International Journal of Recent Technology and Engineering (IJRTE)
ISSN: 2277-3878 (Online), Volume-11 Issue-2, July 2022

Stock Market Analysis and Prediction using Machine Learning

Amaan Aijaz, Komal Rastogi, TK SivaKumar

Abstract: ***The stock market is a field which has spurred the interest of not only researchers, but, ordinary people as well over the years. It has encouraged scientists to develop better predictive models. The stock market is not as simple as it might seem initially. It is a transformative, non-straight dynamical and complex system. The main objectives of the project are to analyze the advantages and disadvantages of using Machine Learning techniques for the purpose of predicting values and comparing different algorithms, along with integrating the best model in a Web Application which will help users to be able to improve their decision making strategies by trying to give them an insight on what could possibly happen in the short term of things. Machine learning can be described as a technology that enables a system to learn on its own through real-world interactions with the help of data and hence be able to recognize similar interactions which the machine initially learned. Machine Learning over the past decade has quite subtly become to become a part of one's everyday life due to its key role in numerous vital applications.***

Keywords: ARIMA, Linear Regression, Machine Learning, Stock Market, Support Vector Machine, Web Application.

I. INTRODUCTION

Given the current scenario of the planet combined with the various success stories of the high returns possessed by people from the stock market, the stock market has been host to an exponential rise in shareholders, both experienced as well as novice. Looking at the high returns might seem enticing, combining the high risks with the high returns is the reality of the stock market. The stock market is not as simple as it might seem initially. It is a transformative, non-straight dynamical and complex system. Long term investment is one of the major investment decisions. It is seen that Machine Learning could be used to guide an investor's decisions. With the current Covid scenario, the Stock Market has seen an exponential rise in the number of shareholders, due to the time at hand of individuals as well as the fact that the stock market is considered one of the easier to methods to gain profits, given the fact that every day more than a billion dollars are traded in the stock market, and behind every trade is someone hoping to either make a profit or to avoid a loss.

Manuscript received on 05 June 2022 | Revised Manuscript received on 17 June 2022 | Manuscript Accepted on 15 July 2022 | Manuscript published on 30 July 2022.

***** Correspondence Author

Amaan Aijaz*, Department of Computing Technologies, SRM Institute of Science and Technology, Chennai (Tamil Nadu.), India. Email: amaanaijaz2000@gmail.com

Komal Rastogi, Department of Computing Technologies, SRM Institute of Science and Technology, Chennai (Tamil Nadu.), India. Email: kr9601@srmist.edu.in

Dr. TK Sivakumar, Department of Computing Technologies, SRM Institute of Science and Technology, Chennai (Tamil Nadu.), India. Email: sivakumt2@srmist.edu.in

Given the dynamic nature of the stock market that relies on numerous parameters, it would be impossible for normal people to take in to account all these parameters and try to predict which stocks will be fruitful. Given the current advancement of Artificial Intelligence, it would seem a better approach to automate the process of looking at all these parameters so as to not only provide a more fruitful insight in to stocks as well as provide a platform for everyone to be able to make profits. Through this project, we analyze different kinds of ML methods to forecast stock price trend.

II. LITERATURE SURVEY

1. Stock Price Predictions with ML Using Python [1][2]

The studies that primarily focus on the stock market price prediction using a number of Machine Learning algorithms and then following it up with LTSM to compare and contrast the working of the various algorithms. The articles mainly focus on ML algorithms like Linear Regression and Moving Averages formed the core of the study. The study focuses on training the machine from data points spread out in the past to predict the future. Machine Learning libraries like Numpy, Scipy, SkLearn were at the core of the technological front. Numpy was used to clean and manipulate data to get it ready for analysis while sci-kit was used for visualizations, analysis and prediction.

2. Stock Market Prediction using Support Vector Machines [3][4]

This study primarily uses Support Vector Machines to make stock market predictions. SVMs offer an alternative method for improving stock market prediction accuracy. The technique uses supervised learning. The survey in these set of articles is performed using R-Studio. The observations are viewed by plotting them against each other. The data is obtained by Quandl which provides a R API package to fetch the stock prices of companies. In these sets of articles actual predictions aren't performed, but, rather proof of how well LR & SVM fits the data using training and testing is done. The performance measures that were used to assess the accuracy of the models include RMSE, MAE, MSE.

3. Stock Market Forecasting Technique using ARIMA Model [5][6]

This study primarily uses ARIMA model to make stock market predictions. ARIMA offer an alternative method for improving stock market prediction accuracy. The articles involved prediction of the movement of Nifty 50 index of the Indian Stock Market. The analysis involves monthly information on the closing stock indices of NIFTY covering a period from 2014-2018, having a total of 1222 observations.

