Stock Market Prediction Description

INTRODUCTION

Prediction of stock price has been the most researched topic by the researchers from different fields. The most common way is to use machine learning techniques. There are various popular machine learning techniques which can be used to predict the stock price of the company such as regression, SVM and neural networks. The regression technique is the most used and widely accepted when it comes to prediction, as it is easy to understand and implement. In this project, we propose a neural network and multi-linear regression algorithm to predict the stock price of the company for any day by using their historical stock data. We would be finding the correlation between the independent factors (date, highest price, lowest price, opening price, closing price, volume of stocks) and the dependent factor (price).

Neural Network

Artificial neural networks (ANNs) or connectionist systems are computing systems inspired by the biological neural networks that constitute animal brains. Such systems learn (progressively improve performance) to do tasks by considering examples, generally without task-specific programming.

Neural networks, with their remarkable ability to derive meaning from complicated or imprecise data, can be used to extract patterns and detect trends that are too complex to be noticed by either humans or other computer techniques.

Regression

Regression analysis is widely used for prediction and forecasting, where its use has substantial overlap with the field of machine learning. Regression analysis is also used to understand which among the independent variables are related to the dependent variable, and to explore the forms of these relationships. In restricted circumstances, regression analysis can be used to infer causal relationships between the independent and dependent variables. However this can lead to illusions or false relationships, so caution is advisable; for example, correlation does not imply causation.

Linear Regression

In linear regression, the relationships are modeled using linear predictor functions whose unknown model parameters are estimated from the data. Such models are called linear models. Most commonly, the conditional mean of y given the value of X is assumed to be an affine function of X; less commonly, the median or some other quartile of the conditional distribution of y given X is expressed as a linear function of X. Like all forms of regression analysis, linear regression focuses on the conditional probability distribution of y given X, rather than on the joint probability distribution of y and X, which is the domain of multivariate analysis.

$$\mathbf{y} = X\boldsymbol{\beta} + \boldsymbol{\varepsilon},$$

Multi – Linear Regression

It is the oldest and most widely used predictive model in the field of machine learning. The goal is to minimize the sum of the squared errors to fit a straight line to a set of data points. Linear regression model that contains more than one predictor variable is called a multiple linear regression model. The following model is a multiple linear regression model with two predictor variables, x_1 and x_2.

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon$$

The model is linear because it is linear in the parameters β_0, β_1 and β_2. The model describes a plane in the three-dimensional space of Y, x_1 and x_2. The parameter β_0 is the intercept of this plane. Parameters β_1 and β_2 are referred to as partial regression coefficients. Parameter β_1 represents the change in the mean response corresponding to a unit change in x_1 when x_2 is held constant. Parameter β_2 represents the change in the mean response corresponding to a unit change in x_2 when x_1 is held constant.

Backward Elimination

It involves starting with all candidate variables, testing the deletion of each variable using a chosen model fit criterion, deleting the variable (if any) whose loss gives the most statistically insignificant deterioration of the model fit, and repeating this process until no further variables can be deleted without a statistically significant loss of fit.

Factors Affecting Stock

There are many factors which can affect the stock price, they can be news releases on earnings and profits, and future estimated earnings, announcement of dividends, introduction of a new product or a product recall, securing a new large contract, employee layoffs, anticipated takeover or merger, a change of management, accounting errors or scandals, and the amount of stocks bought in a particular day. In this paper we would be only considering the amount of stocks bought in a particular day as a key factor, the rest would be ignored.

Collecting Data

The data set used in this project is collected from NASDAQ, Sensex and Nifty 50. It covers daily price from 01-Aug-2017 to 31-Aug-2017: Since the markets are closed on holidays which vary from country to country, we have used NASDAQ, Sensex and Nifty 50 stock prices data. We would be predicting the values of these stock prices for the days from 01-Aug-2017 to 31-Aug-2017.

