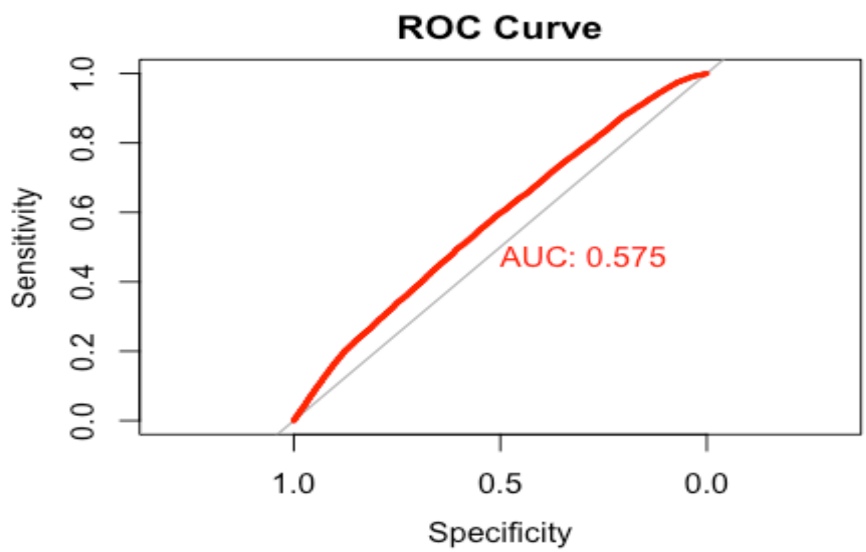


Figure 7: ROC curve and AUC of SVM model by 10-fold cross-validation

After analysis, we think the reason that the AUC is only slightly above 0.5 are as follows. First, as suggested by Efficient Market Hypothesis, which implies no algorithm can make better performance than random guessing since if someone were to gain an advantage by analyzing historical stock data, then the entire market will become aware of this advantage and as a result, the price of the share will be corrected [4]. Second, the features extracted may not be representative enough and we may need to introduce more time-series analysis to model the sophisticated time dependence of data if we want to have a higher accuracy.

5. Interactive website

To make it easy for people to understand what we did and have a good way to visualize the results, we actually developed an interactive website to demonstrate all the results by using R package, shiny. We then deploy this web app on the shinyapps.io server to make it accessible for everyone. Here is the link for the website https://rwandering.shinyapps.io/spdforecast/, however, there is a limit for the active time of this app because we are now using a free plan, so it may not be accessible all the time. The general structure of the webapp can be found in Appendix.

6. Future work

Based on what we have already done, we figured out the following things can be improved in the future.

Firstly, we can collect more data and extract more representative features. Since the accuracy is not satisfying, the most possible factor to blame is the feature. We may select some other convincing features, for example relative strength index, stochastic oscillator and on balance volume [3] to construct a more thorough and comprehensive model to predict the stock directions. Also, training different model, for example LSTM, may give a way to discover more sophisticated latent patterns. Since the prediction of stock direction can be a potential sequence prediction problem, we can't neglect the possibility to cut the day-to-day dependency off. To conquer the inadequacy of time series analysis, we can train another model which takes the correlative information into account. Moreover, the expected ultimate goal is to predict the future price of stocks more than just the direction of it. So, we are looking forward to realizing models that can give us some insights of it.

Reference

[1] Hu, Zhen, J. Zhu, and K. Tse. "Stocks market prediction using Support Vector Machine." *International Conference on Information Management, Innovation Management and Industrial Engineering* IEEE, 2014:115-118.

[2] Saahil Madge. "Predicting Stock Price Direction using Support Vector Machines" (2015).

[3] Khaidem, Luckyson, S. Saha, and S. R. Dey. "Predicting the direction of stock market prices using random forest." (2016).

[4] Malkiel, Burton G. "The Efficient Market Hypothesis and Its Critics." Journal of Economic Perspectives 17.1(2003):59-82.

