Statistical Methods to Predict Industrial Stock Prices

Columbia University

Course: GR5261, 2017 Spring

Instructor: Prof. Zhiliang Ying

Team members:

Jiahao Zhang(Project Design, LDA Coding)
Pinren Chen(Data Cleaning and Neural Network Coding)
Hongjie Ren(Data Collection, Logistic Coding, Presentation)
Peilin Qiu(Random Forest Coding)
Sihuai Yu(Data Collection and Presentation)
Tangming Li(Random Trees Coding)
Yufei Zhao(Naive Bayes Coding)
Yangyu Niu(SVM Coding)

Date: May. 7th, 2017

Introduction:

Qualitative and quantitative analysis have played a vital role in assisting investors in the stock market. Traditionally, conventional fundamental analysis has been widely used to predict fluctuations in the stock market. However, this analysis is unable to capture short term fluctuations, and therefore fails offer timely suggestions on trading strategies. The trend has shifted to quantitative analysis where most models account for a small number of selected stocks in the market but fail to address the intrinsic difference among industries. In addressing these inadequacies, our group focus on individual industries. We categorize more than 2000 stocks based on the industries and explore different statistical models to find the optimized modeling strategy for stocks in each industry.

Objective and Plan:

We compare and summarize the prediction performance of seven statistical models on more than 2000 companies from six major industries. The chosen models are Random Trees, Random Forest, Naïve Bayes, Logistic Regression, Support Vector Machine, Neural Network, Linear Discriminant Analysis.

The data used in this project are acquired from Bloomberg Terminals. Initially, 47 indexes are included. These indexes are divided into two categories: micro indexes and macro indexes. To ensure the data quality and model performance, we only include those that are available during the time span we choose, i.e. from 2009.3.31 to 2017.3.1. For the micro indexes, we include Current Market Capitalization, Book Value Per Share, Asset Turnover, Net Revenue Growth, PE Ratio, Quick Ratio, etc. These indexes comprehensively describe the financial picture of one company. As for the macro indexes, we choose M1, M2, GDP, CPI, Total Amount of Import and Export and some other indexes. In this way, both the financial condition of a company and the economic condition of a nation are all being considered, which we believe will make our project more convincing.

After obtaining the data, we train the seven models on the training set, which is the historical data under 47 selected features from 2009.3.31 to 2016.12.31. Next, we test the fitted models on the 3-month returns ending on 2017.3.31 and finally, compute accuracy by comparing model performance from our estimates and the true values obtained from Bloomberg Terminals.

To make the comparison more convenient, we recode our response variable (the quarterly net return) in the following scheme: If the response variable is greater than or equal to 10 percent, we recode it as ‘ Good’, otherwise ‘ Bad ’. If both predicted value and the true value are recorded as “Good” or “Bad” simultaneously, We count this observation as a match and denote it as “1”, otherwise this observation is “0”. Accuracy is the total number of 1’s divided by the total number of all records in the corresponding dataset.

$$Accuracy = \frac{Total\ Number\ of\ 1's}{(Total\ Number\ of\ '1's + Total\ Number\ of\ '0's)}$$

Before we move on with methodologies for modeling, we want to talk about data cleaning. Due to the high prevalence of NA in the original dataset, we first remove columns with more than 80% missing values. Then a simple way of data imputations is used: We replace the missing value with the mean value of each column.

Methodology:

Among all seven classification models, we first explore three linear classifiers: Logistic Regression, LDA and SVM.

Based on one or more independent variables, Logistic model yields dichotomous outcomes. Typically, the outcomes are "0" and "1", which translate into "bad" and "good" in our case. Logistic model is easy to implement because there is no assumption or restriction on the distribution of the data. The model is also able to analyze both categorical and continuous variables. The model belongs to generalized linear models and tends to confront multicollinearity problems. Therefore, we employ R functions that will give warnings when multicollinearity is detected.

