Stock Market Prediction Using Machine Learning Techniques

Jemish Naliyapara
Department of Physics, IISER, Bhopal
Roll No.:18114
E-mail Id: jemish18@iiserb.ac.in

***Abstract*—Stock market prediction is the act of trying to determine the future value of company stock or other financial instrument traded on a financial exchange. The successful prediction of a stock's future price will maximize investor's gains. This paper proposes machine learning models to predict stock market price. In this paper, I investigated the predictability of the National Stock Exchange Index(NIFTY 50). In this project, seven prediction models are proposing using historical data to predict the stock market movements. The proposed supervised models are Naive Bayes, Linear Regression, Logistic Regression, Support Vector Machine (SVM), Random Forest, and Artificial Neural Network (ANN) and Unsupervised machine learning tachnique: Principal Component Analysis (PCA). I analyzed the results and its overall performance of these models. I compared the results of these models with each other and found that ANN give a good prediction of the stock market. In term of returns strategies, Random forest gives excellent performance compares to other models.**

***Index Terms*—National Stock Exchange(NSE), Naive Bayes, Linear Regression, Logistic Regression, Random Forest, Artificial Neural Network(ANN) and Principal Component Analysis.**

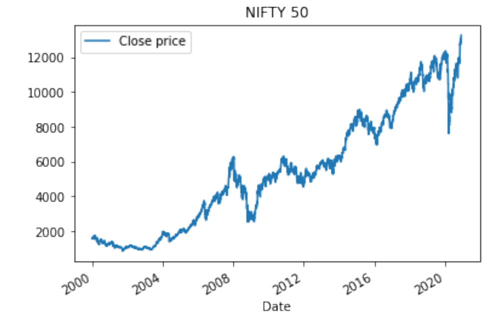
I. Introduction

Quantitative traders with much money from stock markets buy stock derivatives and equities at a low price and later sell them at a high price. The trend in a stock market prediction is not a new thing, and yet this issue is kept being discussed by various organizations. There are two types to analyze stocks that investors perform before investing in a stock. First is the fundamental analysis; in this analysis, investors look at the intrinsic value of stocks and performance of the industry, economy, political, etc., to decide that whether to invest or not. On the other hand, technical analysis is an evolution of stocks employing studying the statistics generated by market activity, such as past prices and volumes.

In recent years, the increasing prominence of machine learning in various industries has enlightened many traders to apply machine learning techniques to the field. Some of them have produced quite promising results.

Stock Market follows the random walk, which implies that the best prediction you can have about tomorrow's value is today's value. Indisputably, forecasting stock indices is very difficult because of the market volatility that needs an accurate forecast model. The stock market indices are highly fluctuating, and it affects the investor's belief. Stock prices are considered to be very dynamic and susceptible to quick changes because of the underlying nature of the financial domain and the mix of known parameters like the previous day's closing price, P/E ratio, and unknown factors like Election results, Rumors etc. It would also affect the stock market prediction using machine learning models.

In this project, I used the dataset of the Indian stock market index (NIFTY 50) in a period between 3^{rd} January 2000 to 4^{th} December 2020. In this dataset, features like Close price, Open price, Volumes, P/E ratio etc., would be beneficial for the prediction.



In this paper, I am using Machine Learning techniques, i.e. Naive Bayes, Linear Regression, Logistic Regression, SVM, Random Forest, ANN, and PCA, to predict the stock market. I analysed the results of these models and discussed them in this paper.

II. Methodology

Stock market prediction seems a complex problem because there are many factors that have yet to be addressed and it doesn't seem statistical at first. But by proper use of machine learning techniques, one can relate previous data to the current data and train the machine to learn from it and make appropriate assumptions. Machine learning as such has many models but this paper focuses on seven of them and made the predictions using them.

A. Naive Bayes

Naive Bayes is supervised machine learning technique. The Naive Bayes model usually used for classification in machine learning. A Naive Bayes classifier is a probabilistic machine learning model that's used for a classification task. The crux of the classifier is based on the Bayes theorem.

If the Naive Bayes model used for classification, how can I apply it in stock market prediction? So I made the strategy to use the probabilistic approach in daily return in the stock market. Using the daily return feature, I classified whether tomorrow's market goes up or down.If the market goes up, it gives positive returns sign (+1), and if the market goes down, it gives the sign of a negative return (-1). After applying the Naive Bayes model, I can predict tomorrow's market go up or down with an accuracy of 52.87% and F1 Score 65.10%.