Appendix I: SVM prediction accuracy with different combinations of m, n1, n2

m	n1	n2	mean	median	max	min
1	5	5	0.50120704	0.50099404	0.54075547	0.45924453
1	5	10	0.49694689	0.49801193	0.52485089	0.46918489
1	5	20	0.49588185	0.5	0.52683897	0.4612326
1	5	90	0.49517183	0.49701789	0.52485089	0.44532803
1	5	270	0.47585913	0.47514911	0.51689861	0.44930417
1	10	5	0.48849759	0.48807157	0.53081511	0.44135189
1	10	10	0.48104232	0.48210736	0.50894632	0.45328032
1	10	20	0.47948026	0.47713718	0.52286282	0.44135189
1	10	90	0.48409543	0.48310139	0.53081511	0.44333996
1	10	270	0.47834422	0.47614314	0.50298211	0.45328032
1	20	5	0.52662596	0.52683897	0.60039761	0.49701789
1	20	10	0.51597558	0.51988072	0.55666004	0.48508946
1	20	20	0.52634195	0.52584493	0.56262425	0.49304175
1	20	90	0.51313547	0.5139165	0.54075547	0.48111332
1	20	270	0.5249219	0.52087475	0.56063618	0.48707753
1	90	5	0.50205907	0.5	0.56262425	0.47514911
1	90	10	0.50461517	0.50397614	0.54075547	0.4612326
1	90	20	0.49680488	0.49900596	0.54473161	0.44930417
1	90	90	0.50582221	0.50497018	0.54274354	0.47514911
1	90	270	0.5054672	0.50099404	0.53677932	0.47117296
1	270	5	0.48721954	0.48807157	0.53479125	0.45129225
1	270	10	0.48807157	0.48906561	0.51888668	0.4612326
1	270	20	0.48771656	0.4860835	0.52286282	0.45924453
1	270	90	0.48260437	0.48210736	0.54075547	0.42743539
1	270	270	0.48956262	0.48906561	0.52485089	0.44930417
5	5	5	0.54745205	0.54709419	0.64328657	0.49298597
5	5	10	0.55489551	0.55210421	0.64729459	0.50300601
5	5	20	0.56441454	0.55911824	0.64328657	0.50701403
5	5	90	0.56054967	0.55811623	0.62324649	0.50501002
5	5	270	0.55861723	0.57014028	0.62925852	0.49699399
5	10	5	0.55181792	0.55210421	0.61923848	0.50501002
5	10	10	0.54287146	0.53006012	0.6492986	0.48496994
5	10	20	0.54101059	0.53807615	0.62124248	0.48496994
5	10	90	0.53549957	0.52705411	0.61322645	0.49098196
5	10	270	0.54058116	0.53106212	0.60521042	0.48296593
5	20	5	0.54623533	0.54408818	0.60320641	0.49699399
5	20	10	0.56348411	0.56312625	0.65531062	0.49498998
5	20	20	0.56949614	0.5741483	0.65330661	0.52104208
5	20	90	0.55782995	0.55410822	0.64128257	0.49699399
5	20	270	0.57192957	0.56412826	0.63126253	0.51903808

m	n1	n2	mean	median	max	min
5	90	5	0.54702262	0.55210421	0.60320641	0.48296593
5	90	10	0.53134841	0.53707415	0.59719439	0.4749499
5	90	20	0.51338391	0.51302605	0.58517034	0.41482966
5	90	90	0.52519324	0.51903808	0.58917836	0.46893788
5	90	270	0.50937589	0.52004008	0.60521042	0.43887776
5	270	5	0.48296593	0.47895792	0.53306613	0.43887776
5	270	10	0.47788434	0.47294589	0.54308617	0.42685371
5	270	20	0.48110507	0.47695391	0.53907816	0.42284569
5	270	90	0.4754509	0.46593186	0.54108216	0.43486974
5	270	270	0.46979674	0.46793587	0.53306613	0.41683367
10	5	5	0.57504338	0.57489879	0.6437247	0.50202429
10	5	10	0.57526027	0.57793522	0.63765182	0.5
10	5	20	0.57930885	0.57995951	0.64777328	0.50607287
10	5	90	0.56550029	0.56477733	0.6417004	0.48582996
10	5	270	0.56991035	0.57388664	0.62145749	0.50607287
10	10	5	0.57576634	0.57287449	0.63562753	0.50607287
10	10	10	0.57316368	0.57489879	0.65384615	0.48380567
10	10	20	0.57482649	0.57894737	0.6417004	0.46356275
10	10	90	0.56701851	0.56680162	0.65182186	0.48178138
10	10	270	0.5689705	0.55870445	0.63562753	0.5
10	20	5	0.5858878	0.58603239	0.6417004	0.53238866
10	20	10	0.58465876	0.5840081	0.65991903	0.50809717
10	20	20	0.5877675	0.58299595	0.6659919	0.52226721
10	20	90	0.58198381	0.5840081	0.65182186	0.51214575
10	20	270	0.58111625	0.57591093	0.64574899	0.50404858
10	90	5	0.56282533	0.55668016	0.65182186	0.48582996
10	90	10	0.53224407	0.52024291	0.61133603	0.46153846
10	90	20	0.50759109	0.51012146	0.59919028	0.42712551
10	90	90	0.54814922	0.54048583	0.63967611	0.46153846
10	90	270	0.51323019	0.52732794	0.62955466	0.37246964
10	270	5	0.47231058	0.4645749	0.56477733	0.41497976
10	270	10	0.47303355	0.46356275	0.56477733	0.4048583
10	270	20	0.47614228	0.47368421	0.5708502	0.41902834
10	270	90	0.47028629	0.46963563	0.548583	0.40688259
10	270	270	0.46659919	0.46153846	0.56680162	0.3805668
20	5	5	0.62049882	0.60847107	0.72520661	0.53305785
20	5	10	0.62204841	0.61570248	0.73553719	0.53719008
20	5	20	0.62116293	0.61260331	0.72520661	0.52066116
20	5	90	0.61430047	0.61673554	0.73140496	0.51446281
20	5	270	0.61031582	0.60330579	0.70867769	0.53512397
20	10	5	0.62455726	0.61053719	0.72520661	0.5392562
20	10	10	0.62175325	0.60847107	0.70867769	0.55165289