Data Decomposition is the main factor for the analysis. After decomposing the data, it was verified whether the data was stationary or not using the ADF (Augmented Dicky Fuller Test). Following the ADF, an autocorrelation and partial autocorrelation graph was plotted followed by the discussion and implementation of ARIMA model. The analysis concluded by showcasing how close the predictions made by the ARIMA model were, to that of the actual price, which indicates the successful implementation of the ARIMA model on time series data.

III. PROPOSED WORK

In this proposed system, we center our system around analyzing the different machine learning algorithms and try forecasting stock values with each model so as to try to reach a conclusion wherein we will have obtained the better model for stock forecasting. We look into a number of machine learning models namely Support Vector Machine Regression, Linear Regression, Naive Model, Moving Average Model and Auto Regressive Integrated Moving Average Model (ARIMA). The system is completely python based due to the reason that python provides numerous libraries to aid the purpose of Machine Learning models. In this system, we used real time data obtained from python library 'YFinance', which provided us with various data points from the past timeline to make a prediction on the future price of the stock. The other libraries used include that of Pandas, Numpy, Sci-py for the purpose of data cleaning and manipulation, Scikit Learn was used for the purpose of analysis as well as prediction, while Matplotlib and Seaborn were utilized for the purpose of creating visualizations. The dataset was obtained using the YFinance library of python, which was then pre-processed and split according to an 85-15 split in train test data, which is 85% data was used to train the machine while 15% was used to test it. Initially, the Linear Regression model was developed and analyzed followed by a Support Vector Machine Regression model. The below diagram, Figure 1, showcases an overall representation of the proposed system including all the models to be analyzed:

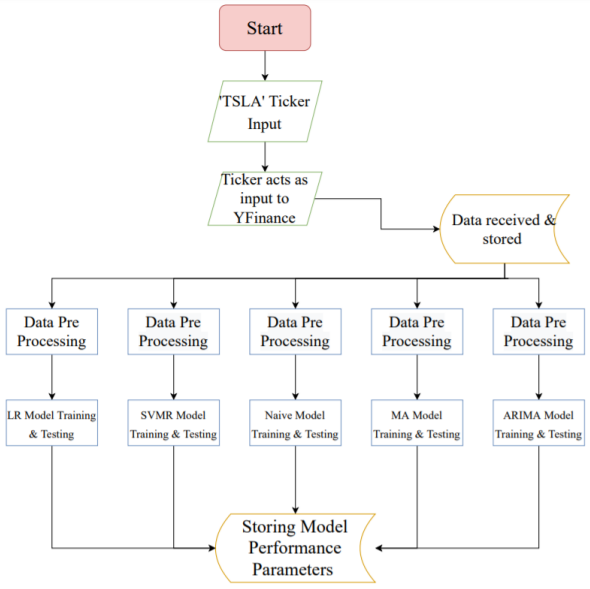

Figure 1: Block Diagram for Model Analysis

IV. METHODOLOGY

The basic methodology for a supervised learning model is its ability to identify and learn any particular pattern and relationships in the data making use of the training set and then, testing the framed hypothesis on the test set.

A. Environment

This analysis has been performed using Python, which is one of the most resourceful language for statistical analysis. Google Colaboratory and Jupyter Notebook were made use of to ensure individual records of each model be kept. Python provides a number of libraries for the purpose of training and testing Machine Learning models, like Keras and Scikit Learn, which are widely used across the Data Science fields.

B. Linear Regression

Regression is a statistical method that attempts to determine the robustness of the relationship between one dependent variable (denoted Y) and other independent variables. This method is one of the mostly widely used methods for the purpose of forecasting and determining the correlations amongst the variables. The general hypothesis function of Linear Regression is of the form,

$$y = a+bx \quad (1)$$

where, b is the regression coefficient, a is the regression intercept, y is the y-coordinate and x is the x-intercept [1] It can be noted that this equation is like that of a straight line. The main effort can be considered to be that of to try to get various possible values of a and b, and to try to acquire values which in turn provide the most appropriate fit through the data when mapped on the x-y plane.

C. Support Vector Machine Regression

Support Vector Machines is a type of Machine Learning algorithm that was initially developed as a Classification algorithm. The primary task of this algorithm is to identify an N-dimensional space that is distinguishably able to classify the numerous data points. The Support Vector Machine model is a representation of data points mapped in such a manner that the various data points are divided by a margin as large as possible. Based on the position of the data points relative to that of the decision boundary, the data points are classified accordingly. The dimensions of the hyperplane rely on the number of attributes.