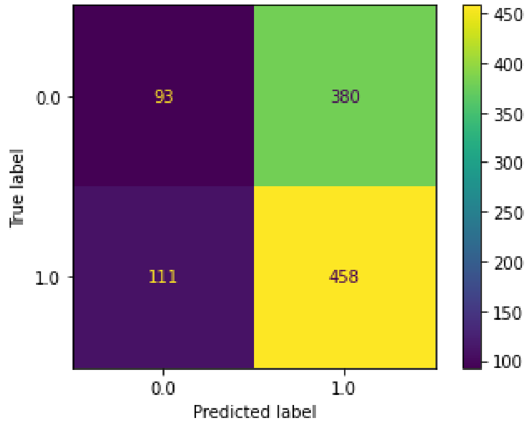


Fig. 1: Confusion Matrix

The disadvantage of this model is that it is biased toward the positive return because the share market index always shows an upward moment in the long period as the country grows. Another disadvantage of this model is, it cannot predict how much the market goes up or down. This model is not very reliable for investment purpose.

B. Linear Regression

Linear Regression is a machine learning algorithm based on supervised learning. It performs a regression task. It is mostly used for finding out the relationship between variables and forecasting. Linear regression performs the task to predict a dependent variable value (y) based on a given independent variable (x). Linear regression can be used to find a relationship between two or more variables of interest and make predictions once these relationships are found.

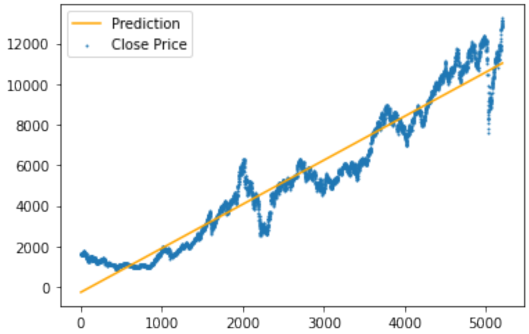


Fig. 2: Linear Regression

After applying the linear regression technique on the dataset, I found the relationship between the closing price of nifty 50 and its day. I found the line of best fit with slope 2.17048. I got R^2 value 0.934. Which is pretty good and means this model did pretty well explaining the variance in the data.

The limitation of this model is that Simple linear regression will not make you so much money. Relying on this strategy will most likely make you lose significantly more than you would win. This linear regression model will fail if any random and unpredictable event occurs.

C. Logistic Regression

Logistic regression falls under the category of supervised learning; it measures the relationship between the categorical dependent variable and one or more independent variables by estimating probabilities using a logistic/sigmoid function. The logistic regression model computes a weighted sum of the input variables similar to the linear regression. However, it runs the result through a particular non-linear function, the logistic function or sigmoid function, to produce the output y and give the results in binary form.

I made the strategy to use logistic regression to predict the stock market. My strategy is that If tomorrow's closing price is higher than today's closing price, then a trader will buy the stock put it in category(1), else he will sell it (-1). If the output is 0.7, then we can say that there is a 70% chance that tomorrow's closing price is higher than today's closing price and classify it as 1.

This model predicts the signal to buy (1) or sell (-1). I calculated cumulative Nifty 50 returns and the cumulative strategy return based on the signal predicted by the model in the dataset.

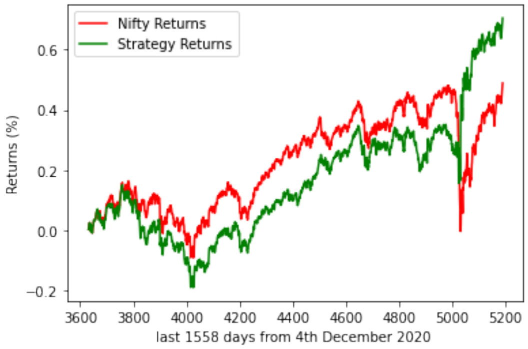


Fig. 3: Logistic Regression

It can be observed that the logistic regression model in python predicts the classes with an accuracy of 53.91% and generates good returns. from the figure.3, we can say that this strategy is beneficial in the extended period. This strategy gives promising results when we apply it with individual stock.

D. Support Vector Machine (SVM)

Support Vector Machine or SVM is one of the most popular supervised learning algorithms used for classification and regression problem. The objective of the support vector machine algorithm is to find a hyperplane in N-dimensional space (N is the number of features) that distinctly classifies the data points.

I used the SVM model in stock market prediction. In this model, I have used the radial basis function kernel (RBF kernel), a popular kernel function used in various kernelised learning algorithms.
Formula for RBF kernal:

$$K(x, x') = \exp^{-\gamma|x-x'|^2}$$

In the python programme, we have to tune the Regularization parameter (C) and gamma value. C is inversely proportional to the strength of the regularization, and gamma is the coefficient of the kernel. While adjusting this parameter, I observed that the predicted value by the SVM model is very fluctuating for the low value of C. When the C value is 10^5, and the gamma value is 10^{-6}, it gives excellent results.