Inspired by Logistic model, we implement another linear classifier: the Linear Discriminant Analysis (LDA) model. LDA is a generalization of Fisher's Linear Discriminant, a method used in Statistics, Pattern Recognition and Machine Learning. While both Logistic Regression and LDA are widely used for developing classification models, they are different in that LDA assumes the explanatory variables to be normally distributed and an equal class covariance. Since each model is constructed for an entire industry where stocks share some intrinsic similarities, it is reasonable to assume equal class covariance between "Good" and "Bad" classes. Another motivation for using LDA lies in its stable performance with relatively small sample size and a large number of explanatory variables.

Our third model is Support Vector Machine (SVM). SVM algorithm outputs an optimal hyperplane, which categorizes new examples, based on training data. SVM has two important parameters: the cost parameter "c" and the "kernel". The cost parameter determines the width of the margin. A large "C" signifies a harsh penalty on misclassification and hence leads to a narrow margin. SVM hyperplane assigns more weights on points near the decision boundary, and these points, consequently, have a much larger influence on how the hyperplane is placed. Another parameter is "kernel". The "kernel" gives us options to kernelize our model. When the number of features is more than one and the relation among them is not linear, kernel functions are used. Nevertheless, we use the default "linear" function for the simplicity of our model as well as for computational efficiency.

Next we expand our list of models with some tree based methods. We first try out Decision Trees model. All input features of Decision Trees have finite discrete domains, and there is a single target feature called the classification. A decision tree includes one root node, multiple internal nodes labeled with input features, and multiple leaf nodes. Each leaf corresponds to a decision, which is labeled with a class or a probability distribution over the classes. In regards to our project, the leaf nodes indicate our judgement of the change of stock prices, values of "good" or "bad". A tree is grown by splitting the source set into subsets based on attribute values. The process is complete when the subset at every node is below a threshold, or when splitting no longer adds value to the predictions. We calculate the information gain using all attributes, divide the root nodes with the optimal attribute, and then divide each branch node with the optimal attribute from the remaining until we run out of the attributes. The tree gets overwhelmingly complicated when it reaches a depth of 33. This requires us to implement pruning in order to avoid overfitting while pruning can also bring about the increase of bias.

We also use another tree based algorithm called Random Forest. Random Forest is an ensemble learning method for classification, regression and other tasks. Random Forest makes a prediction by averaging the outcome of many decision trees into a single vote. Compared with

the Decision Trees, Random Forest has a lower overfitting risk and generally makes a better prediction. While utilizing Random Forests, we restrict the number of variables to consider in each split. This produces a greater diversity of trees or weak learners and reduces the variance of the average prediction. Random Forests, however, weakens the interpretability of the model and requires intensive computations. Here we will briefly explain the algorithm of the Random Forests: First, we fit decision trees to different Bootstrap samples. Then we select a random sample of m<p predictors to consider in each step and grow k trees in total (in our case, k = 500. Finally, average the prediction of each tree.

Naive Bayes is a straightforward tool for constructing classifiers. In addition, Naive Bayes classifier fully utilizes Bayes' theorem to make an accurate prediction with strong independence assumptions among the features. The advantage of this method is its ability to compute with efficiency. Nevertheless, we cannot overlook its weakness: Responses have to be categorical, and features need to be strongly independent. According to Bayes' theorem, we can get scaled posterior distribution by multiplying likelihood with prior distribution. The probabilistic model enables us to make the final decision about the class of the observation. If the assumed response is in a certain category and the posterior distribution is maximized, this category is the predicted class of this observation under Naive Bayes method.

Finally, we try the Neural Network model. The central idea in Neural Network is to extract linear combinations of the inputs(our indicators) as derived features, and then model the target(stock return) as a nonlinear function of these features. The nonlinear function offered by Neural Network is highly flexible and involves activation function(threshold) and back-propagation equations. For our model, two hidden layers are used. Three neurons are included in the first layer and two in the second. One advantage of this model is that it can detect all possible interactions between predictor variables, including complex linear and nonlinear relationships. The model also has a major disadvantage: it could be computationally quite expensive. But it is still affordable for our model since there are about 35 columns and 5000 rows for each industry.