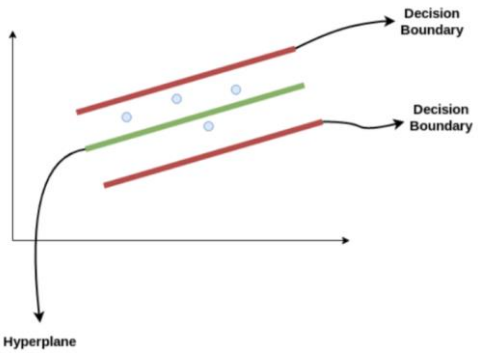

Figure 2: Working of Support Vector Machines

Retrieval Number: 100.1/ijrte.B70830711222
DOI: 10.35940/ijrte.B7083.0711222
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

A Support Vector Machine for Regression was proposed by Vladimir N. Vapnik, Harris Drucker, Christopher J. C. Burges, Linda Kaufman and Alexander J. Smola in 1996, where the model produced is dependent only on a subset of the training data. [2]

D. Naïve Forecasting

Such types of forecasting models run exclusively on the basis of historical data observations of variables, like sales, cash flow. Unlike the other models, this model doesn't make much of an attempt at providing an explanation of the underlying casual relationships between variables. A naive forecast is considered to be naive due to the fact that the input that the model takes is the price from the previous day and uses that price for the prediction of tomorrow. This is surprisingly effecting in this scenario due to the relatively due to autocorrelation. The price of today is what potentially predicts the price of the stock tomorrow. The naïve model for the purpose of analysis for this project is a simple projection model which requires inputs in the form of recent observations in the data in order to produce the output. The advantage of such a model is its inexpensiveness, be it in operating and maintaining the model, developing the model or storing the data. The disadvantage however is that mentioned earlier, wherein the model doesn't consider any possibility of some casual relationships between variables.

E. Moving Average Forecasting

When seen from a statistical perspective, a moving average is a mathematical calculation which helps in analyzing data points by the creation and calculation of average of various subsets of the data. A moving average model can be used to forecast future values and shouldn't be confused with moving average smoothing which is used for the purpose of estimating a tread cycle of the past historical values. Moving Averages can't be realistically considered as prediction models. When the phrases "smooth" or "de-trend" data are seen, they more often than not refer to the implementation of sort of moving average. There are numerous types of moving average models, but, if the two most prominent had to be pointed out, it would have to be that of Simple and Exponential. Simple moving average, as the name shall suggest, is the average value, in this case the stocks, over a desired time interval. Exponential moving average can be considered a bit more complex as it also takes into account a weight factor to each and every time step in the window. The weights allow the moving average to respond much more quickly to abrupt changes. For the purpose of this project a Simple Moving Average is taken in to account spread over two time periods, the two time periods being a five-day time period and a twenty-day time period. The five-day time period is taken into account as the stock market is open five days a week and the twenty-day time period is taken into account as the stock market is open for roughly twenty days a month.

F. Auto Regressive Integrated Moving Average Model

ARIMA is an acronym for Auto Regressive Integrated Moving Average. Over the past years, the interest in this sort of model has been over of the roof as it has been suggested to be a good machine learning fit to that of the neural networks, and that can be understood as neural networks have their own drawbacks, some being cost expensive, slow convergence rate, local maxima and minima, to name a few. Even though neural networks consist of drawbacks they were still preferred for time series data. Time series is basically a series, or to put it better, a sequence wherein a metric is recorded over the same intervals of time. Time series data can also be called time stamped data, as it is collected at different points spread across time. The data collected basically comprises of successive measurements that have been made from the source over a particular time interval. ARIMA forecasting is basically making use of the ARIMA forecasting algorithm which uses the information present in the historical data to make future predictions. It tries to provide an explanation for a given time series based on its own lags and lagged errors, which can be used to predict future values.

V. SYSTEM SETUP

A. Data Collection

Historical prices were retrieved by using the YFinance library in Python. YFinance allows us to fetch the stock prices of companies for any time range in a few lines of code. This automation proves to be of higher efficiency than other means of data collection. The library operates on the use of a stock 'ticker' as an input from the user and then goes on to retrieve the data. We store the data in CSV format before adding the data to data frame in order to avoid any indexing issues that might persist. For our analysis, we retrieve seven years' worth of stock data of TESLA, from January 2014 to October 2021. The data obtained from YFinance consists of several columns namely Date, Open, High, Low, Close, Adj Close, Volume.