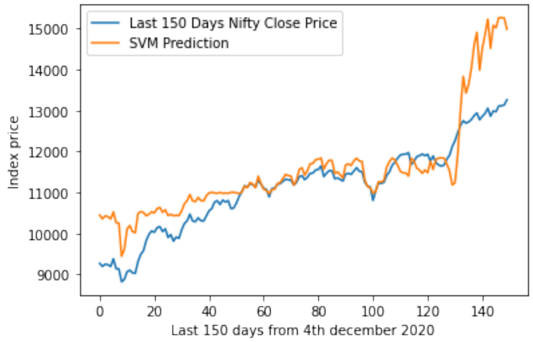


Fig. 4: SVM [Regularization Parameter (C)=1e5,gamma=1e-6]

As you can see in the figure, the SVM model predicts the stock market with a confidence level of 0.9515, which is a promising result.

E. Random Forest

Random forest is a supervised classification machine learning algorithm that uses the ensemble method. Ensemble simply means a group or collection, which in this case, is a collection of decision trees. A random forest is made up of numerous decision trees and helps to tackle the problem of overfitting in decision trees. These decision trees are randomly constructed by selecting random features from the given dataset.

Random forest arrives at a decision or prediction based on the maximum number of votes receives from the decision trees. The outcome which is arrived at, a maximum number of times through the numerous decision trees is considered as the final outcome by the random forest.

Since Random forest is a classification model, I need to make the strategy to implement random forest to predict the stock market. I have used '(Open-Close)/Open','(High-Low)/Low', a standard deviation of last 5 days returns, and the average of last 5 days returns as an input variable. I made classification define as if tomorrow's close price is greater than today's close price then output variable is set to 1 and otherwise set to -1.here 1 indicates to buy the stock and -1 indicates to sell the stock.

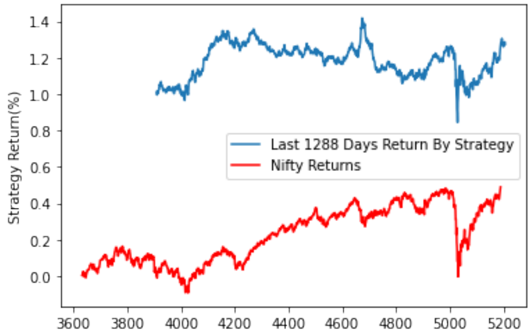


Fig. 5: Random Forest

As you can see in the figure, this strategy makes excellent returns as compare with the NIFTY 50 returns. The Random Forest model gives great results compare to other models in machine learning techniques. This model classifies data with 51.46% accuracy but, It gives quite promising returns.

F. Artificial Neural Network (ANN)

Artificial Neural Network (ANN), is one of the intelligent data mining techniques that identify a fundamental trend from data and generalize from it. ANN is capable of simulating and analysing complex patterns in unstructured data as compared to most of the conventional methods. The model uses the basic structure of a Neural Network having neurons with different layers. The model work with three layers. It consists of the input layer, hidden layer and output layer. The input layer consists of variables which are High price, Low price, Open price, Volumes etc. The weights on each input load are multiplied and added and sent to the neurons. The hidden layer or the activation layer consists of these neurons. The total weight is calculated and is moved to the third layer which is the output layer. The output layer consists of only one neuron which will give the predicted value in terms of the closing price of the stock.

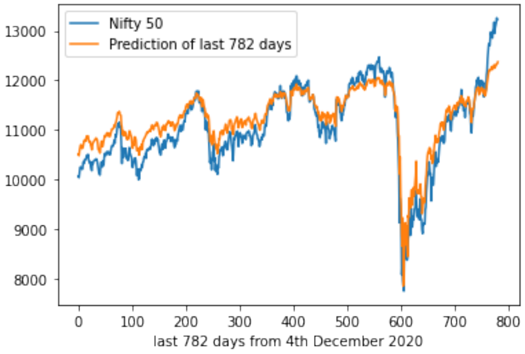


Fig. 6: Artificial Neural Network

I made an ANN model with Close and Open price which mean my model predicts tomorrow's open price based on today's closed price. This model used 'adam' optimizer and 'mean squared error' as loss function to predict stock market price. This model predicts price as seen in the above figure. As we can see in the figure, the predicted price is higher than the Nifty50 Opening price. The drawback of this model is that ANN acts like the Black box, so traders cannot know the reason behind the prediction.

G. *Unsupervised Learning: Principal Component Analysis*

Principal component analysis (PCA) is a statistical technique to convert high dimensional data to low dimensional data by selecting the essential features that capture complete information about the dataset. The features are selected on the basis of variance that they cause in the output. The feature that causes the highest variance is the first principal component. The feature responsible for the second-highest variance is considered the second principal component, and so on. It is important to mention that principal components do not have any correlation with each other.