m	n1	n2	mean	median	max	min
20	10	20	0.62145809	0.60123967	0.72107438	0.53305785
20	10	90	0.61260331	0.6053719	0.69214876	0.54752066
20	10	270	0.61297226	0.60640496	0.71900826	0.5392562
20	20	5	0.63053424	0.61466942	0.73760331	0.5392562
20	20	10	0.63141972	0.6177686	0.71487603	0.54958678
20	20	20	0.6359209	0.625	0.74380165	0.55578512
20	20	90	0.61813754	0.63946281	0.70454545	0.47520661
20	20	270	0.61747344	0.61673554	0.74793388	0.52066116
20	90	5	0.61297226	0.61570248	0.68595041	0.51239669
20	90	10	0.59526269	0.59607438	0.6714876	0.48347107
20	90	20	0.61739965	0.63016529	0.69834711	0.5268595
20	90	90	0.5893595	0.57541322	0.73140496	0.46487603
20	90	270	0.53652597	0.54338843	0.66528926	0.39256198
20	270	5	0.4724026	0.46280992	0.59297521	0.37396694
20	270	10	0.47188607	0.46487603	0.58677686	0.37809917
20	270	20	0.4788961	0.46900826	0.59917355	0.38842975
20	270	90	0.45476682	0.45557851	0.52272727	0.37396694
20	270	270	0.46229339	0.45454545	0.59090909	0.36363636
90	5	5	0.77708765	0.79710145	0.94444444	0.50483092
90	5	10	0.77726018	0.80072464	0.94927536	0.50483092
90	5	20	0.76846101	0.78623188	0.94202899	0.49033816
90	5	90	0.65838509	0.63405797	0.94202899	0.30917874
90	5	270	0.76138716	0.76086957	0.93961353	0.50724638
90	10	5	0.76293996	0.78502415	0.92512077	0.54830918
90	10	10	0.76328502	0.78502415	0.93961353	0.54589372
90	10	20	0.75276052	0.7826087	0.91545894	0.5410628
90	10	90	0.55098344	0.52294686	0.90821256	0.2826087
90	10	270	0.75491718	0.74758454	0.9178744	0.50241546
90	20	5	0.77786404	0.80193237	0.94927536	0.50724638
90	20	10	0.77717391	0.80193237	0.94927536	0.51207729
90	20	20	0.76466529	0.78502415	0.95169082	0.46859903
90	20	90	0.5405452	0.52294686	0.91304348	0.29468599
90	20	270	0.74611801	0.73913043	0.92270531	0.50724638
90	90	5	0.29623879	0.25241546	0.49275362	0.15458937
90	90	10	0.29813665	0.24516908	0.50724638	0.16183575
90	90	20	0.29960317	0.26328502	0.50966184	0.16183575
90	90	90	0.34782609	0.31763285	0.71014493	0.19323671
90	90	270	0.35714286	0.33816425	0.5942029	0.25603865
90	270	5	0.77380952	0.79830918	0.94202899	0.50241546
90	270	10	0.77139406	0.79468599	0.93961353	0.49758454
90	270	20	0.77199793	0.78140097	0.93961353	0.47584541
90	270	90	0.769755	0.7910628	0.94927536	0.5

m	n1	n2	mean	median	max	min
90	270	270	0.74258109	0.74637681	0.92270531	0.44927536
270	5	5	0.28418803	0.10470085	0.96581197	0.03846154
270	5	10	0.26465201	0.09401709	0.95726496	0.02136752
270	5	20	0.28434066	0.11111111	0.94871795	0.04273504
270	5	90	0.28403541	0.11752137	0.88034188	0
270	5	270	0.23275336	0	0.98717949	0
270	10	5	0.31364469	0.16880342	0.85470085	0.12393162
270	10	10	0.32051282	0.18803419	0.9017094	0.0982906
270	10	20	0.3519536	0.2542735	0.85042735	0.12820513
270	10	90	0.31547619	0.22222222	0.81623932	0.00854701
270	10	270	0.23275336	0	0.98717949	0
270	20	5	0.42078755	0.38034188	0.64957265	0.27350427
270	20	10	0.35149573	0.27777778	0.84615385	0.14529915
270	20	20	0.40873016	0.37820513	0.76068376	0.18803419
270	20	90	0.36782662	0.32478632	0.83333333	0.02991453
270	20	270	0.23153236	0	0.98717949	0
270	90	5	0.45344933	0.44871795	0.64102564	0.29487179
270	90	10	0.45558608	0.44871795	0.61538462	0.28205128
270	90	20	0.45970696	0.46153846	0.64957265	0.28205128
270	90	90	0.46199634	0.4508547	0.7008547	0.28632479
270	90	270	0.23946886	0.00641026	0.98717949	0
270	270	5	0.22710623	0	0.98717949	0
270	270	10	0.22863248	0	0.98717949	0
270	270	20	0.23046398	0	0.98717949	0
270	270	90	0.25320513	0.00213675	0.98717949	0
270	270	270	0.29532967	0.14529915	0.98717949	0