B. Analysis Method

For the purpose of evaluation of various Machine Learning techniques, it is deemed appropriate to showcase that the predicted model fits the data as precisely as possible. In doing so, proof of how well Linear Regression and Support Vector Machine Regression fit the data using the test dataset is given. For the Linear Regression Model, the target attribute is taken to be that of the Close value of the stock, while the rest of the columns are used to predict the target attribute. In a similar manner, for the Support Vector Machine model, "Close" is used as the target attribute.

C. Performance Evaluation Methods

The comparison of the models is to be done on the basis of their performance on the data, how accurate are the predictions. For this purpose, the performance measures for proposed Machine Learning system, used were that of Mean Absolute Error (MAE), Root Mean Square Error (RMSE) as well as the correlation coefficient. These are used to measure how close to the actual value or distant from the actual values are the predicted values.

D. Commands Used

The Linear Regression Analysis was performed by making use of Scikit Learn library's Linear Regression command.

The Support Vector Machine model was made use of by using the SVR command in the Scikit Learn library.
For the purpose of checking the stationarity of the data, the adfuller command was used, while to train the ARIMA model, the ARIMA command was made use of. For the purpose of calculation of performance parameters (MAE, RMSE), other functions of Scikit Learn were used.

VI. EXPERIMENTAL RESULTS

A. Linear Regression

For the Linear Regression model, similar to any Machine Learning model, the initial step is to get data. In this case, the data was retrieved using the YFinance library. The stock prices of the company TESLA within date range of 1-Jan-2014 to 30-Oct-2021 were obtained. The obtained data consisted of seven columns, out of which, the Date and Close column were plotted against each other to showcase the value of TESLA stocks throughout the years, as can be seen in figure 3.

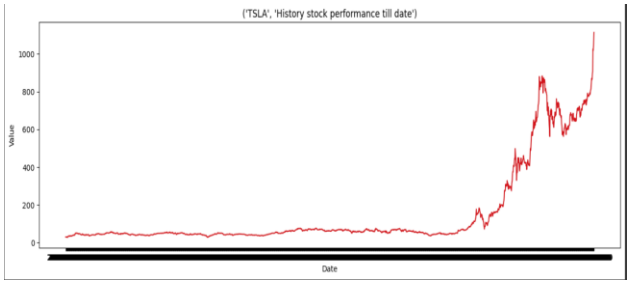

Figure 3: TESLA Stock performance

Two independent data frames were then created from the original data, one of which contained the target attribute Close, while the other data frame consisted of the attributes Open, High, Low and Volume. The two data frames were then spilt into train test data.
A linear model was then fit to the training set data from where the regression coefficient and regression intercept were obtained. The trained model was then used to provide predictions on the test data set, the results of which were stored in a separate data frame. A data frame of the predictions along with the actual price of the stock was created so as to be able to showcase how well the model did. Prediction chart for the same can be seen below:

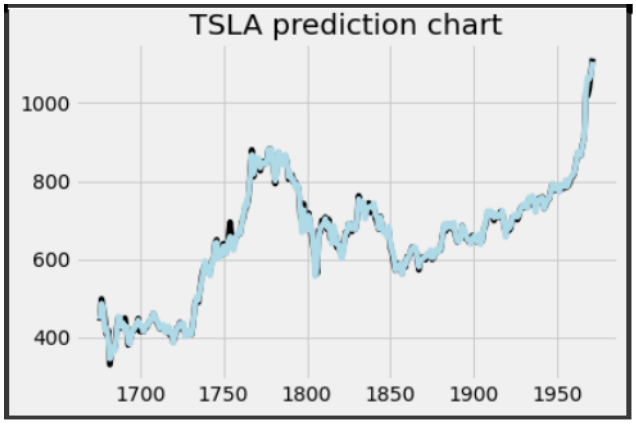

Figure 4: Linear Regression Predictions

The figure overhead, Figure 4, showcases the predictions made by the model in light blue against that of the actual stock prices in black. To get a better sense of the performance of the model, the below table, Table 1, can be referred:

Table 1: Performance Parameters for Linear Regression

Method	Result
MAE	6.44
RMSE	9.02
R-Squared	0.95

B. Support Vector Machine Regression

The initial steps performed were similar to that of Linear Regression, followed by making use of Predictions using Support Vector Machines, with the changes being addition of the “Dates” column to the independent variables data frame, which were then converted to integers. Apart from this, the only change was that of calling of the SVR command rather than that of the Linear Regression command.

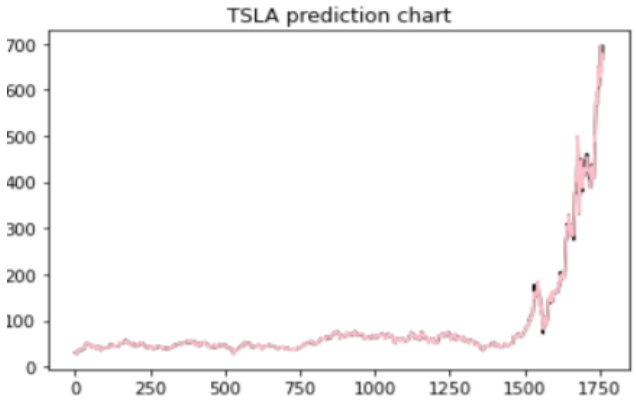

Figure 5: Support Vector Machine Predictions

The figure overhead, Figure 5, showcases the predictions made by the model in pink against that of the actual stock prices in black. To get a better sense of the performance of the model, the below table, Table 2, can be referred:

Table 2: Performance Parameters for Support Vector Regression

Method	Result
MAE	10.58
RMSE	16.02
R-Squared	0.91

C. Naïve Forecasting

For the purpose of the final three models, the data split was a bit different as compared to the previous two models. Arbitrary dates were chosen to separate the data set into train and test sets. The data was split in such a way that the training set comprised of the initial years of the data set, from the start of the year 2016, while for the test data, it comprised of the last year of the stocks. A series data frame was created which consisted of all the previous closing prices, which then was used to make a naïve forecast using the test-split indices. The below figure, figure 6, visualizes the predictions made by the forecast in comparison to that of the original prices.

Retrieval Number: 100.1/ijrte.B70830711222
DOI: 10.35940/ijrte.B7083.0711222
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

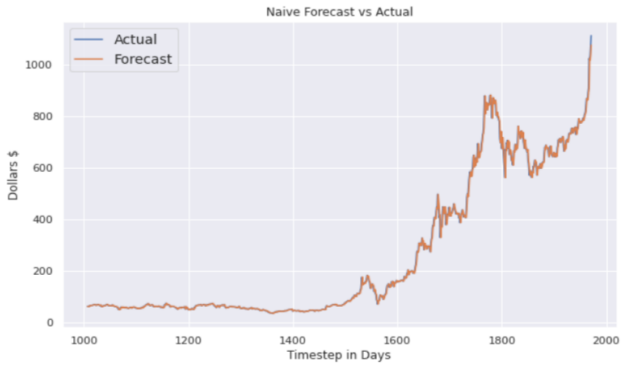

Figure 6: Naïve Forecast Predictions

The figure overhead, Figure 6, showcases the predictions made by the model in yellow against that of the actual stock prices in blue. To get a better sense of the performance of the model, the below table, Table 3, can be referred:

Table 3: Performance Parameters for Naïve Forecast

Method	Result
MAE	7.17
RMSE	13.87

D. Moving Average Forecasting

For the purpose of this project a Simple Moving Average is taken in to account spread over two time periods, the two time periods being a five-day time period and a twenty-day time period. The five-day time period is taken into account as the stock market is open five days a week and the twenty-day time period is taken into account as the stock market is open for roughly twenty days a month The train test split of the data was similar to that of the above model. A required window size was created, for the purpose of this project, the window sizes created were that of twenty for the monthly average and five for the weekly average, after which a moving average was created over the entire data set.

I. *Monthly Moving Average Model*

This model consists of the window size being set as 20. The predictions made by the model are visualized in figure 7.

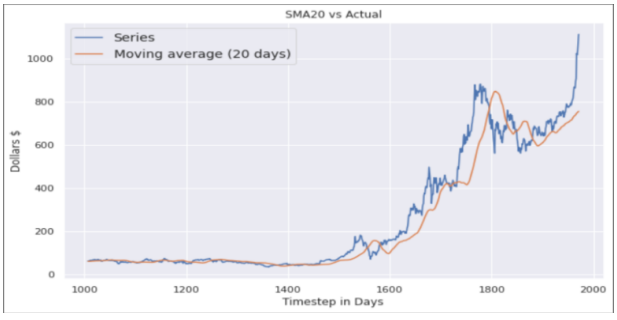

Figure 7: Monthly Moving Average Predictions

The figure overhead, Figure 7, showcases the predictions made by the model in yellow against that of the actual stock prices in blue. To get a better sense of the performance of the model, the below table, Table 4, can be referred:

Table 4: Performance Parameters for Monthly Moving Average

Method	Result
MAE	44.60
RMSE	76.19

II. *Weekly Moving Average Model*

This model consists of the window size being set as 5. The predictions made by the model are visualized in figure 8.

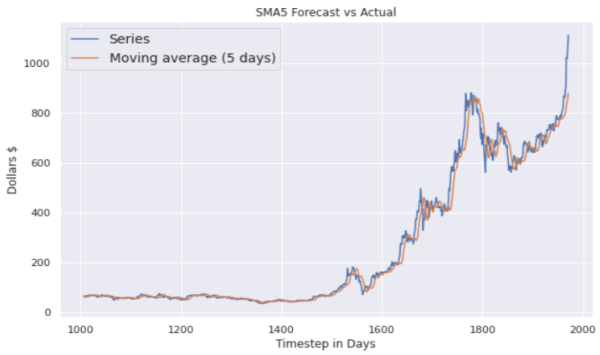

Figure 8: Weekly Moving Average Predictions

The figure overhead, Figure 8, showcases the predictions made by the model in yellow against that of the actual stock prices in blue. To get a better sense of the performance of the model, the below table, Table 5, can be referred:

Table 5: Performance Parameters for Weekly Moving Average

Method	Result
MAE	18.95
RMSE	35.10

E. ARIMA Model

The initial data split is equivalent to that of the previous two models. Null hypothesis to check the stationarity of the data was crated. With the help of the ADFuller test, p-value of 1.00 was obtained along with a test static of 3.059. The test static was greater than any of the critical values and the p value obtained as well was way greater than the significant level which is approximately equal to that of 0.05. On the basis of the above findings, the null hypothesis couldn't be rejected, meaning, our data was in fact non-stationary. A visualized graph of the same can be seen below in figure 9:

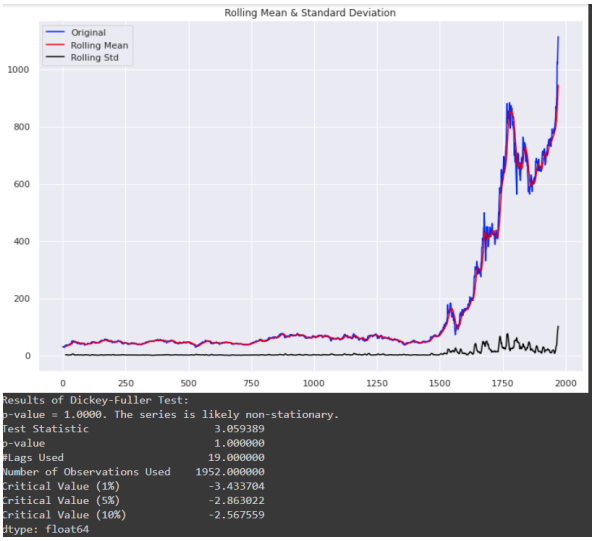

Figure 9: Initial data after ADFuller Test

Retrieval Number: 100.1/ijrte.B70830711222
DOI: 10.35940/ijrte.B7083.0711222
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

To tackle non-stationary data, differencing was used, which basically involves taking value at a time T and subtract the value at T-1 to get the difference in values. After differencing, the data obtained was run through the ADFuller test again, where in this time the p value obtained was less than the significant values as well as the test static was less than the critical value, hence, the null hypothesis could be rejected, meaning, now the data had been made stationary. The below figure, figure 10, visualizes the same:

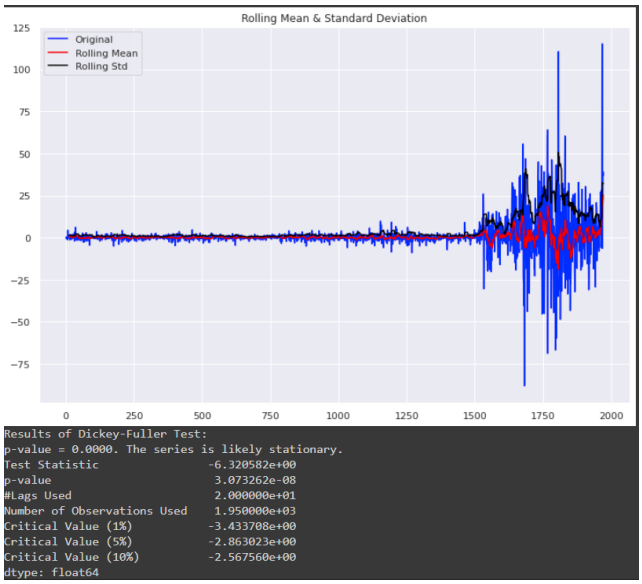

Figure 10: Differenced data after ADFuller Test

Following the AD Fuller test, ACF and PACF plots were looked and potential inputs for the ARIMA() command were looked into. Several combinations were tried, after which the best suited combination for Arima parameters was found to be (4,2,0). The same model was fitted, and tested. The predictions made by the fitted model can be seen in figure 11 below:

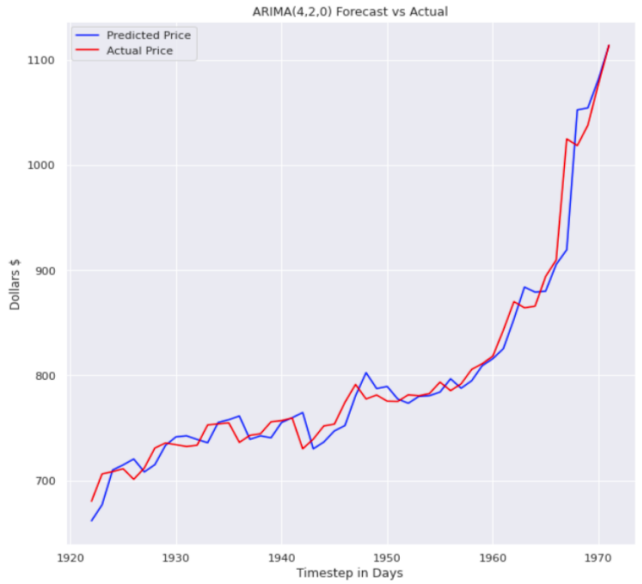

Figure 11: ARIMA Model Predictions

The figure overhead, Figure 11, showcases the predictions made by the model in blue against that of the actual stock prices in red. The error in the predictions made by the model can be graphically seen below depicted in Figure 12:

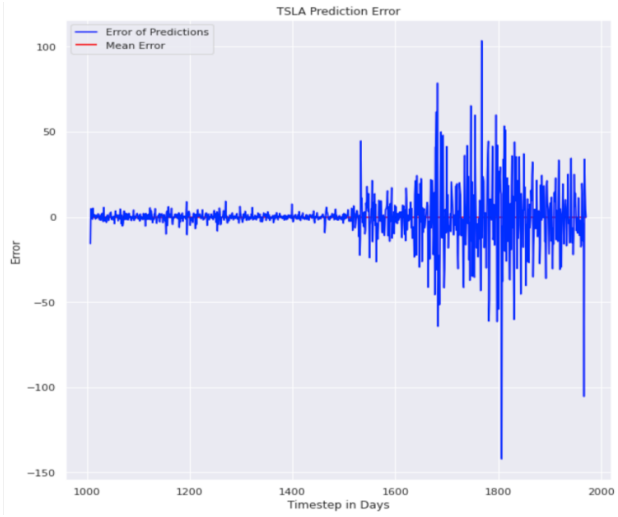

Figure 12: ARIMA Model Prediction Error

To get a better sense of the performance of the model, the below table, Table 6, can be referred:

Table 6: Performance Parameters for ARIMA Model

Method	Result
MAE	5.94
RMSE	15.23

F. Comparison

Considering an interpretation standpoint, Mean Absolute Error can be considered as a better parameter for linear problems than Root Mean Square Error, as not only does Root Mean Square Error describe average error alone, but also has other implications which are far more complex to understand. Along with this, Root Mean Square Error also gives greater importance to the considerably larger errors, due to which models will try to minimize these as much as possible. The following graph, figure 13, can be seen for the comparison of the MAE values of the models.

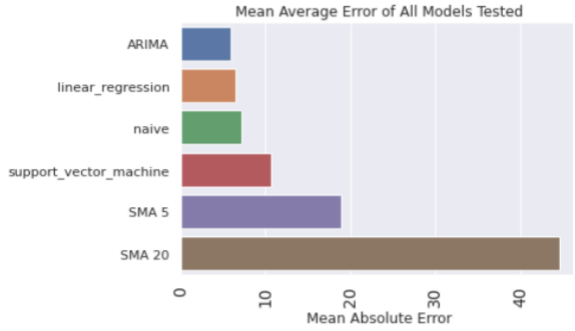

Figure 13: Comparison of Mean Absolute Error

The above figure 13 represents the Mean Absolute Error of all the models tested in an ascending manner, wherein, the model at the top has the smallest Mean Absolute Error, which makes it the best performing model, and the model represented at the bottom has the largest Mean Absolute Error. It can be seen from the figure that the ARIMA Model has a better performance in comparison to that of the rest of the model, closely followed by the Linear Regression Model.