I made binary class if tomorrow's close price is higher than today's close price put in 1 (buy signal); otherwise, set in -1 (sell signal). I applied the PCA model to the dataset made in the random forest model. I tried to predict the stock market returns using PCA. I used in strategy, which is discussed in the Random Forest model. I found that it gives a little higher accuracy in predicting a class compared to the Random Forest class prediction.

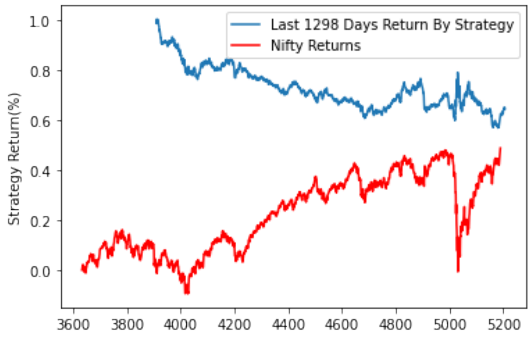


Fig. 7

This model predicts class with 51.77% accuracy and 54.96% F1-Score, which is higher than the Random Forest model. In this model, the first principal component is responsible for 51.28% variance. Similarly, the second principal component causes a 36.72% variance in the dataset. Collectively we can say that (51.28+36.72) 88% of the classification information contained in the feature set is captured by the first two principal components.

We can see that this model's Returns consistently go down compared to Nifty 50 returns in the figure. This model gives poor returns compared to returns made by the Random Forest model.

III. Conclusion

Predicting the stock market and its returns are challenging tasks due to consistently changing stock values dependent on multiple parameters that form complex patterns. The historical dataset of Nifty 50 consists of only a few features like high, low, open, close, volumes etc. which are not sufficient enough. New variables have been created using the existing variables to obtain higher accuracy in the predicted price value. In this project, the Naive Bayes model predicts tomorrow's stock market goes up or down using probability. The linear regression model just gives the general trend of the stock market. It is not a reliable model for trading as compared to other model described in this project. We can make quite promising returns using logistic regression compared to Nifty 50 returns. But this model gives poor returns when we compared returns make by the Random Forest model. SVM make a good prediction of the stock market but, the problem with this model is that the predicted value of the stock market is very fluctuating. It is very risky for the traders to trade with a high fluctuating value. Random Forest gives excellent returns with the strategy as discussed in this project. It gives a very high return as compared to Nifty 50 returns. ANN makes the best prediction of the stock price compared to other techniques as discussed in this project. Principal Component Analysis (PCA) model gives poor returns as compared to the returns made by the Random Forest model. We conclude that the Random Forest model gives the best returns strategy compared to the other model discussed in this project, and ANN makes an excellent prediction of the stock market. I firmly believed that a machine learning model could be developed which consider financial news articles along with financial parameters such as a closing price, traded volumes, profit and loss statements, balance sheet of various companies etc., for possibly better results.

Acknowledgment

I am grateful to Dr. Kushal Kumar Shah for providing me an opportunity of a project which gave me a platform to try and use the skills learnt in our academic curriculum for some original work.

References

[1] DataSet:
https://www.kaggle.com/sudalairajkumar/nifty-indices-dataset?select=NIFTY+50.csv

[2] Naive Bayes Classifier:
https://towardsdatascience.com/naive-bayes-classifier-81d512f50a7c

[3] Linear Regression:
Website-1:https://www.geeksforgeeks.org/ml-linear-regression/#:~:text=Linear%20Regression%20is%20a%20machine,relationship%20between%20variables%20and%20forecasting.&text=Hence%2C%20the%20name%20is%20Linear%20Regression.
Website-2:https://medium.com/analytics-vidhya/stock-prediction-using-linear-regression-cd1d8351f536#:~:text=Now%20we%20can%20use%20Scikit,predictions%20for%20each%20x%20value.&text=We%20can%20plot%20the%20actual,line%20on%20top%20of%20it.

[4] Logistic Regression:
https://blog.quantinsti.com/machine-learning-logistic-regression-python/

[5] Support Vector Machine (SVM):
Website-1:https://medium.com/@rupesh1684/stock-market-prediction-using-machine-learning-model-svm-e4aaca529886
Website-2:https://scikit-learn.org/stable/modules/svm.html

[6] Random Forest:
https://blog.quantinsti.com/random-forest-algorithm-in-python/

[7] Mahar Vijh, Deeksha Chandola, Vinay Anand Tikkiwal, Arun Kumar paper on ”Stock Closing Price Prediction using Machine Learning Techniques” publised in ScienceDirect, ELSEVIER

[8] Principal component Analysis (PCA):
https://stackabuse.com/implementing-pca-in-python-with-scikit-learn/