Appendix II: Random Forest prediction accuracy with different combinations of m, n1, n2

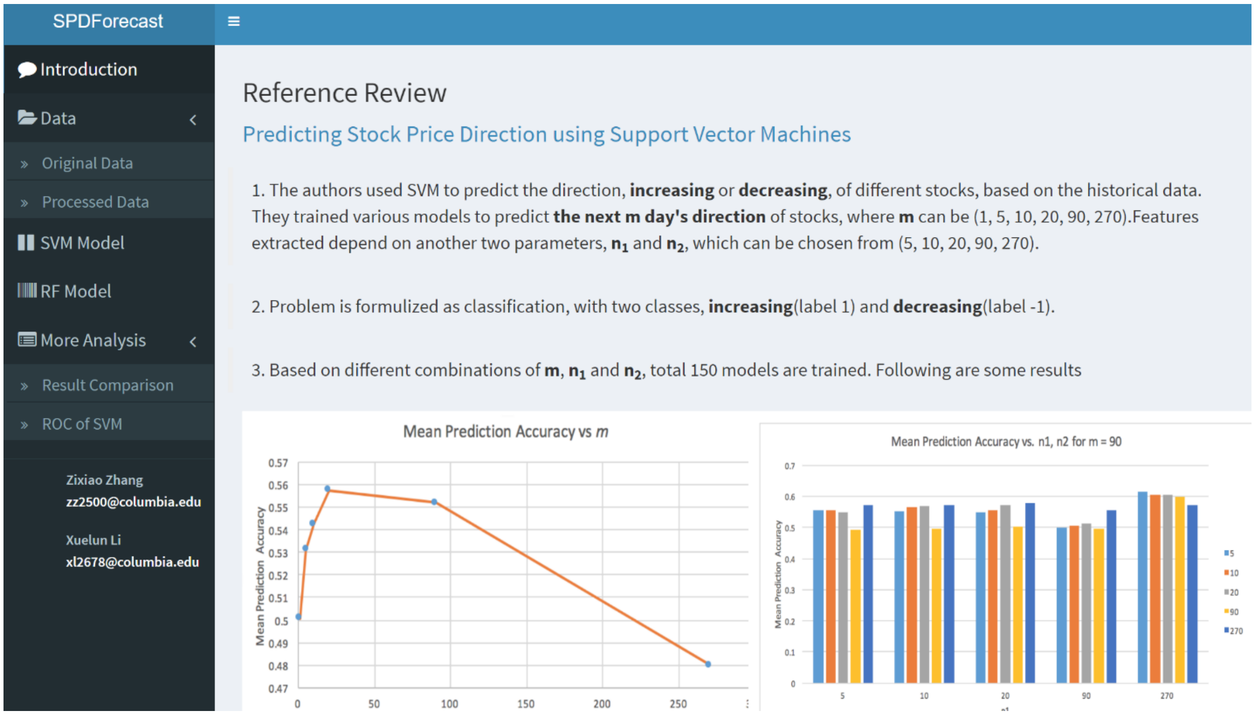
m	n1	n2	mean	median	max	min
1	5	5	0.49701789	0.50099404	0.53081511	0.45526839
1	5	10	0.49964499	0.49701789	0.55666004	0.4473161
1	5	20	0.49233172	0.49304175	0.51888668	0.45526839
1	5	90	0.50575121	0.50198807	0.54473161	0.46719682
1	5	270	0.50951434	0.50994036	0.53479125	0.43737575
1	10	5	0.48750355	0.48906561	0.52087475	0.44930418
1	10	10	0.4973019	0.49403579	0.55467197	0.46918489
1	10	20	0.48501846	0.48508946	0.53479125	0.44333996
1	10	90	0.50198807	0.50198807	0.5387674	0.45328032
1	10	270	0.49360977	0.49304175	0.52087475	0.4473161
1	20	5	0.51079239	0.5139165	0.55666004	0.47912525
1	20	10	0.50255609	0.50695825	0.56063618	0.45328032
1	20	20	0.5110054	0.51292247	0.55864811	0.4612326
1	20	90	0.50170406	0.50099404	0.56262425	0.46918489
1	20	270	0.50340812	0.49701789	0.5387674	0.45924453
1	90	5	0.4970889	0.49005964	0.54075547	0.46322068
1	90	10	0.49786992	0.5	0.53479125	0.45725646
1	90	20	0.4944618	0.49403579	0.53081511	0.42942346
1	90	90	0.49581085	0.49602386	0.53677932	0.4612326
1	90	270	0.48267538	0.48011928	0.52683897	0.44333996
1	270	5	0.48019029	0.47912525	0.51491054	0.44135189
1	270	10	0.49211872	0.48906561	0.54075547	0.45129225
1	270	20	0.49971599	0.49900596	0.54274354	0.44930418
1	270	90	0.47869923	0.47415507	0.53479125	0.4473161
1	270	270	0.48949162	0.48409543	0.54274354	0.45725646
5	5	5	0.50837389	0.50801603	0.55310621	0.46092184
5	5	10	0.5136702	0.51002004	0.5991984	0.46893788
5	5	20	0.51667621	0.51302605	0.55911824	0.47695391
5	5	90	0.5098769	0.51402806	0.5511022	0.46492986
5	5	270	0.51302605	0.51302605	0.59318637	0.45490982
5	10	5	0.51152305	0.51202405	0.5751503	0.45290581
5	10	10	0.50178929	0.50701403	0.56312625	0.4488978
5	10	20	0.50687089	0.50601202	0.56112224	0.46292585
5	10	90	0.50586888	0.50601202	0.55711423	0.46693387
5	10	270	0.49534784	0.49398798	0.53907816	0.44488978
5	20	5	0.49842542	0.501002	0.53907816	0.4509018
5	20	10	0.49756656	0.498998	0.53106212	0.46693387
5	20	20	0.50214715	0.50400802	0.54108216	0.43086172
5	20	90	0.5037933	0.51002004	0.55511022	0.43687375
5	20	270	0.51481535	0.51703407	0.58517034	0.43086172