To fit these abovementioned models, we include all 35 explanatory variables using the "train" function in R. When applicable, a five fold cross validation is performed on the training set to tune parameters, and the optimal model obtained is then tested on the test set. Prediction accuracy is computed for each of the industry, and the adjusted count is calculated to ensure the goodness of fit. No warnings are given during the model fitting process, and therefore the possibility of multicollinearity can be excluded.

Result:

We summarize the prediction accuracy of all the models into the following table:

	Banks	Capital Goods	Pharma, Biotech & Life Sciences	Software & Services	Health Care Equipment & Services	Real Estate
Decision Tree	0.078	0.772	0.581	0.676	0.387	0.879
Random Forest	0.102	0.747	0.533	0.649	0.622	0.888
SVM	0.080	0.260	0.529	0.357	0.389	0.125
Logistic	0.869	0.778	0.577	0.682	0.605	0.869
Naïve Bayes	0.243	0.735	0.445	0.662	0.412	0.271
Neural Network	0.239	0.735	0.581	0.662	0.613	0.888
Linear Discriminant Analysis	0.771	0.673	0.618	0.635	0.653	0.708

Form 4-1 Prediction Accuracy of Seven Models

From the table, we draw the following conclusions:

The Decision Trees model has satisfactory results in Real estate and Capital goods industry. The reason why these two industries outperform others is because they share common features: both of them are capital intensive and have low turnover rate, which demands a longer period to get the cash back. By capturing these key factors, the model can finally get fairly accurate results. Besides, the model also performs well in Software & Services(67.6%) and Pharma, Biotech & Life Science industry(58.1%). Among all the industries, Decision Trees performs worst in Banks, which may result from the high correlations and linkages of the indicators. We need to either improve the model or adopt another method to solve this problem.

The random forest model performs slightly better than decision tree in predicting Real estate and Banks, increasing the accuracy by 0.9% and 2.4%, respectively. One notable change is the accuracy of Health Care Equipment & Services in random forest is much better than Decision Trees (increase from 38.7% to 62.2%). Other three industries, however, generate slight decreases in accuracy. Overall, Banks industry still has the lowest prediction rate, and Real Estate has the highest prediction rate. Although other industries have a moderate good prediction, we still need to investigate a more suitable model for Banks.

The SVM does not provide good predictions. Its highest accuracy is merely 57.7% in Pharmaceutical industry. Since there are many companies in each industry, they each need to differentiate themselves to find a niche in the market. This may results in companies not similar with each other even if they are in the same group/ industry. This compromises the model's ability to accurately track the industry-wise fluctuation. For industries like Banks and Real estate, this method is almost invalid.

The logistic model performs decently well in predicting almost every industry. The top two industries are Real estate and Banks, with around 87% prediction accuracies for both. Since logistic model can adopt the linear relationships between indicators it is better than previous models. And the actual results show that both financial indicators and economic indicators are correlated, which are all parts of the whole interactive system.

Naive Bayes outperforms SVM overall but has weaker functionality than other methods such as Logistic method and Neutral Network method. By applying this methods to 6 industries, we can see Naive Bayes has the highest prediction rate in Capital Goods industry and the lowest prediction rate in Banks industry. For the other 4 industries, it performs at an average level. Hence, it is better to use Naive Bayes method to predict the stock markets in Capital Goods industry than Banks industry.

Neural Network performs the best for Real Estate industry and the worst for Banks. For other three industries, the accuracy are all above 50%. The neural network neutralizes the multicollinearity and does increase the accuracy in predicting healthcare industry, compared to decision tree model.