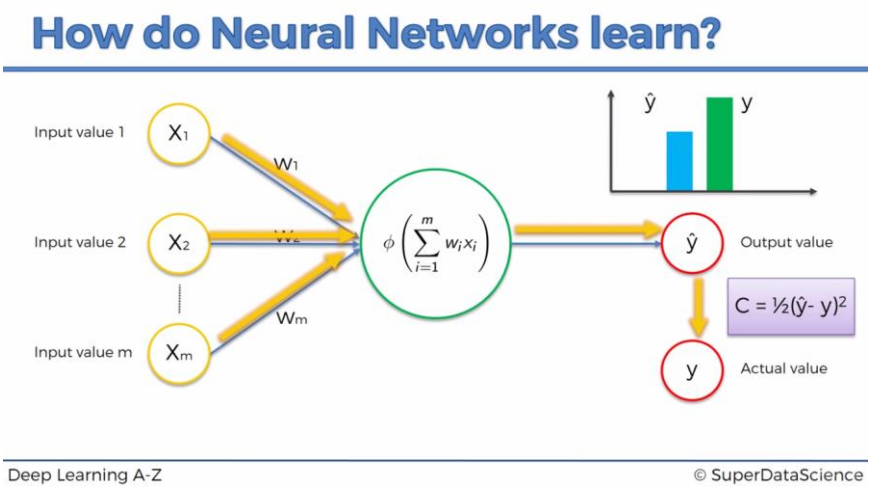
Algorithm

In this project we are predicting the stock market price by using two algorithms

- Neural Network

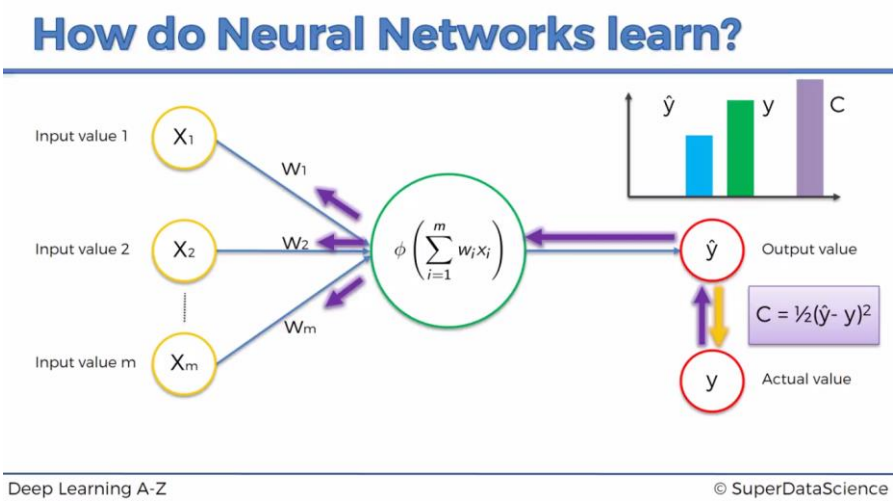
  In neural networks, each input variable or independent variable is a node.

  For every input, there is a weight associated to it. The weighted sum of the inputs is taken and an activation function is applied on it. This is Forward Propagation.



Forward Propagation

The cost function, which calculates the deviation of the calculated value from original value is calculated and the weights are re-adjusted to minimize the cost function value. This is called **Backward Propagation**



Backward Propagation

- Multi-Linear Regression model.

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon$$

In this equation, β_0, β_1 and β_2 act as parameters while x_1 and x_2 are variables. Now since we have assumed that Date, Open, High, Low, Close and Volume are independent factors which affect the Adj Close (Dependent Factor). Hence, these independent factors would be the variables in the Multi-Linear Regression equation.

Therefore,

X_1 is Date;

X_2 is Opening Price of the Stock;

X_3 is Highest Price of the Stock;

X_4 is Low;

X_5 is the Closing Price of the Stock;

X_6 is the Volume of the Stock Purchased;

And, Y is the Adj Close or Price of the Stock.

Now when substituting the above in the Multi-Linear regression equation, we would be obtaining the value of the parameters which would help us in predicting the stock prices. The values of the variables would be obtained from the data set. When we apply this algorithm on our data set we would be able to predict the stock prices of the company.

Backward Elimination

In this project, our aim is to predict the stock prices with minimum error and memory wastage. Now we have predicted the stock prices in the previous step, now we work upon the minimum memory wastage. In any general multi-regression algorithm, the first step we do is assume that all the factors would be affecting the stock price, hence in this step we would start eliminating those independent factors which don't affect the stock price. Now these factors could be anything, there are chances that not even one factor would get eliminated or all of them would get eliminated. For the backward elimination, we need to set a fixed threshold of the P-test, which we are taking as 0.05. This step occurs after the set-up of multi-linear regression equation. Now, in the following images we will be showing which factor is eliminated and which factor stays in the algorithm.