m	n1	n2	mean	median	max	min
5	90	5	0.49155454	0.49398798	0.53106212	0.44689379
5	90	10	0.49155454	0.49098196	0.54108216	0.44689379
5	90	20	0.49305754	0.49498998	0.53907816	0.45290581
5	90	90	0.49663613	0.50400802	0.53306613	0.41282565
5	90	270	0.47144289	0.4739479	0.5250501	0.41683367
5	270	5	0.48897796	0.48897796	0.55310621	0.41683367
5	270	10	0.48668766	0.48496994	0.53306613	0.43086172
5	270	20	0.48647295	0.48196393	0.56713427	0.42685371
5	270	90	0.47423418	0.47795591	0.5250501	0.39478958
5	270	270	0.47788434	0.47695391	0.53707415	0.4248497
10	5	5	0.52038751	0.51619433	0.55465587	0.47975709
10	5	10	0.52212261	0.51720648	0.58906883	0.48380567
10	5	20	0.51438693	0.51518219	0.59919028	0.46558705
10	5	90	0.51279641	0.51417004	0.55060729	0.44736842
10	5	270	0.51149508	0.5131579	0.56477733	0.46558705
10	10	5	0.53752169	0.53238866	0.60931174	0.49797571
10	10	10	0.53961828	0.54149798	0.59109312	0.49190283
10	10	20	0.53701562	0.548583	0.5708502	0.46153846
10	10	90	0.52385772	0.52327935	0.57287449	0.46356275
10	10	270	0.52696646	0.52935223	0.57692308	0.47975709
10	20	5	0.4981203	0.50404858	0.53643725	0.451417
10	20	10	0.50224118	0.5	0.56275304	0.43117409
10	20	20	0.50759109	0.50809717	0.56882591	0.4534413
10	20	90	0.5093262	0.51214575	0.57287449	0.41295547
10	20	270	0.52074899	0.52226721	0.58906883	0.45951417
10	90	5	0.50722961	0.50809717	0.5708502	0.43522267
10	90	10	0.50426547	0.51012146	0.58097166	0.42712551
10	90	20	0.49949393	0.5	0.56072875	0.44736842
10	90	90	0.50730191	0.50809717	0.5465587	0.4534413
10	90	270	0.46515327	0.47165992	0.53036437	0.40890688
10	270	5	0.48402256	0.49089069	0.5465587	0.41902834
10	270	10	0.46652689	0.46153846	0.52631579	0.36842105
10	270	20	0.46182765	0.4645749	0.50607287	0.37651822
10	270	90	0.49804801	0.50101215	0.5951417	0.42307692
10	270	270	0.4906738	0.48481781	0.61133603	0.4291498
20	5	5	0.55239079	0.55268595	0.63223141	0.49586777
20	5	10	0.53962515	0.54545455	0.60330579	0.48347107
20	5	20	0.53512397	0.53719008	0.58471074	0.47520661
20	5	90	0.51645514	0.51859504	0.56198347	0.45867769
20	5	270	0.52132527	0.52066116	0.63429752	0.41528926
20	10	5	0.54752066	0.55371901	0.59917355	0.49173554
20	10	10	0.54818477	0.54338843	0.6177686	0.49380165

m	n1	n2	mean	median	max	min
20	10	20	0.54766824	0.54235537	0.60123967	0.49793388
20	10	90	0.51925915	0.51859504	0.56818182	0.44834711
20	10	270	0.52213695	0.52272727	0.58884298	0.46280992
20	20	5	0.5225059	0.52479339	0.59917355	0.44214876
20	20	10	0.52855667	0.52995868	0.58677686	0.48347107
20	20	20	0.52486718	0.52995868	0.57438017	0.45454546
20	20	90	0.53091795	0.52789256	0.59297521	0.47520661
20	20	270	0.50538666	0.51859504	0.61157025	0.41528926
20	90	5	0.52671192	0.53099174	0.58264463	0.45867769
20	90	10	0.50900236	0.50723141	0.58677686	0.44421488
20	90	20	0.52914699	0.52479339	0.6177686	0.46487603
20	90	90	0.52405549	0.52479339	0.59710744	0.43595041
20	90	270	0.46118654	0.45144628	0.58057851	0.3553719
20	270	5	0.43572904	0.43698347	0.52892562	0.34917355
20	270	10	0.44104191	0.43801653	0.5392562	0.32231405
20	270	20	0.44067296	0.44318182	0.54338843	0.37603306
20	270	90	0.44399351	0.43698347	0.55785124	0.36157025
20	270	270	0.43786895	0.42871901	0.6053719	0.33057851
90	5	5	0.60593513	0.61594203	0.67874396	0.52415459
90	5	10	0.59773982	0.60507246	0.69323672	0.51932367
90	5	20	0.56884058	0.5736715	0.62318841	0.5
90	5	90	0.51466529	0.53140097	0.71014493	0.36714976
90	5	270	0.56254313	0.55434783	0.84782609	0.46135266
90	10	5	0.55443409	0.56400966	0.60386473	0.48067633
90	10	10	0.56858178	0.57608696	0.64975845	0.48550725
90	10	20	0.54848171	0.5531401	0.61352657	0.48792271
90	10	90	0.50577985	0.49396135	0.68115942	0.39371981
90	10	270	0.56625259	0.56038647	0.81884058	0.42512077
90	20	5	0.55641822	0.56280193	0.6352657	0.47342995
90	20	10	0.56444099	0.56884058	0.63768116	0.46376812
90	20	20	0.56961698	0.56400966	0.64975845	0.51449275
90	20	90	0.53476536	0.53381643	0.69806763	0.44927536
90	20	270	0.56625259	0.5615942	0.84057971	0.44927536
90	90	5	0.36982402	0.35024155	0.52415459	0.27777778
90	90	10	0.3744824	0.36231884	0.51932367	0.26328502
90	90	20	0.36300897	0.35144928	0.53140097	0.25845411
90	90	90	0.38992409	0.3647343	0.66425121	0.25603865
90	90	270	0.44228779	0.4384058	0.73671498	0.3115942
90	270	5	0.48904417	0.50362319	0.71014493	0.35990338
90	270	10	0.52562112	0.53019324	0.67874396	0.42270531
90	270	20	0.51863354	0.5205314	0.66425121	0.35024155
90	270	90	0.51466529	0.50241546	0.84541063	0.29710145

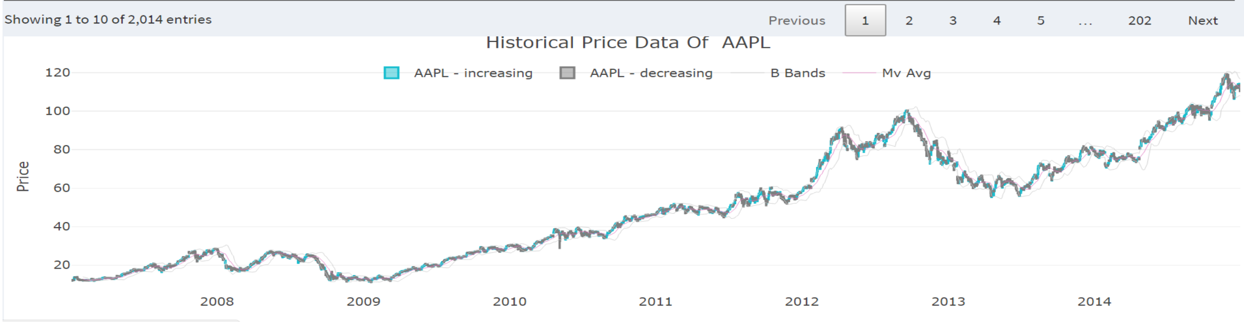
m	n1	n2	mean	median	max	min
90	270	270	0.49939614	0.48913044	0.85024155	0.30676329
270	5	5	0.44093407	0.40598291	0.64957265	0.30769231
270	5	10	0.46077534	0.43589744	0.62393162	0.35042735
270	5	20	0.44856532	0.42521368	0.65811966	0.31196581
270	5	90	0.46688034	0.45512821	0.66239316	0.32905983
270	5	270	0.39010989	0.36752137	0.85470086	0.10683761
270	10	5	0.48946886	0.48717949	0.56410256	0.44017094
270	10	10	0.48748474	0.48290598	0.57264957	0.41452992
270	10	20	0.49221612	0.48931624	0.57264957	0.41452992
270	10	90	0.49328449	0.48290598	0.58974359	0.35470086
270	10	270	0.41437729	0.36111111	0.87606838	0.15811966
270	20	5	0.4702381	0.46153846	0.58119658	0.38888889
270	20	10	0.46489622	0.44017094	0.60683761	0.37606838
270	20	20	0.46504884	0.46153846	0.61965812	0.34188034
270	20	90	0.46092796	0.45940171	0.6025641	0.31196581
270	20	270	0.39713065	0.33974359	0.86324786	0.10683761
270	90	5	0.48473749	0.48717949	0.55555556	0.36752137
270	90	10	0.49099512	0.5042735	0.56410256	0.33333333
270	90	20	0.49175824	0.51282051	0.58974359	0.34615385
270	90	90	0.49923687	0.5042735	0.65384615	0.32051282
270	90	270	0.38110501	0.3034188	0.84615385	0.14102564
270	270	5	0.28983517	0.11965812	0.90598291	0.07264957
270	270	10	0.29624542	0.14102564	0.89316239	0.07264957
270	270	20	0.29075092	0.14102564	0.9017094	0.05982906
270	270	90	0.30753968	0.17307692	0.91025641	0.04273504
270	270	270	0.35363248	0.27777778	0.96581197	0.02136752

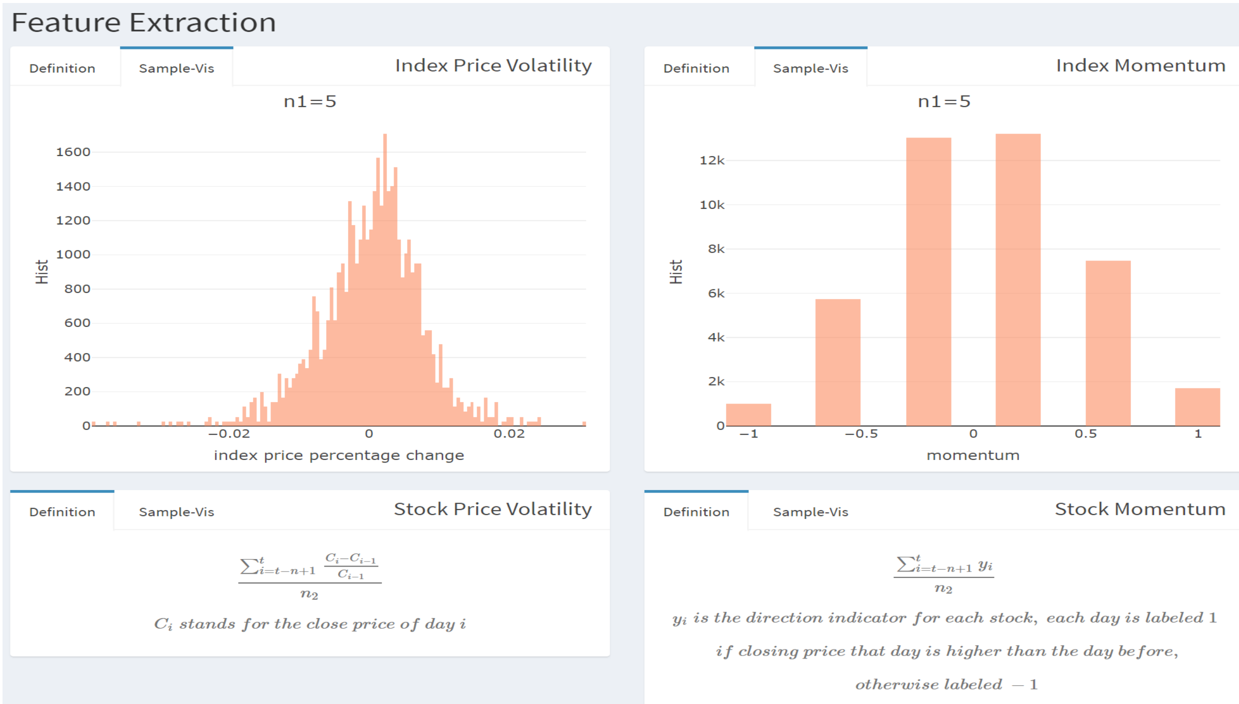
Appendix III:



	Date	Open	High	Low	Close	Volume	Adjusted
1	2007-01-03	12.327143	12.368571	11.7	11.971429	309579900	8.104136
2	2007-01-04	12.007143	12.278571	11.974286	12.237143	211815100	8.284014
3	2007-01-05	12.252857	12.314285	12.057143	12.15	208685400	8.225021
4	2007-01-08	12.28	12.361428	12.182858	12.21	199276700	8.265638
5	2007-01-09	12.35	13.282857	12.164286	13.224286	837324600	8.952269
6	2007-01-10	13.535714	13.971429	13.35	13.857142	738220000	9.380682
7	2007-01-11	13.705714	13.825714	13.585714	13.685715	360063200	9.264633
8	2007-01-12	13.512857	13.58	13.318571	13.517143	328172600	9.150517
9	2007-01-16	13.668571	13.892858	13.635715	13.871428	311019100	9.390354
10	2007-01-17	13.937143	13.942857	13.545714	13.564285	411565000	9.182431