LDA has an equally decent performance in all industries. Comparing the results given by LDA and those of other methods, it is obvious that LDA, with a variance of 25.5, has a much more stable performance across all industries. This confirms the stability, a major property of LDA. Notice that LDA gives the best prediction among all models in healthcare and biotechnology industries. This could result from the fact that healthcare sector is engaged in the production and delivery of medicine and health care–related goods and services. The constant needs make the sector less sensitive to economic cycles. As a result, LDA's assumption on

normal distribution as well as equal covariance are probably easier to satisfy under the stable performance of the industry.

Conclusion:

We generate a general performance visualization by combining the seven models together:

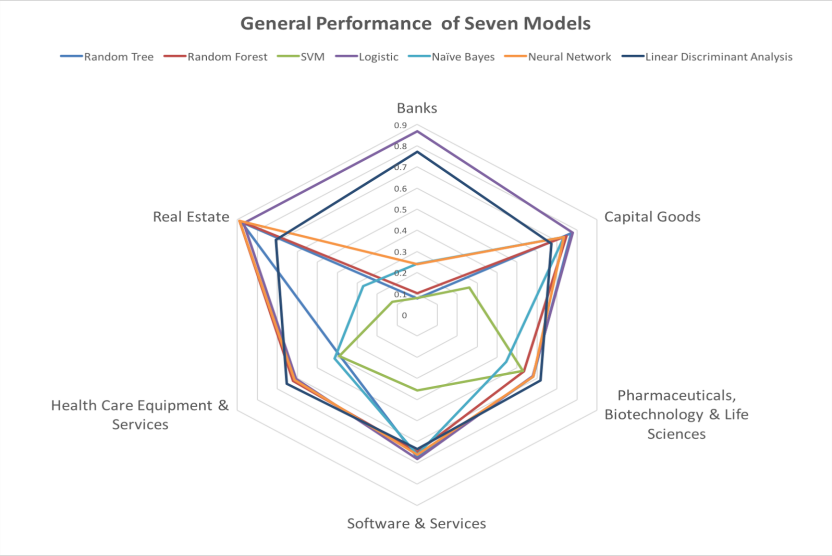


Chart 5-1 Prediction Accuracy of Seven Models

In this radar chart, different colors indicate the different models we use. Overall, both logistic regression and LDA do a good job in our prediction. They are both applicable in the prediction of all six industries. Most models do not perform well on banks, which is possibly due to the differences between financial institutions in the market. Furthermore, SVM fails to predict fluctuations/outcomes of something. we suggest investors to avoid this method when drafting their investment strategies.

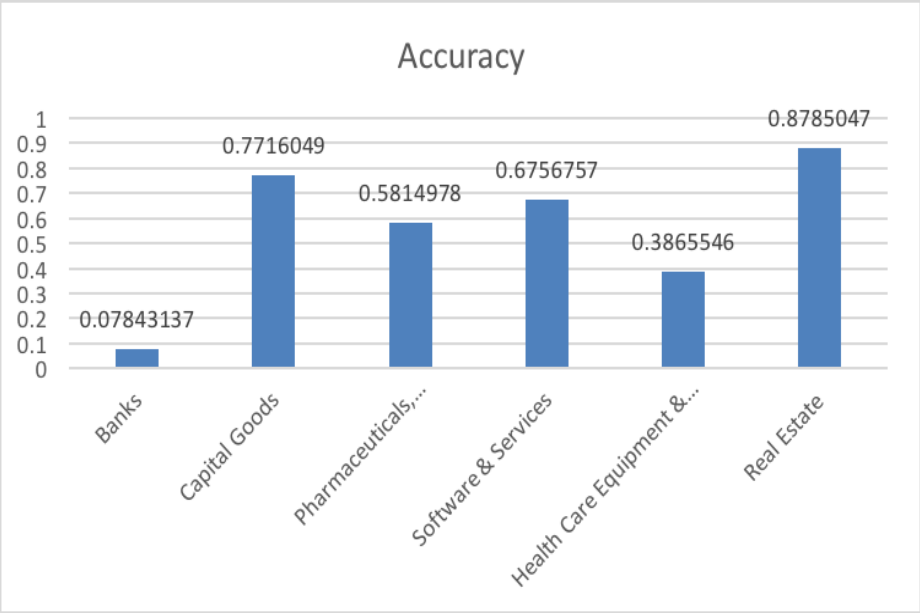
To conclude, one who invests in Banks, Software & Services industries should consider Logistic Regression. In regards to Capital Goods, Pharmaceuticals, Biotechnology & Life Sciences and Health Care Equipment & Services, Linear Discriminant Analysis Model is a good predictive tool. Investors who are interested in Real Estate industry can utilize Neural Network in their decision making process.