Initial Regression Table

```
Residuals:
       Min         1Q     Median         3Q        Max
-2.689e-13 -1.539e-13 -5.168e-14  1.248e-13  8.670e-13

Coefficients:
             Estimate Std. Error    t value Pr(>|t|)
(Intercept)  6.307e+03  8.213e-14  7.680e+16   <2e-16 ***
Date        -2.593e-13  1.954e-13 -1.327e+00    0.214
Open         1.317e-14  3.226e-13  4.100e-02    0.968
High         9.749e-14  4.159e-13  2.340e-01    0.819
Low         -9.794e-14  4.191e-13 -2.340e-01    0.820
Close        5.719e+01  3.143e-13  1.819e+14   <2e-16 ***
Volume      -2.141e-13  1.666e-13 -1.285e+00    0.228
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.386e-13 on 10 degrees of freedom
Multiple R-squared:      1,     Adjusted R-squared:      1
F-statistic: 7.606e+28 on 6 and 10 DF,  p-value: < 2.2e-16
```

Initial Regression

Now as it can be seen the Open variable is having the P-value 0.968 hence we would be eliminating that factor.

Backward Elimination 1

```
       Min         1Q     Median         3Q        Max
-2.739e-13 -1.416e-13 -6.800e-14  1.200e-13  8.452e-13

Coefficients:
              Estimate Std. Error    t value Pr(>|t|)
(Intercept)  6.307e+03  7.623e-14  8.273e+16   <2e-16 ***
Date        -2.550e-13  1.649e-13 -1.546e+00    0.150
High         8.198e-14  2.755e-13  2.980e-01    0.772
Low         -6.849e-14  3.642e-13 -1.880e-01    0.854
Close        5.719e+01  2.377e-13  2.406e+14   <2e-16 ***
Volume      -2.030e-13  1.535e-13 -1.322e+00    0.213
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.143e-13 on 11 degrees of freedom
Multiple R-squared:      1,     Adjusted R-squared:      1
F-statistic: 1.059e+29 on 5 and 11 DF,  p-value: < 2.2e-16
```

Backward Elimination 1

Now after removing the Open variable coefficient, we need to remove the next maximum P value variables which are Low and High.

Backward Elimination 2

```
Residuals:
       Min         1Q     Median         3Q        Max
-2.780e-13 -1.464e-13 -8.954e-14  1.261e-13  8.521e-13

Coefficients:
              Estimate Std. Error    t value Pr(>|t|)
(Intercept)  6.307e+03  7.021e-14  8.983e+16   <2e-16 ***
Date        -2.274e-13  1.113e-13 -2.044e+00   0.0618 .
Close        5.719e+01  8.170e-14  7.000e+14   <2e-16 ***
Volume      -1.687e-13  1.036e-13 -1.628e+00   0.1275
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 2.895e-13 on 13 degrees of freedom
Multiple R-squared:      1,     Adjusted R-squared:      1
F-statistic: 2.081e+29 on 3 and 13 DF,  p-value: < 2.2e-16
```