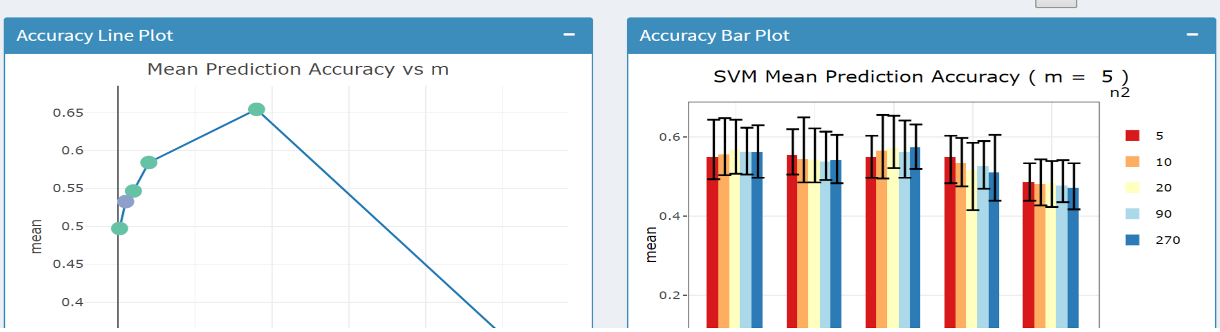
Showing 1 to 10 of 2,014 entries Previous 1 2 3 4 5 … 202 Next





	m	n1	n2	mean	median	max	min
26	5	5	5	0.547452046951045	0.547094188376753	0.643286573146293	0.492985971943888
27	5	5	10	0.554895505296307	0.552104208416834	0.647294589178357	0.503006012024048
28	5	5	20	0.564414543372459	0.559118236472946	0.643286573146293	0.507014028056112
29	5	5	90	0.560549670770112	0.55811623246493	0.623246492985972	0.50501002004008
30	5	5	270	0.558617234468938	0.570140280561122	0.629258517034068	0.496993987975952
31	5	10	5	0.551817921557401	0.552104208416834	0.619238476953908	0.50501002004008
32	5	10	10	0.542871457200115	0.530060120240481	0.649298597194389	0.48496993987976
33	5	10	20	0.541010592613799	0.538076152304609	0.62124248496994	0.48496993987976
34	5	10	90	0.535499570569711	0.527054108216433	0.613226452905812	0.490981963927856
35	5	10	270	0.540581162324649	0.531062124248497	0.605210420841683	0.482965931863727

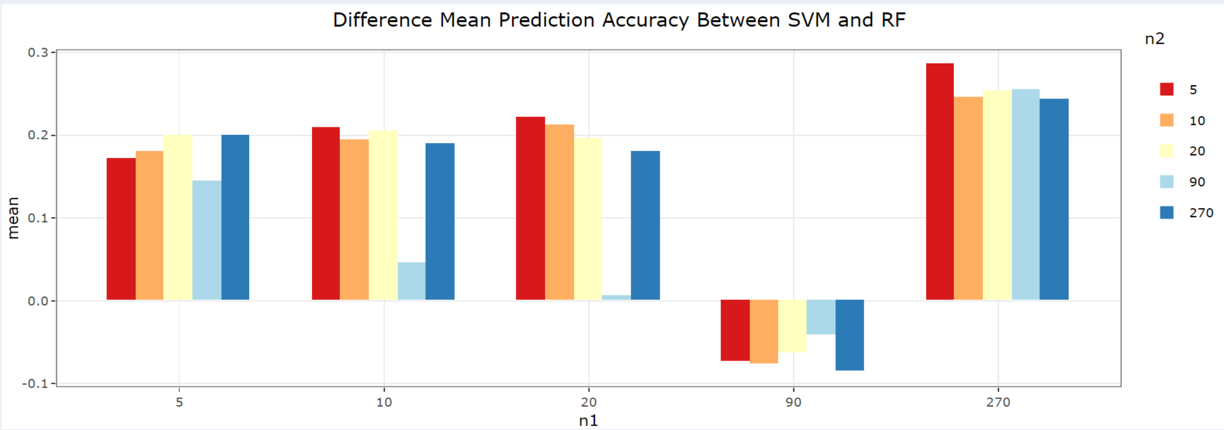
Showing 1 to 10 of 25 entries Previous 1 2 3 Next



Results Comparison

Select m

90



We compare the test accuracy of different models, and the result shows in most cases, SVM outperform RF. But when **m** keeps increasing, the RF model can do better in some $\mathbf{n_1}$ and $\mathbf{n_2}$ combination. When **m** increases to 270, though the accuracy of RF exceeds SVM, their absolute accuracies are both not satisfying.