Though we adopt many factors(macroeconomic impacts/industrial impacts) when modeling and predicting stock returns, there are still limitations in our analysis. For instance, we did not consider the inter-industrial impacts when modeling. As a result we might have neglected the existence of the industrial synergy. Moreover, many companies are global giants and are more sensitive to international market impacts. Politics also plays an critical role in financial market. All these factors demand more in-depth research and we look forward to improve our analysis in the future.

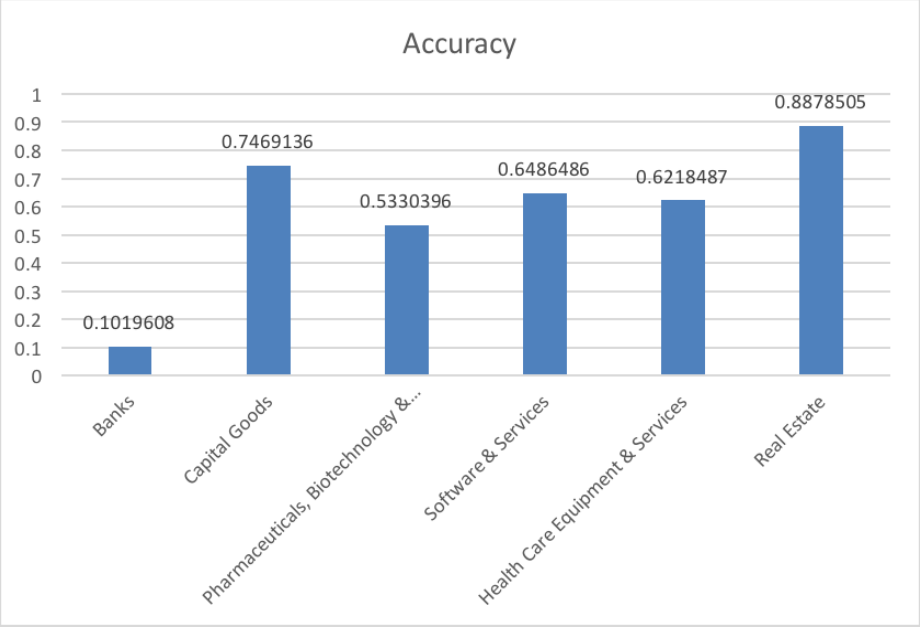
Acknowledgement:

We thank Prof. Zhiliang Ying for the great instructions and insightful suggestions during the whole project.

