**Predicting Performance of the Tesla Stock**

**Dipika Sharma**

**Bellevue University**

**Predictive Analytics 630**

**Andrew Hua**

Table of Contents

[Final Paper 3](#_Toc151220772)

[**Introduction:** 3](#_Toc151220773)

[**Data Selection:** 3](#_Toc151220774)

[**Project Methodology:** 5](#_Toc151220775)

[**Conclusion:** 14](#_Toc151220776)

[References: 15](#_Toc151220777)

## Final Paper

### **Introduction:**

Stock is a part of share in company ownership that depends on different assets of the company and its earning. The one who purchase the stocks are called stockholders and they considered to be the partial owners of the company. When company grow or falls the stock price also increase or decrease. The stock market is the place where one can purchase or sell stocks. Person can make easy money by buying the company shares at low price and sell them at higher price. Stock Market plays an important role in economy. This is important to predict trends in stock market so that we can reduce loss and can increase profit. However, this system has proven to be dynamic and complex, which indicate about the risk involved. In order to reduce risk factor, experienced investors have found it advantageous to use technology that predicts future stock prices.

The goal of this project is to predict stocks, Using the time-series forecasting I am trying to determine the future value based on previously observed values of the stock. This predictive model is beneficial for investment firms, financial institutes, and stock market investors.

### **Data Selection:**

Data for Stock market project is downloaded from Yahoo Finance cite a public domain that provides financial news, data, and commentary updated in real time. The Yahoo dataset was selected due to the high quality of data compared to other available stock datasets. They have historical data from Jan 2015 to Sep 2021. I am using the TESLA stock market data for this project.

A screenshot of a graph

Description automatically generated

The dataset contained 1699 rows and 7 attributes. The details of each attribute are as follow:

* Date: Represents the date of the relevant Transaction Day.
* Open: Represents the initial share price of the relevant Trading Day.
* High: Represents the highest price of the relevant Trading Day.
* Low: It represents the lowest price of the relevant trading day.
* Close: It represents the closing price of the stock on the relevant trading day.
* Adj Close: Represents the adjusted closing price of the stock of the relevant trading day.
* Volume: It represents the trading volume information of the relevant trading day.

### **Project Methodology:**

As mentioned , the objective of the project is to predict future values based on previously observed values. A classic approach to Stock Market Prediction includes fundamental and technical analysis, but a more modern approach includes machine learning and sentiment analysis. The model capable of this modern approach are time-series forecasting models. The Auto-Regressive Integrated Moving Average (ARIMA), which is a classical statistical method, will be the primary method and discussed in detail. The Long-Short Term Memory (LSTM) method, which is under deep machine learning, will be the secondary method. The stock market price time series data that expands over 7 years for a group of Tesla company will be used and prediction of closing stock price will be presented by both ARIMA and LSTM model

The project’s selected model is ARIMA model. This model was selected over LSTM model because of its ease of application and interpretability.

Once data cleansing and exploratory analysis was completed, I started working on the steps that we need to perform before building the time series model. Time series analysis only work with stationary data so as first step I performed ADF (Augmented Dickey-Fuller) Test to determine whether the series is stationary.

A graph with numbers and lines

Description automatically generated

Looking at the above figure we can see that the p value is bigger than 0.05 and hence we cannot rule out the NULL hypothesis. Also, the test statistics is greater than the critical values which mean data is non-linear. Hence our data is non-stationary. It is also important to separate the seasonal and trend from our series otherwise it can cause the resulting series to become stagnant. Using the decompose for the same, see below.

A graph of a graph

Description automatically generated with medium confidence

To eliminate trend or reduce the magnitude, first take the log of the close column and Since ADF test failed so we need first convert the non-stationary data to stationary data. In order to do that I took the log of the ‘Close’ column and then subtract the moving average from that. This approach makes my data to stationary.

A graph with red and blue lines

Description automatically generated

As we can see above that now the p value is less than 0.005 and test statistics is smaller than the critical values, this mean, data is now stationary.

The next step is to split the data into training and testing datasets. The training dataset will be used to prepare the ARIMA model and generate a prediction. The testing dataset, which is much smaller than the training dataset, will be the part of the data that we will try to predict. The size of our testing dataset, with dates ranging from January 2021 to September 2021, is about 11 percent of the size of our training dataset, which ranges back to the beginning of 2015 to 2020

A graph with a line graph

Description automatically generated

There are three integers that are typically used to parametrize nonseasonal ARIMA models. They are p, the number of autoregressive terms (AR order), d, the number of nonseasonal differences (differencing order), and q, the number of moving-average terms (MA order). I used auto\_arima function from the pmdarima package to find out the optimal order for an ARIMA model p, q, and d.