Backward Elimination 2

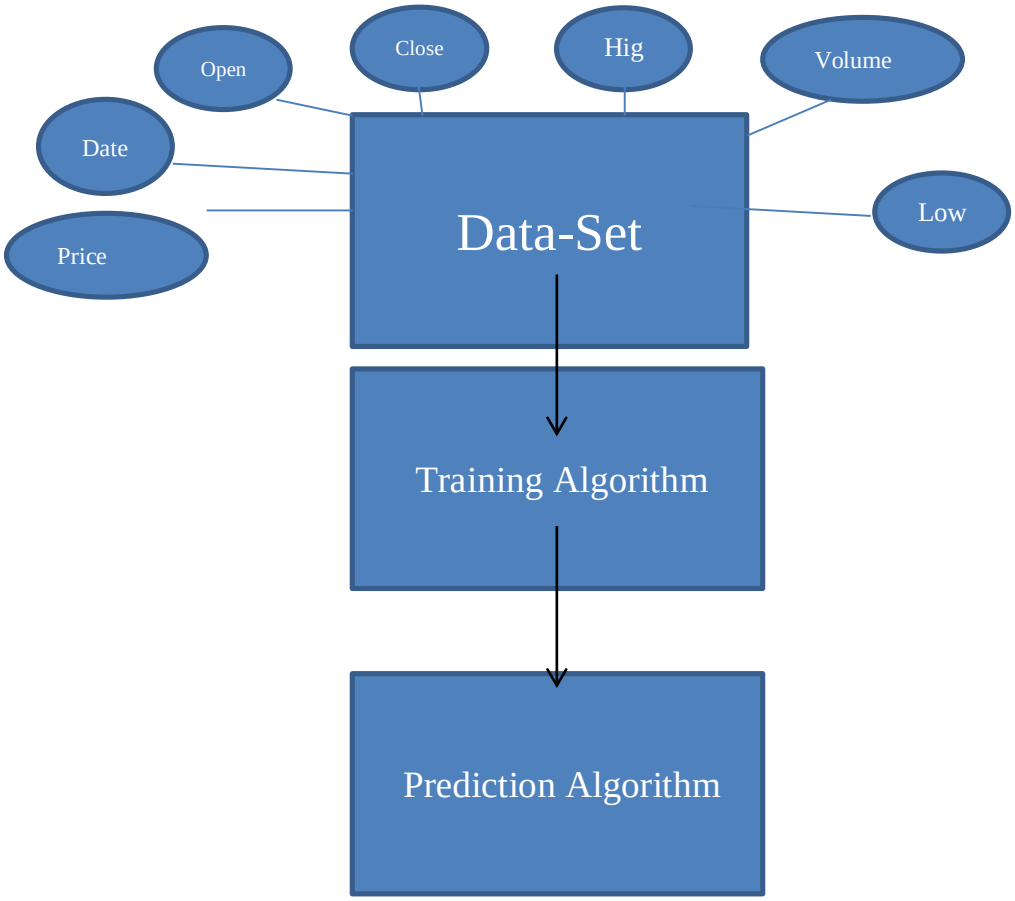
Now from the above it can be seen that all the remaining variables meet the minimum threshold of 0.5.

Assumptions

The data set obtained from NASDAQ, Sensex and Nifty 50 have Date, Open, High, Low, Close, Volume and Adj Close. Now we are assuming that the Date, Open, High, Low, Close and Volume are independent factors which affect the Adj Close (Dependent Factor). Hence, these

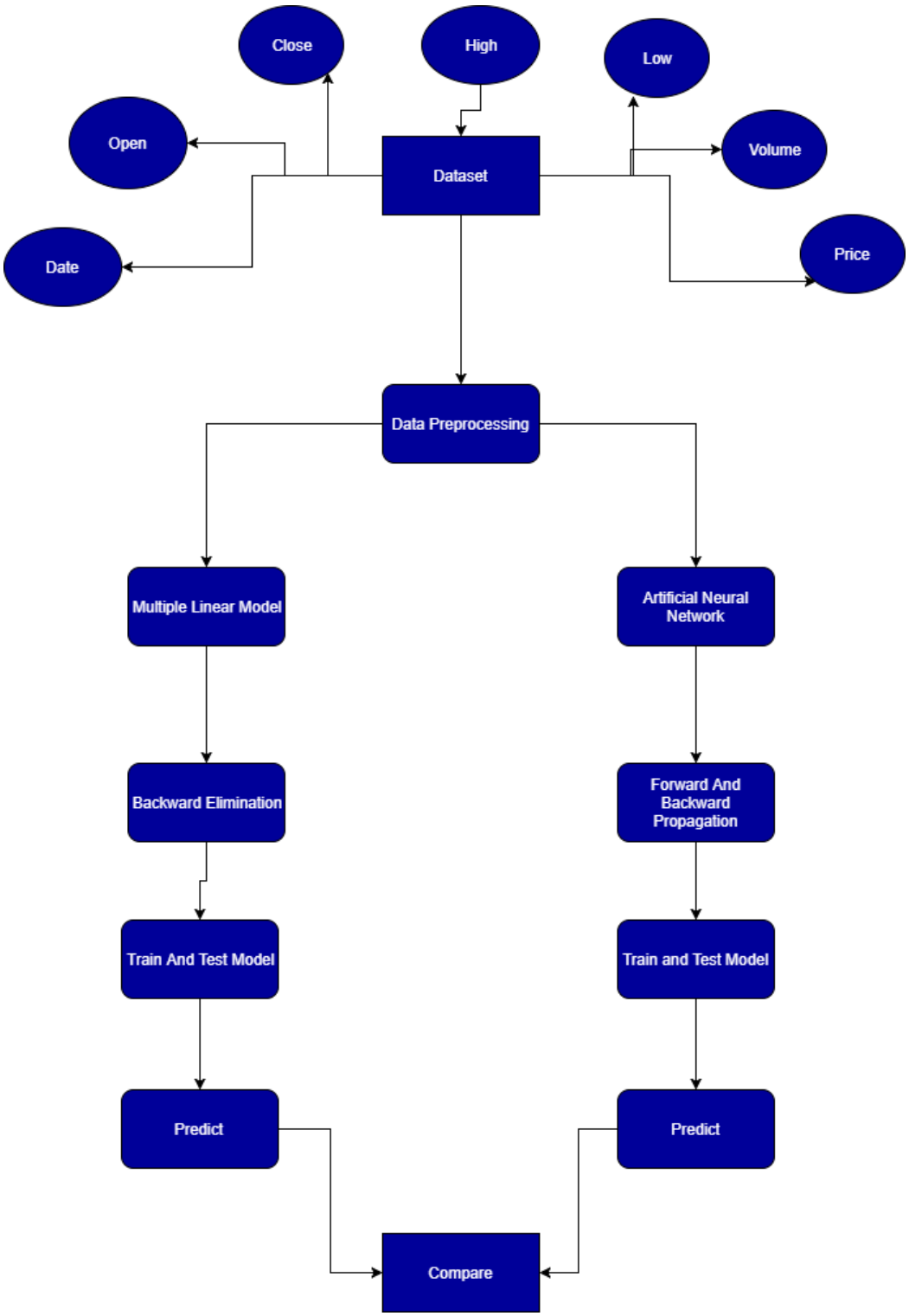
independent factors would be the variables in the Multi-Linear Regression equation and as nodes in Neural Network.