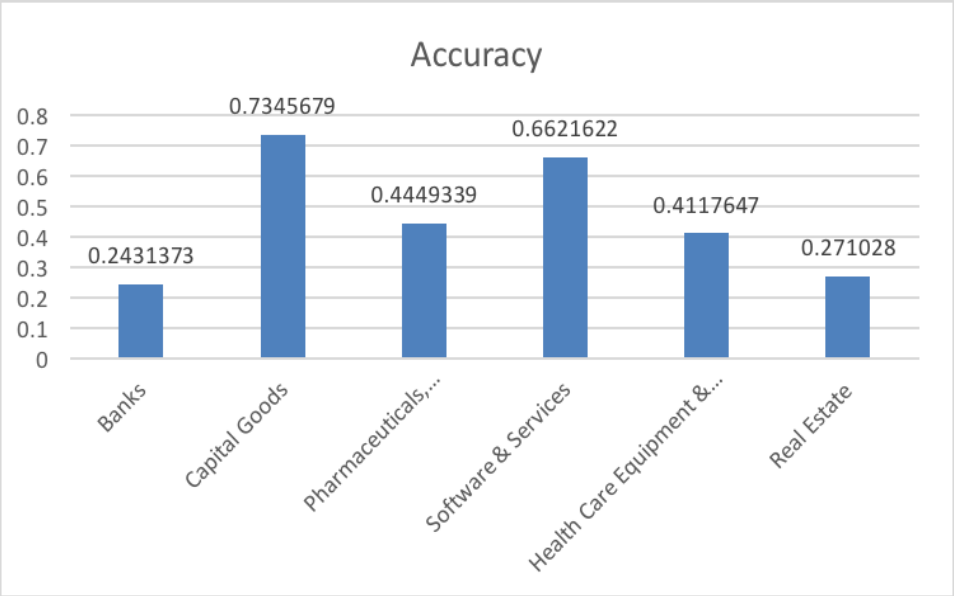
Attachment:
(Performance bar plots)
1. Decision Trees:



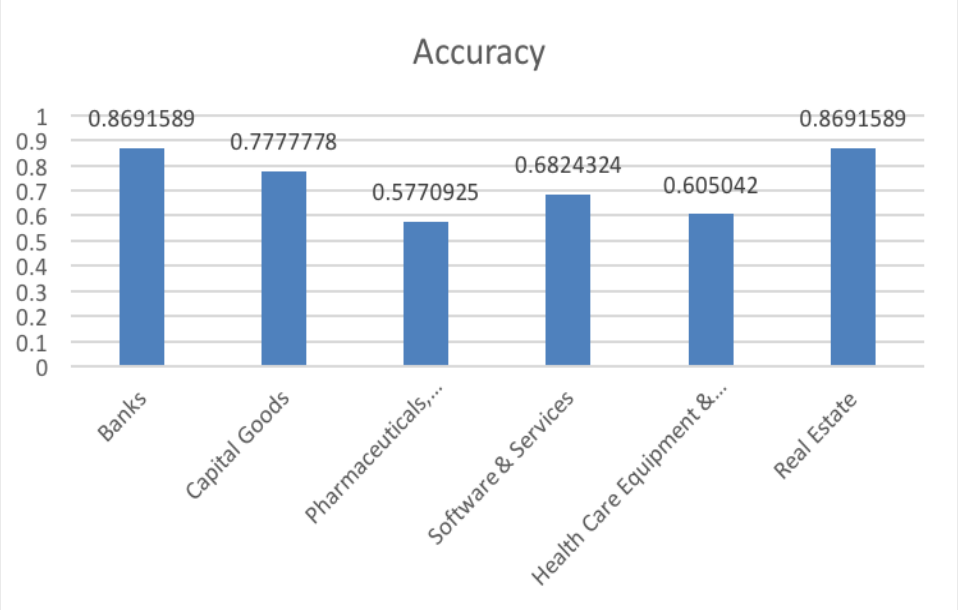
2. Random Forest:



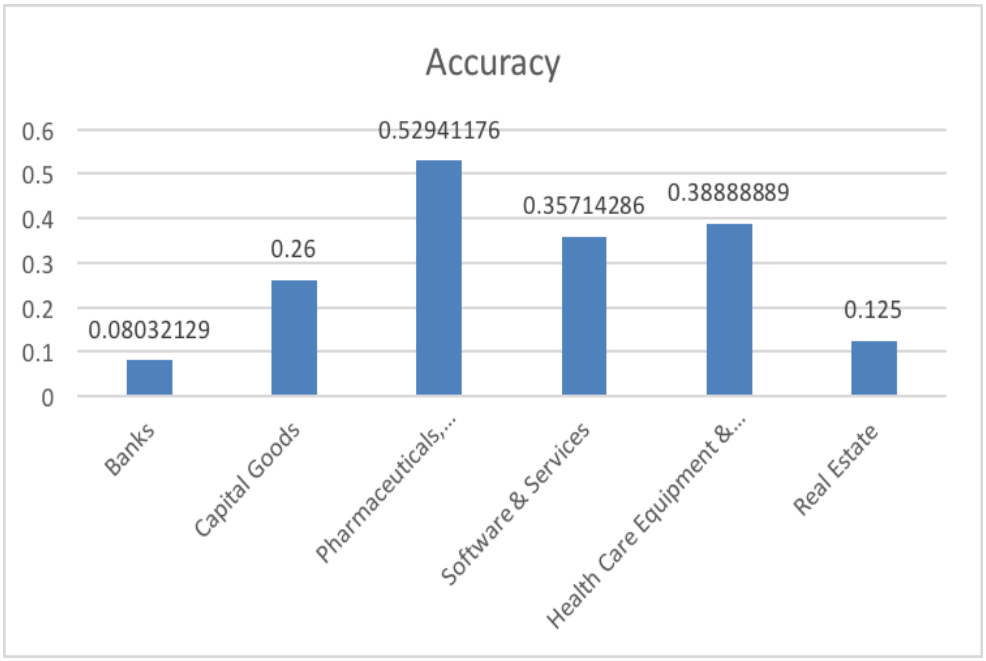
3. Naive Bayes:



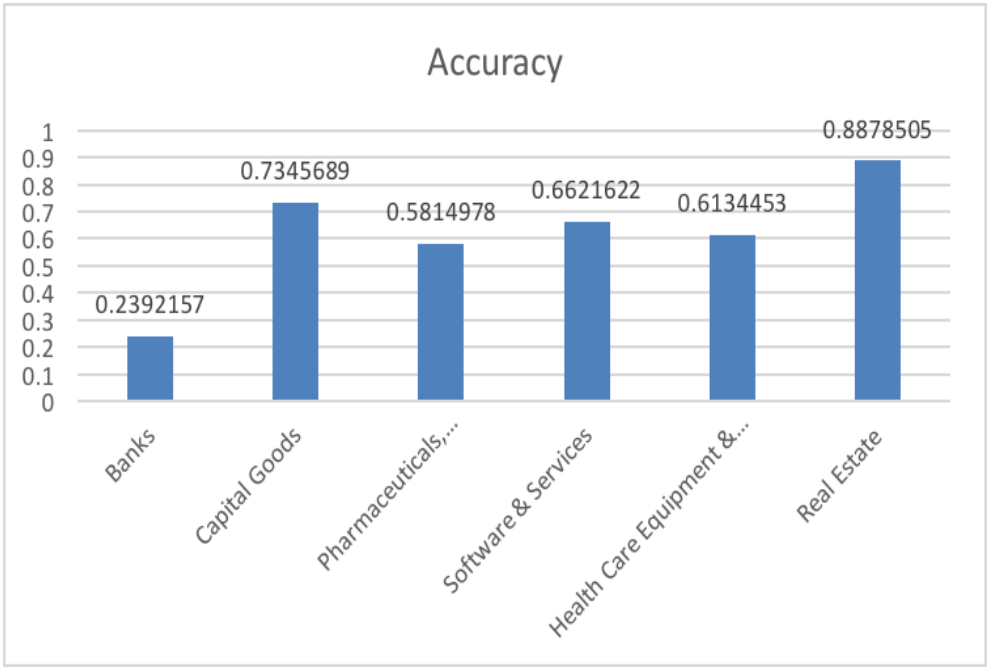
4. Logistic Regression:



5. Support Vector Machine:



6. Neural Network:



7. Linear Discriminant Analysis: