Stock Price Prediction: A Comparative Study between Traditional Statistical Approach and Machine Learning Approach

Indronil Bhattacharjee
Department of Computer Science and Engineering
Khulna University of Engineering & Technology
Khulna, Bangladesh
ibprince.2489@gmail.com

Pryonti Bhattacharja
Department of Economics
Shahjalal University of Science and Technology
Sylhet, Bangladesh
bhattacharjadipty@gmail.com

***Abstract*—Stock market is one of the most important sectors of a country's economy. Prediction of stock prices is not easy since it is not stationary in nature. The objective of this paper is to find the best possible method to predict the closing prices of stocks through a comparative study between different traditional statistical approaches and machine learning techniques. Predictions using statistical methods like Simple Moving Average, Weighted Moving Average, Exponential Smoothing, Naive approach, and machine learning methods like Linear Regression, Lasso, Ridge, K-Nearest Neighbors, Support Vector Machine, Random Forest, Single Layer Perceptron, Multi-layer Perceptron, Long Short Term Memory are performed. Moreover, a comparative study between statistical approaches and machine learning approaches has been done in terms of prediction performances and accuracy. After studying all the methods individually, the machine learning approach, especially the neural network models are found to be the most accurate for stock price prediction.**

Keywords—Stock price prediction, Machine Learning, Statistical methods, Neural Networks, Multi-layer Perceptron, Long Short Term memory.

I. INTRODUCTION

Stock market is an aggregation stockbrokers and traders who can buy and sell shares of stocks. Many large companies have their stocks listed on a stock market. This makes the stock liquid and thus more attractive to the investors [1]. There is a large number of investors who invest handsome amounts in a stock market. However, it involves risk since prices of stock may rise or fall within no time [2]. That is why predicting stock prices is not an easy task and many researchers are working on it.

Stock price predictions have been performed using different approaches. Some statistical approaches like Simple Moving Average (SMA), Weighted Moving Average (WMA), Exponential Smoothing, and Naive Approach were used traditionally to predict stock prices in the earlier days. Since statistical approaches are linear in nature, it hampers prediction performances in case of sudden rise or fall of prices of the stocks. As stock data is non-stationary, chaotic, random and depends on several technical parameters, statistical approaches are not found to be so accurate [3].

In modern days of artificial intelligence, machine learning plays an important role in time series predictions. Machine learning algorithms are found to have better predictability in stock price prediction. Some promising techniques like Simple Linear Regression, Lasso, Ridge, Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Random Forest and neural network models like Single Layer Perceptron (SLP), Multi-Layer Perceptron (MLP), Long Short Term Memory (LSTM) are used for stock price prediction in this paper.

The focus of this paper is to make a comparison in terms of performances between the traditional statistical methods and machine learning techniques. Moreover, finding a better approach, which predicts prices of the stocks more accurately and reduces error in prediction.

II. METHODOLOGY

A. Statistical Methods

1) Simple Moving Average (SMA): In this method, an unweighted mean of a specific number of previous data is considered to be the predicted value for the next day [1]. The formula used for SMA is as follows-

$$F_t = \frac{A_{t-1} + A_{t-2} + A_{t-3} + \cdots + A_{t-n}}{100} \quad (1)$$

2) Weighted Moving Average (WMA): The difference between SMA and WMA is that a weight is used with the previous values to predict the future value. The formula used for WMA is as follows-

$$F_t = \frac{A_{t-1}w_{t-1} + A_{t-2}w_{t-1} + \ldots + A_{t-n}w_{t-1}}{100} \quad (2)$$

Here,

$$w_{t-i} = W * (n - i) \quad (3)$$

$$W = \frac{100}{1 + 2 + 3 + \ldots + n} \quad (4)$$

978-1-7281-6040-5/19/$31.00 ©2019 IEEE

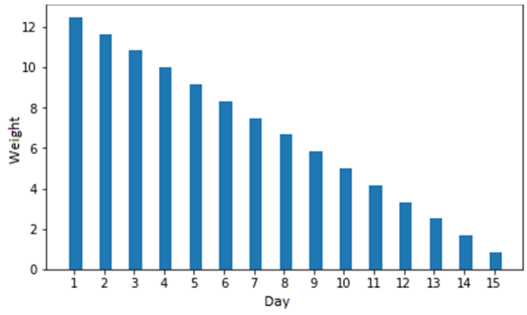


Fig. 1. Weights for i[th] last day of 15-day WMA

3) Exponential Smoothing: An smoothing constant, α is used for smoothing the prediction value from the previous prediction in Exponential Smoothing method [3]. This smoothing constant maximizes prediction accuracy from the last prediction. The formula used for exponential smoothing is as follows-

$$F_t = A_{t-1} + \alpha * (A_{t-1} - F_{t-1}) \quad (5)$$

4) Naïve Approach: Naive approach is often called Last Value method. That is, the last actual value is considered as the predicted value for the next day. The formula used for Naive approach is as follows-

$$F_t = A_{t-1} \quad (6)$$

In equation 1-6,

F_i = Predicted closing price for i[th] day
A_i = Actual closing price at i[th] day
n = Number of days considered for prediction
w_i = Weight used for i[th] day
W = Unit weight
α = Smoothing constant

B. Machine Learning Methods

1) Simple Linear Regression: Simple linear regression algorithm is one of the fundamental supervised machine learning algorithms used for regression.

2) Ridge Regression: Ridge is one of the techniques which reduces model complexity and prevents over-fitting. In ridge regression, the coefficients are shrunk and this reduces the complexity of the model from Simple Linear Regression [6].

3) Lasso Regression: Lasso stands for Least Absolute Shrinkage and Selection Operator [6]. Lasso finds the central point where data values are shrunk.

4) K-Nearest Neighbor (KNN): K-Nearest Neighbors is a simple algorithm to predict numerical target based on a similarity measure. KNN calculates the average of the numerical target of the K nearest neighbors [7]. Distance between points is measured using the following formula-

$$D(x_i, y_i) = \left(\sum_{i=1}^{n} |x_i - y_i|^{q} \right)^{\frac{1}{q}} \quad (7)$$

5) Random Forest: Random Forest algorithm is one of the ensemble learning algorithms [8]. It is an additive model that makes prediction by combining decisions of a sequence of decision trees. Random forest prediction formula is as follows-

$$g(x) = f_0(x) + f_1(x) + \cdots + f_n(x) \quad (8)$$

In (8),

g(x) = Final prediction
f_i (x) = Decision function of i[th] decision tree

6) Support Vector Machine (SVM): SVM can be used as regression method. Support Vector Regression uses the same principles as the SVM for classification, with only a few minor differences. A margin of tolerance (ϵ) is set in approximation to the SVM [9].

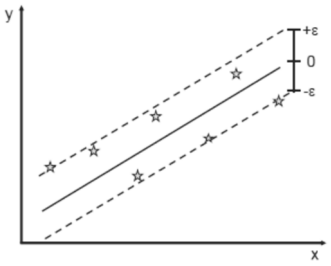
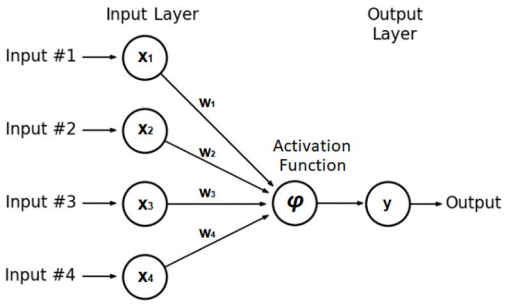


Fig. 2. Support Vector Machine

7) Single Layer Perceptron (SLP): Single Layer Perceptron model is a neural network model which consists of only one input layer and one output layer. There will be no hidden neuron layer in between.