High Level Design



High Level Design

Low Level Design



Low Level Design

Test Case Generation

In this step, we would be testing our model with the new dates. After obtaining the results from the model, we will compare the test data with the real time data and find out the difference. Our acceptance criteria is that the model should provide at least 70% accuracy.

Testing

Multi-Linear Regression

Nasdaq

Date	Actual	Predicted	Error
7 August 2017	6383.77	6381.93	0.0288%
10 August 2017	6216.87	6227.46	0.1703%
15 August 2017	6333.01	6334.95	0.3065%
22 August 2017	6297.48	6302.06	0.0728%
28 August 2017	6283.02	6288.68	0.0901%

Comparison of the Original and the Tested Value

Average Error Percentage: 0.0785 %

Sensex

Date	Actual	Predicted	Error
7 August 2017	32273.67	32523.02	0.7726%
10 August 2017	31531.33	31448.43	0.2629%
15 August 2017	31770.89	31795.21	0.0765%
22 August 2017	31568.01	31501.52	0.2106%
28 August 2017	31646.46	31615.09	0.0991%

Comparison of the Original and the Tested Value

Average Error Percentage: 0.2843 %

Nifty 50

Date	Actual	Predicted	Error
7 August 2017	10057.40	10109.86	0.5216%
10 August 2017	9820.25	9790.07	0.3072%
15 August 2017	9897.30	9893.97	0.0335%
22 August 2017	9852.50	9833.56	0.1921%
28 August 2017	9884.40	9876.58	0.0791%

Comparison of the Original and the Tested Value

Average Error Percentage: 0.2267 %

Neural Network

Nasdaq

Date	Actual	Predicted	Error
7 August 2017	6383.77	6373.32	0.1636%
10 August 2017	6216.87	6248.91	0.5153%
15 August 2017	6333.01	6325.44	0.1194%
22 August 2017	6297.48	6294.01	0.0549%
28 August 2017	6283.02	62860.87	0.3525%

Comparison of the Original and the Tested Value

Average Error Percentage: 0.2412 %

Sensex

Date	Actual	Predicted	Error
7 August 2017	32273.67	32503.05	0.7107%
10 August 2017	31531.33	31527.98	0.0106%
15 August 2017	31770.89	31730.45	0.1273%
22 August 2017	31568.01	31463.10	0.3323%
28 August 2017	31646.46	31623.33	0.0730%

Comparison of the Original and the Tested Value

Average Error Percentage: 0.2508 %

Nifty 50

Date	Actual	Predicted	Error
7 August 2017	10057.40	10106.42	0.4874%
10 August 2017	9820.25	9779.56	0.4142%
15 August 2017	9897.30	9917.78	0.2069%
22 August 2017	9852.50	9839.26	0.1343%
28 August 2017	9884.40	9890.46	0.0613%