A screenshot of a computer

Description automatically generated

A close-up of graphs

Description automatically generated

The top left standardized residual errors appear to have a uniform variance and fluctuate around a mean of zero. The next top Right density plot suggests a normal distribution with a mean of zero. The bottom left shows that the red line is perfectly aligned with almost all the dots. And the last bottom right shows that the residual errors are not autocorrelated. From above summary results, ARIMA model finds the optimal order which is (0,1,0), it means p =0,d=1,q=0.

Using above values, I fit the ARIMA model on the train dataset and predict the stock price for test data. Below plot showing the Actual Price and Predicted price.

A graph with red and green lines

Description automatically generated

A line plot is created showing the Actual Price (red) compared to the rolling forecast predictions (Green). We can see the values show some trend and are in the correct scale.

Checking the top 10 values below to see the actual and predicted data more closely.

A table of numbers with black text

Description automatically generated

It’s quite visible that both Actual values and Predicted values are matching approximately and hence the built ARIMA model is predicting the stock price closely. In order to evaluate the quality of build regression model I used the below metrics.

A black text with black numbers

Description automatically generated with medium confidence

As we can see that the R2 score is near 1 which means that the model is able to predict the future values very well. Also mean squared error is low, MAPE is lower, and the explained variance score is high which indicate that the build model is able to predict stock price.

Lastly, a LSTM model was built as a comparison to the ARIMA model, and visualization have done to see the predicted values with actual stock price.

A graph with red and blue lines

Description automatically generated

Looking at the above plot we can see that the LSTM model also performed well, and we are able to predict future values with LSTM model. The model can perfectly follow most unusual jumps/drops in plot; however, we can observe that the model predicted lower/higher values than the actual stock price. In order to compare both ARIMA and LSTM models, the same metrics were used to evaluate the quality of LSTM model.

A black text on a white background

Description automatically generated

Looking at the above results, we can see that the R2 score is small and not close to 1 like ARIMA model. Also mean squared error is low but ARIMA model values are lower, and the explained variance score is negative which indicate that the quality of LSTM model is not good with current data and ARIMA model is better in predicting stock price.

### **Conclusion:**

The goal of the project is to predict the stock price of the Tesla. The ARIMA model is built, and the actual stock prices is compared with predicted stock price values by using the plot as well as by displaying the top 10 values and it was noted that both values are close to each other. To confirm the model performance, the different coefficient MS, MAE, RMSE, MAPE, R2 and explained variance score are calculated using test and predicted values. The results are quite encouraging. Overall, the ARIMA model is ultimate choice for predicting the future values of stock price.

ARIMA and LSTM models are best for time series forecasting. Both ARIMA and LSTM model capable of predicting future stock price and have admirably performed on the test dataset. Both techniques can be effective and valuable to investigate trends in stock market prices over large periods of time and provide some important information for investors to make decisions when investing in the stock market, but LSTM model have predicted slightly lower or higher values of the stock’s price compared to the actual stock price whereas the ARIMA model predicted the values which are much closer to the actual stock price. The performance coefficient of ARIMA model shows better results as R2 score close to 1, low mean squared error, high explained variance score and lower MAPE value compared to LSTM model. Although we can use any one of these models to predict future stock prices but with the current dataset, the recommended model is ARIMA, and it is best fit to predict the future stock prices and no other tweaking to the model or searching for better models is required now. There is future interest in using the built models on different dataset to check the quality of the model and to see how the built model will perform on different dataset.

## References:

1. Adam Hayes. (April21, 2023). What is a Stocks? from <https://www.investopedia.com/terms/s/stock.asp>
2. <https://finance.yahoo.com/quote/TSLA/history?p=TSLA>
3. Accidents Data from Kaggle website; <https://www.kaggle.com/datasets/sobhanmoosavi/us-accidents?resource=download>
4. The Wall Street Journal; Tesla historical Data; <https://www.wsj.com/market-data/quotes/TSLA/historical-prices>
5. Yahoo Finance API; Get stock price data using Python; <https://thecleverprogrammer.com/2021/12/21/get-stock-price-data-using-python/>
6. Computing Regression Accuracy; <https://subscription.packtpub.com/book/data/9781789808452/1/ch01lvl1sec12/computing-regression-accuracy>