Fig. 3. Single Layer Perceptron Model



Weight update rule for each iteration:

$$w_i(t+1) = w_i(t) + \Delta w_i(t) \quad (9)$$

In (9),

w_i (t+1) = Updated weight for (t+1)[th] iteration
w_i (t) = Weight used for t[th] iteration
Δw_i (t) = Change of weight in t[th] iteration

Now, change of weight in t[th] iteration,

$$\Delta w_i(t) = \alpha * x_i(t) + e(t) \quad (10)$$

In (10),

α = Learning rate
x_i (t) = i[th] input for t[th] iteration
e (t) = Error in t[th] iteration

8) Multi-Layer Perceptron (MLP): There will be one or more hidden layers present in between one input layer and one output layer in an MLP model.

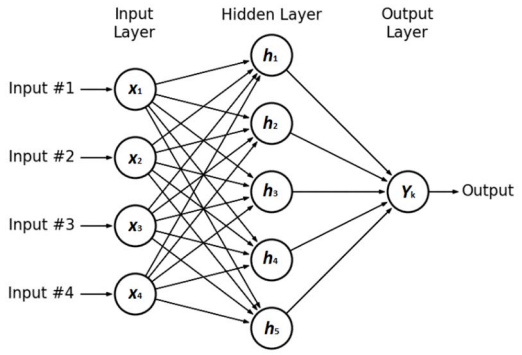


Fig. 4. Multilayer Perceptron Model

Output calculation formula for output layer is as follows –

$$y_k(t) = \varphi\left[\sum_{j=1}^{n} x_{jk}(t) * y_{jk}(t) - \theta_k\right] \quad (11)$$

In (11),

y_k = Final output
φ = Activation function
θ_k = Threshold
n = Number of neurons

9) Long Short Term Memory (LSTM): LSTM is associated with Recurrent Neural Network. It introduces memory unit, forget gate and update gate with simple RNN [10].

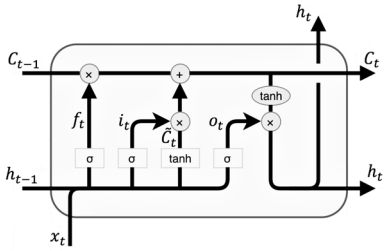


Fig. 5. Long Short Term memory

C. Evaluation Measures

To evaluate the performance of these prediction models, Mean Squared Error (MSE) and Mean Absolute Percentage Error (MAPE) are calculated. The formulae of calculating these evaluation measures are as follows-

1) Mean Squared Error (MSE):

$$MSE = \frac{\sum_{t=1}^{n}(A_t - F_t)^2}{n} \quad (12)$$

2) Mean Absolute Percentage Error (MAPE):

$$MAPE = \left(\frac{100}{n}\right) * \left|\frac{A_t - F_t}{A_t}\right| \quad (13)$$

In (12) and (13),

F_t = the predicted closing price for t^{th} day
A_t = the actual closing price at t^{th} day
n = the number of days predicted

III. The Proposed System

A. Dataset

Stock market data of (a) Tesla from 29-06-2010 to 17-03-2017 and (b) Apple from 02-01-2014 to 31-12-2018 have been used as dataset in this system. The datasets have (a) 1693 rows and 6 columns, (b) 1259 rows and 6 columns. Each row represents the information of a single day. In case of columns-

1st column - the date,
2nd column - the opening price of that day,
3rd column - the highest price of that day,
4th column - the lowest price of that day,
5th column - the closing price of that day and
6th column - the volume of shares traded on that day.

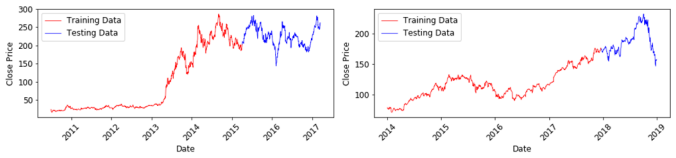
B. Data Preprocessing

Data preprocessing includes checking out for missing values and discards those data from the data set, looking for categorical values and drop out unnecessary information in the data set.

C. Data Splitting

The dataset has been split into two parts as training data and test data. Here, 1200 data has been considered as training data and rest 492 data has been kept for testing.

(a) Training Data (1200 days): 29-06-2010 to 06-04-2015
Testing Data (492 days): 07-04-2015 to 17-03-2017
(b) Training Data (1000 days): 02-01-2014 to 19-12-2017
Testing Data (258 days): 20-12-2017 to 31-12-2018



(a)Tesla Dataset (b) Apple Dataset

Fig. 6. Training data and Testing data

D. Scaling

All the data are scaled with standard scaler to limit the ranges of the variables. By scaling of data, those can be compared on common environments in case of all the methods.

E. Feature Selection

For predicting future values, selection of features is an important task. Selecting best features can make the prediction performance higher, whereas, selection of worst features can direct the prediction to a wrong way. In this system, three features have been selected for closing price prediction. These are the opening price, the highest price and the lowest price.

F. Prediction

Prediction has been performed using both traditional statistical and machine learning approaches. In case of statistical prediction, training of model is not necessary. This

approach predicts data by using statistical mean of previous closing prices. We have performed four such methods.

- Prediction using Simple Moving Average method is done for 10-day, 15-day and 30-day as previous data consideration.
- Weighted Moving Average method is also performed for 10-day, 15-day and 30-day period. But in this method, weight has been calculated accordingly using (3) and (4) which is multiplied with the data to predict prices.
- Exponential Smoothing varies with the change of smoothing constant (α). Prediction has been performed using α = 0.75, 0.5, 0.25, and the most accurate one is selected.
- Naïve approach predicts the prices using the previous day's closing price. This method is applied here also, but sudden rise or fall of price has made some difficulties for this method to predict.

In case of machine learning approaches, the models are trained first using the training data and then performs prediction on the testing data. Nine machine learning methods are performed.

- Simple Linear Regression method is applied for closing price prediction using opening price, highest price and lowest price as training attributes. Moreover, Lasso and Ridge regressions are also used to minimize error and increase prediction accuracy.
- K-Nearest Neighbor algorithm has been used with algorithm = auto and leaf_size = 30 for prediction. Scaled values are used as training attributes.
- Prediction has been performed using Random Forest algorithm for several number of estimators. Values of n are 100, 250, 500 and 100 respectively.
- Support Vector Machine has been used as regressor for price prediction. Kernel = Linear, Gamma = Auto Depreciated are used as model attributes.
- SLP model is used with optimizer = adam and loss = mean_squared_error, Batch_size = 50 and epoch = 500. Number of neurons for input layer is 16. Prediction value has been calculated using (11).
- MLP model is used where number of hidden layers are 2, 3 and 5 respectively. Number of neurons for input layer is 25 and number of neurons in hidden layers are 75 and 100. Activation function used was relu. Optimizer = adam and loss = mean_squared_error, Batch_size = 50 and epoch = 500.
- Finally, LSTM has been used as two hidden layers in the neural network model. Number of neuron used is 100 and return_sequences = True. Optimizer = RMSProp and loss = mean_squared_error, Batch_size = 100 and epoch = 50.

G. *Error Calculation:*

MSE and MAPE has been calculated for all the methods applied using (12) and (13). Performance evaluation is done from MSE and MAPE values of the models.