Comparison of the Original and the Tested Value

Average Error Percentage: 0.2608 %

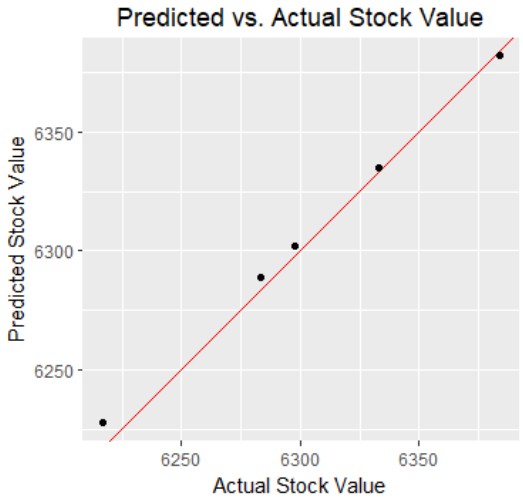
Test Results

As it can be seen from the above tables there are quite a few deviations, this is because we have taken only few factors which are affecting the stock price of the company. However, there are many other factors which affect the stock market they can be news releases on earnings and profits, and future estimated earnings, announcement of dividends, introduction of a new product or a product recall, securing a new large contract, employee layoffs, anticipated takeover or merger, a change of management, accounting errors or scandals, and the amount of stocks bought in a particular day.