Retrieval Number: 100.1/ijrte.B70830711222
DOI: 10.35940/ijrte.B7083.0711222
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

VII. CONCLUSION

In this paper we have tested the performance of various Machine Learning algorithms namely Linear Regression, Support Vector Machine Regression, Naive Model, Moving Average Model and Auto Regressive Integrated Moving Average Model (ARIMA) on stock price data. On comparison of the various algorithms, it can be seen that the performance of the ARIMA model, in terms of the performance parameters is the most efficient amongst the different algorithms, closely followed by Linear Regression. This can be explained as the various models might not be able to establish proper relationships between variables due to the non-stationary nature of the time series data, which the Arima model is better suited for.

ACKNOWLEDGMENT

We are immensely grateful to Dr. TK Siva Kumar, Asst. Prof. (Sr. Grade), Dept. of Computer Science and Engineering, SRM Institute of Science and Technology, Chennai whose consistent guidance and support was a milestone for this project.

REFERENCES

1. Puneet Singh, Prem Kumar Yadav, Shashank Chaurasia ,Shubham Bhardwaj, “Stock Price Predictions With ML Using Python”, IJSREM ,2020
2. Vaishnavi Gururaj , Shriya V R and Dr. Ashwini “Stock Market Prediction using Support Vector Machines”, IJEAR ,2019
3. “Support-Vector Networks” Corinna Cortes, Vladimir Vapnik
4. Model Bijesh Dhyani, Manish Kumar, Poonam Verma, Abhisheik Jain “Stock Market Forecasting Technique using ARIMA”,IJRTE,(2020) [CrossRef]
5. Prof.A.C.Taskar, Faisal Momin, Sunny Patel, “STOCK MARKET PREDICTION SYSTEM USING MACHINE LEARNING APPROACH” , IJSDR,2019
6. K. Hiba Sadia, Aditya Sharma, Adarrsh Paul, SarmisthaPadhi, Saurav Sanyal, “Stock Market Prediction Using Machine Learning Algorithms”, IJEAT,2020

AUTHORS PROFILE

Amaan Aijaz, was a student at Department of Computing Technologies, SRM Institute of Science and Technology, Chennai, India. His current research interests include Machine Learning, Deep Learning, Data Warehousing and Data Mining. Completed B. Tech from SRMIST (First Class with Distinction) with a Merit Scholarship. Well-informed on latest Machine Learning advancements. He has worked extensively in the areas requiring gathering, cleaning and organizing data for use by technical and non-technical personnel. Advanced understanding of statistical, algebraic and other analytical techniques. He has a number of certifications in the fields of Machine Learning, Data Science and Big Data Handling. He is presently workings on many more papers related to the fields of Machine Learning, Deep Learning.

Komal Rastogi, was a student at Department of Computing Technologies, SRM Institute of Science and Technology, Chennai, India. Her research interests lie in the fields of user experience design, user survey methodology, UI/UX Design. Completed B. Tech from SRMIST (First Class with Distinction). Advanced understanding of User Interaction, collaboration, user empathy, curiosity, communication skills, and visual communication skills. She has a number of certifications in the fields of UI and UX Designs. She is presently workings on many more papers related to the fields of Block Chain Technology. She has worked extensively in designing various applications for an improved user experience.

Dr. TK Sivakumar, is an Asst. Prof. (Sr. Grade), Department of Computing Technologies, SRM Institute of Science and Technology, Chennai, India. Completed Ph.D from Bharathiar University, Coimbatore. Research Interests: Network Security, SET Protocols, Network Security, Software Engineering, Cyber Crime, Distributed Operating Systems, Data Warehousing and Data Mining. Awarded Brainbench – (the measure of achievement), JAVA 2, November, 2000. Has published a number of other papers including “Enhanced Secure Data Encryption Standard (ES-DES) Algorithm Using Extended Substitution Box (S-Box),” International Journal of Applied Engineering Research (IJAER). Presented a number of papers. Currently acting as a Asst. Professor at SRMIST, dedicating his professional life towards the teaching of Networking and Network Security field to enthusiastic students.