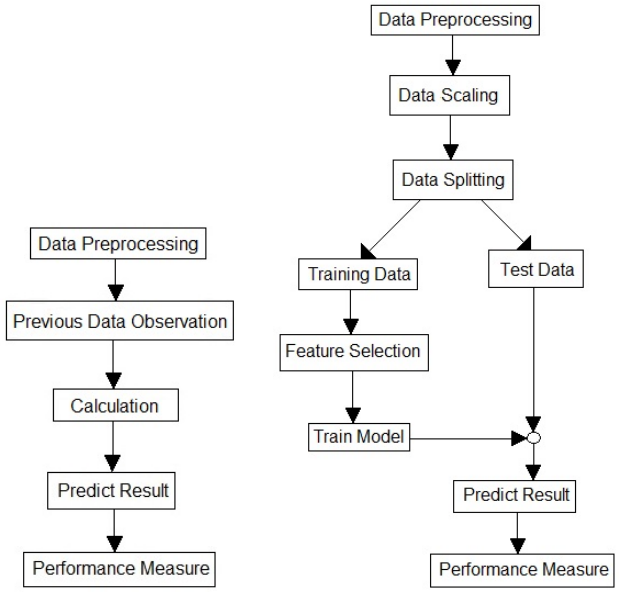


Fig. 7. Flow Diagram of the system
(a) Statistical Methods and (b) Machine Learning Methods

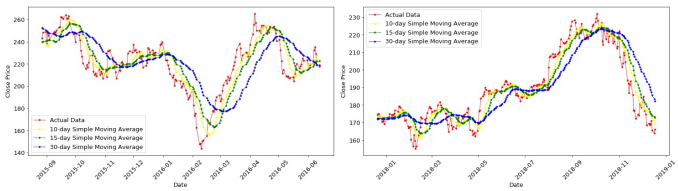
IV. RESULTS

Performance measures and predictions using different prediction methods are illustrated in Table I-VIII and Fig. 8-15.

A. *Simple Moving Average (SMA)*

TABLE I. PERFORMANCE MEASURES OF DIFFERENT SIMPLE MOVING AVERAGE METHODS

SMA	Errors for Tesla Dataset		Errors for Apple Dataset	
	MSE_T	$MAPE_T$	MSE_A	$MAPE_A$
30-Day SMA	322.4261	6.4752	125.42905	4.6513
15-Day SMA	141.0808	4.3776	53.5439	3.1054
10-Day SMA	87.1748	3.3761	32.6763	2.3601



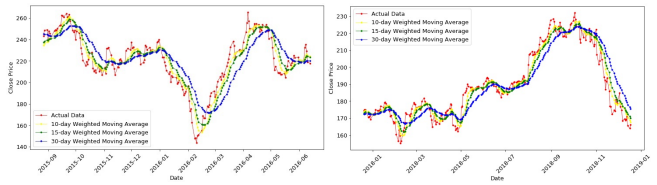
(a) Tesla Dataset (b) Apple Dataset

Fig. 8. Prediction using different simple moving averages

B. Weighted Moving Average (WMA)

TABLE II. PERFORMANCE MEASURES OF DIFFERENT WEIGHTED MOVING AVERAGE METHODS

WMA	Unit Weight W	Errors for Tesla Dataset		Errors for Apple Dataset	
		MSE_T	$MAPE_T$	MSE_A	$MAPE_A$
30-Day WMA	0.22	184.6105	4.9482	69.8537	3.5392
15-Day WMA	0.83	81.1320	3.2736	30.3559	2.2954
10-Day WMA	1.82	50.4788	2.8051	18.3376	1.7408



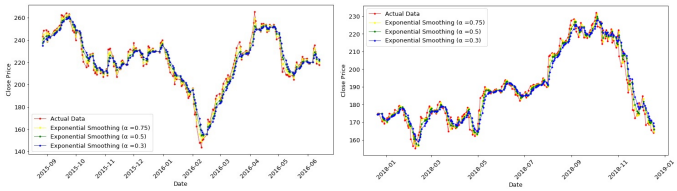
(a) Tesla Dataset (b) Apple Dataset

Fig. 9. Prediction using different weighted moving averages

C. Exponential Smoothing

TABLE III. PERFORMANCE MEASURES OF EXPONENTIAL SMOOTHING METHOD WITH DIFFERENT SMOOTHING CONSTANTS

Smoothing Constant α	Errors for Tesla Dataset		Errors for Apple Dataset	
	MSE_T	$MAPE_T$	MSE_A	$MAPE_A$
0.3	60.5101	2.7077	22.3635	1.9181
0.5	40.9085	2.1394	15.0884	1.5516
0.75	31.8758	1.8353	12.1082	1.3517