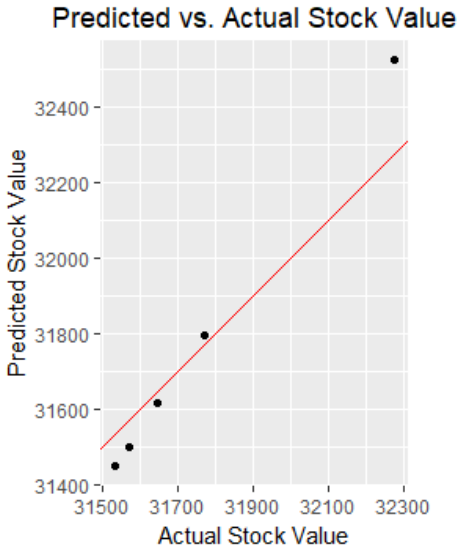
Results and Discussion

Output

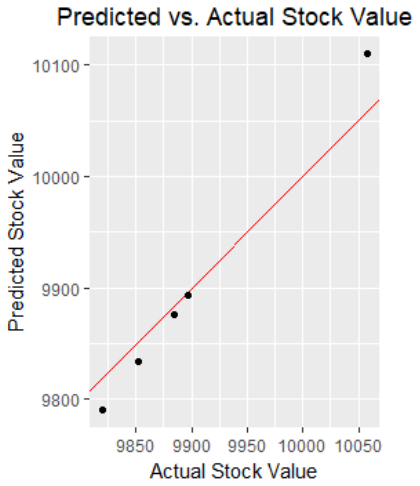
Multi-Linear Regression



Nasdaq Graph

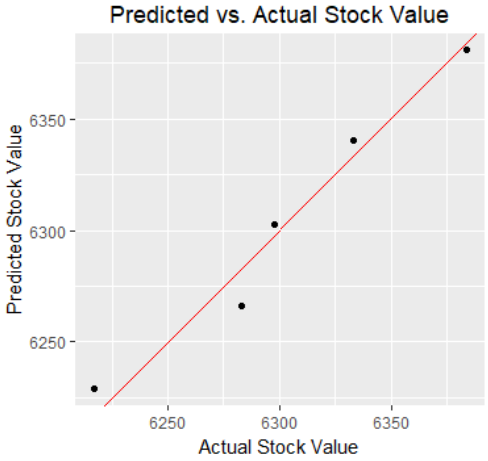


Sensex Graph

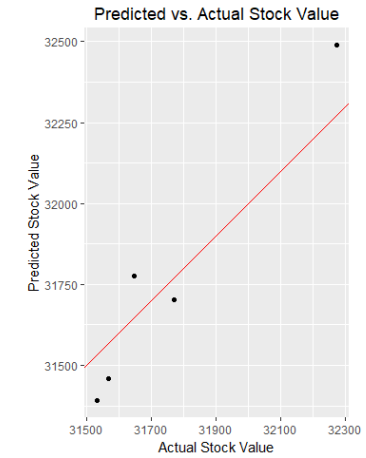


Nifty 50 Graph

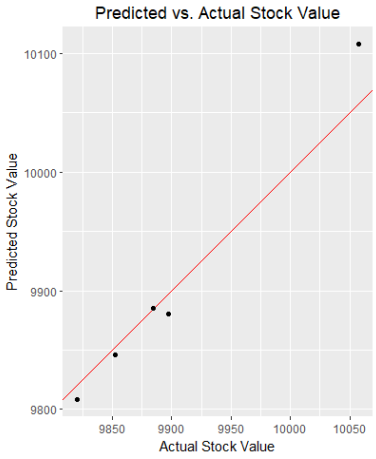
Neural Network



Nasdaq Graph



Sensex Graph



Nifty 50 Graph

Result Analysis

Multi-Linear Regression

After performing Backward Elimination, it is visible that the model is highly accurate. In the graph, nearer the dot is to the line, y=x, more efficient is the model. For all three companies, average error percentage is calculated which is less than 1%. This can be improved with addition of more independent variables.

Neural Network

In the Neural Network Graph, we can see that the dots are far from the line compared to the ones in Multiple Regression. Also the average error percentage is higher. Thus we can conclude that Multiple Regression is more efficient than Neural Network in this case.

There are many other factors which affect the stock market they can be news releases on earnings and profits, and future estimated earnings, announcement of dividends, introduction of a new product or a product recall, securing a new large contract, employee layoffs, anticipated takeover or merger, a change of management, accounting errors or scandals, and the amount of stocks bought in a particular day.

Discussion

Stock Market prediction is not an easy job, there are many factors which affect the stock prices. Since, we have not taken those factors in account we have difference in the values. But, there is a pattern in our model when the stock prices range from 2 figure value to 4 figure value, our residue is not huge. But when the value changes to 5 figure value, then we there is a presence of a significant amount of residue this is because the equation we have is that, the regression coefficient is multiplied by the date and for those huge numbers our regression coefficient does not have a huge value hence our residue is significant.

APPENDIX A

Nasdaq Data Set

	A	B	C	D	E	F	G
1	Date	Open	High	Low	Close	Volume	Price
2	01/08/2017	6372.16	6375.75	6345.75	6362.94	1812590000	6362.94
3	02/08/2017	6393.1	6394.21	6313.43	6362.65	2104130000	6362.65
4	03/08/2017	6366.24	6368.53	6331.14	6340.34	2113440000	6340.34
5	04/08/2017	6350.79	6361.49	6329.73	6351.56	1907170000	6351.56
6	07/08/2017	6361.06	6386.03	6356.23	6383.77	1701550000	6383.77
7	08/08/2017	6373.33	6423.35	6355.88	6370.46	1914670000	6370.46
8	09/08/2017	6322.92	6355.04	6309.44	6352.33	2033760000	6352.33
9	10/08/2017	6312.65	6318.28	6214.41	6216.87	2204700000	6216.87
10	11/08/2017	6222.18	6266.89	6216.19	6256.56	1794850000	6256.56
11	14/08/2017	6306.11	6346.83	6305.55	6340.23	1698560000	6340.23
12	15/08/2017	6350.51	6350.74	6324.75	6333.01	1580060000	6333.01
13	16/08/2017	6348.11	6374.56	6330.27	6345.11	1804730000	6345.11
14	17/08/2017	6322.72	6334.23	6221.91	6221.91	2041220000	6221.91
15	18/08/2017	6222.46	6254.22	6193.38	6216.53	1964590000	6216.53
16	21/08/2017	6216.32	6226.93	6177.19	6213.13	1576070000	6213.13
17	22/08/2017	6241.21	6302.84	6241.21	6297.48	1597700000	6297.48
18	23/08/2017	6263.47	6291.3	6263.29	6278.41	1533220000	6278.41
19	24/08/2017	6294.82	6302.85	6244.57	6271.33	1616780000	6271.33
20	25/08/2017	6293.81	6308.72	6257.1	6265.64	1442900000	6265.64
21	28/08/2017	6286.01	6292.26	6267.85	6283.02	1558690000	6283.02
22	29/08/2017	6228.9	6311.26	6228.73	6301.89	1637640000	6301.89
23	30/08/2017	6308.68	6374.47	6303.57	6368.31	1712070000	6368.31

Appendix A.1: Nasdaq Dataset

Sensex Dataset

	A	B	C	D	E	F	G	H
1	Date	Open	High	Low	Close	Volume	Price	
2	01/08/2017	32579.8	32632.02	32462.25	32575.17	9000	32575.17	
3	02/08/2017	32641.58	32686.48	32394.89	32476.74	10500	32476.74	
4	03/08/2017	32502.55	32502.55	32194.58	32237.88	12600	32237.88	
5	04/08/2017	32191.12	32352.19	32107.99	32325.41	11300	32325.41	
6	07/08/2017	32377.8	32396.14	32235.82	32273.67	10400	32273.67	
7	08/08/2017	32341.05	32354.77	31915.2	32014.19	9400	32014.19	
8	09/08/2017	31926.14	31967.28	31731.91	31797.84	9300	31797.84	
9	10/08/2017	31750.73	31756.27	31422.8	31531.33	13000	31531.33	
10	11/08/2017	31355.92	31379.2	31128.02	31213.59	15200	31213.59	
11	14/08/2017	31299.52	31526.4	31298.9	31449.03	9700	31449.03	
12	16/08/2017	31566.24	31805.99	31399.35	31770.89	12500	31770.89	
13	17/08/2017	31919.17	31937.51	31714.1	31795.46	8500	31795.46	
14	18/08/2017	31729.88	31729.88	31349.13	31524.68	16400	31524.68	
15	21/08/2017	31609.93	31641.81	31220.53	31258.85	23600	31258.85	
16	22/08/2017	31393.93	31484.28	31241.5	31291.85	10800	31291.85	
17	23/08/2017	31407.47	31593.39	31379.25	31568.01	10000	31568.01	
18	24/08/2017	31673.44	31678.19	31546.05	31596.06	8900	31596.06	
19	28/08/2017	31756.87	31809.7	31701.67	31750.82	10100	31750.82	
20	29/08/2017	31724.84	31739.8	31360.81	31388.39	6800	31388.39	
21	30/08/2017	31534.57	31727.98	31533.02	31646.46	5400	31646.46	
22	31/08/2017	31685.44	31757.18	31551.85	31730.49	17900	31730.49	
23								

Appendix A.2: Sensex Dataset

Nifty 50 Dataset

	A	B	C	D	E	F	G	H
1	Date	Open	High	Low	Close	Volume	Price	
2	01/08/2017	10101.05	10128.6	10065.75	10114.65	184300	10114.65	
3	02/08/2017	10136.3	10137.85	10054.2	10081.5	161500	10081.5	
4	03/08/2017	10081.15	10081.15	9998.25	10013.65	192700	10013.65	
5	04/08/2017	10008.6	10075.25	9988.35	10066.4	178600	10066.4	
6	07/08/2017	10074.8	10088.1	10046.35	10057.4	137400	10057.4	
7	08/08/2017	10068.35	10083.8	9947	9978.55	203300	9978.55	
8	09/08/2017	9961.15	9969.8	9893.05	9908.05	169200	9908.05	
9	10/08/2017	9872.85	9892.65	9776.2	9820.25	235500	9820.25	
10	11/08/2017	9712.15	9771.65	9685.55	9710.8	285900	9710.8	
11	14/08/2017	9755.75	9818.3	9752.1	9794.15	195600	9794.15	
12	16/08/2017	9825.85	9903.95	9773.85	9897.3	219500	9897.3	
13	17/08/2017	9945.55	9947.8	9883.75	9904.15	197600	9904.15	
14	18/08/2017	9865.95	9865.95	9783.65	9837.4	246300	9837.4	
15	21/08/2017	9864.25	9884.35	9740.1	9754.35	205300	9754.35	
16	22/08/2017	9815.75	9828.45	9752.6	9765.55	183600	9765.55	
17	23/08/2017	9803.05	9857.9	9786.75	9852.5	168600	9852.5	
18	24/08/2017	9881.2	9881.5	9848.85	9857.05	184700	9857.05	
19	28/08/2017	9907.15	9925.75	9882	9912.8	159600	9912.8	
20	29/08/2017	9886.4	9887.35	9783.75	9796.05	173300	9796.05	
21	30/08/2017	9859.5	9909.45	9850.8	9884.4	157800	9884.4	
22	31/08/2017	9905.7	9925.1	9856.95	9917.9	327700	9917.9	
23								

Appendix A.3: Nifty 50 Dataset