(a) Tesla Dataset (b) Apple Dataset

Fig. 10. Prediction using exponential smoothing with different smoothing constants

D. Naïve Approach

TABLE IV. PERFORMANCE MEASURES OF NAÏVE APPROACH METHOD

Method	Errors for Tesla Dataset		Errors for Apple Dataset	
	MSE_T	$MAPE_T$	MSE_A	$MAPE_A$
Naïve Approach	28.8352	1.7313	11.2314	1.2905

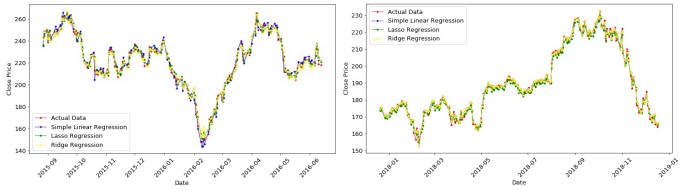
(a) Tesla Dataset (b) Apple Dataset

Fig. 11. Prediction using Naïve approach

E. Simple Linear Regression, Lasso and Ridge

TABLE V. PERFORMANCE MEASURES OF DIFFERENT REGRESSION METHODS

Regression Method	Errors for Tesla Dataset		Errors for Apple Dataset	
	MSE_T	$MAPE_T$	MSE_A	$MAPE_A$
Simple Linear Regression	10.2627	1.1311	7.4275	1.0987
Lasso Regression	10.1814	1.1287	7.4111	1.0978
Ridge Regression	9.3313	1.1045	7.2778	1.0512



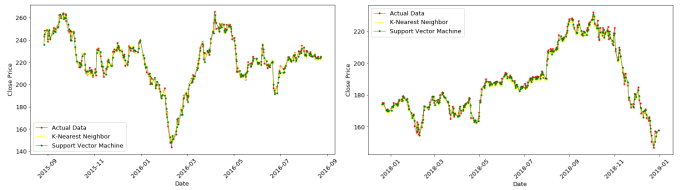
(a) Tesla Dataset (b) Apple Dataset

Fig. 12. Prediction using Simple Linear, Lasso and Ridge Regressions

F. K-Nearest Neighbors and Support Vector Machine

TABLE VI. PERFORMANCE MEASURES OF KNN AND SVM

SMA	Errors for Tesla Dataset		Errors for Apple Dataset	
	MSE_T	$MAPE_T$	MSE_A	$MAPE_A$
K-Nearest Neighbors	6.6241	0.8869	6.6314	0.9423
Support Vector Machine	8.3624	0.9947	4.1864	0.8111



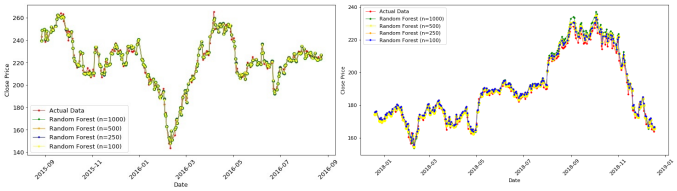
(a) Tesla Dataset (b) Apple Dataset

Fig. 13. Prediction using KNN and SVM

G. Random Forest

TABLE VII. PERFORMANCE MEASURES OF RANDOM FOREST WITH DIFFERENT NUMBER OF ESTIMATORS

Number of estimators	Errors for Tesla Dataset		Errors for Apple Dataset	
	MSE_T	$MAPE_T$	MSE_A	$MAPE_A$
100	7.1052	0.9131	6.5189	0.9785
250	6.9909	0.9129	6.5121	0.9778
500	7.0017	0.9144	6.0489	0.9433
1000	6.9326	0.9119	6.6759	0.9827



(a) Tesla Dataset (b) Apple Dataset

Fig. 14. Prediction using random forest with different number of estimators

H. Neural Network Models

TABLE VIII. PERFORMANCE MEASURES OF DIFFERENT SIMPLE MOVING AVERAGE METHODS

Neural Network Model	Number of hidden layers	Errors for Tesla Dataset		Errors for Apple Dataset	
		MSE_T	$MAPE_T$	MSE_A	$MAPE_A$
Single Layer Perceptron	0	8.4909	1.0094	5.2322	0.8753
Multi-Layer Perceptron	2	3.9911	0.7112	5.2214	0.8694
	3	3.7941	0.7020	5.0113	0.8393
	5	5.6965	0.8616	4.8978	0.7907
Long Short Term Memory	5	3.7708	0.6993	3.5248	0.7013

(a) Tesla Dataset (b) Apple Dataset

Fig. 15. Performance Measures of Different Neural Network Models

Predictions has been performed and performance measures have been calculated in terms of Mean Squared Error and Mean Absolute Percentage Error for all the applied methods.

It is observed that MSE values for statistical methods ranges within 322.4 to 28.8 approximately for Tesla dataset and 125.4 to 11.2 approximately for Apple dataset, MAPE values ranges within 6.5 to 1.7 approximately for Tesla dataset and 4.7 to 1.3 approximately for Apple dataset. On the contrary, that MSE values for machine learning methods ranges within 10.3 to 3.8 approximately for Tesla dataset and 7.4 to 3.5 approximately for Apple dataset, MAPE values ranges within 1.13 to 0.69 approximately for Tesla dataset and 1.09 to 0.7 approximately for Apple dataset. This shows a huge difference between average errors of statistical methods and machine learning methods, which proves that machine learning algorithms are more appropriate than traditional statistical methods.

V. CONCLUSION

A comparative study between statistical approaches and machine learning approaches has been done in terms of prediction performances and accuracy. After studying all the methods individually, machine learning methods, especially, MLP and LSTM are found to be the most accurate to predict stock prices for having the least MSE and MAPE values.

REFERENCES

[1] R. S. Dhankar, Capital Markets and Investment Decision Making, 1st ed. Springer India, 2019, ch. Stock Market Operations and Long-Run Reversal Effect.

[2] M. Usmani, S. H. Adil, K. Raza, and S. S. A. Ali, "Stock market prediction using machine learning techniques," in 2016 3rd International Conference on Computer and Information Sciences (ICCOINS), Aug 2016, pp. 322–327.

[3] H. Grigoryan, "A stock market prediction method based on support vector machines (svm) and independent component analysis (ica)," Database Systems Journal, vol. 7, no. 1, pp. 12–21, 2016.

[4] S. Hansun, "A new approach of moving average method in time series analysis," in 2013 International Conference on New Media Studies, CoNMedia 2013, Nov 2013, pp. 1–4.

[5] E. Ostertagova and O. Ostertag, "The simple exponential smoothing model," 09 2011.

[6] Saptashwa, "Ridge and Lasso Regression: A Complete Guide with Python Scikit-Learn,"https://towardsdatascience.com/ridge-and-lassoregression-a-complete-guide-with-python-scikit-learn-e20e34bcbf0b, Sep 26, 2018.

[7] D. S. Sayad, "K Nearest Neighbors - Regression," http://saedsayad.com/k nearest neighbors reg.htm.

[8] J. Patel, S. Shah, P. Thakkar, and K. Kotecha, "Predicting stock market index using fusion of machine learning techniques," Expert Syst. Appl., vol. 42, pp. 2162–2172, 2015.

[9] D. S. Sayad, "Support Vector Machine - Regression (SVR)," https://www.saedsayad.com/support vector machine reg.htm.

[10] S. Rathor, "Simple RNN vs GRU vs LSTM :- Difference lies in More Flexible control," https://medium.com/@saurabh.rathor092/simple-rnnvs-gru-vs-lstm-difference-lies-in-more-flexible-control-5f33e07b1e57, Jun 2, 2018