

FE 511 Final Project Report

CWID: 10453187

Sushant Kumbhar

skumbha2@stevens.edu

Objective:

This project focuses on predicting the trend of the stock price of Facebook over the period of time using statistical model – ARIMA.

The next main objective is to build a Trading Strategy, meaning when to buy or sell the stock

The entire project has been done in Python using Jupyter notebook.

Data Extraction and Understanding:

The source of the data is Bloomberg Terminal using Historical Price “HP” and Excel Add-in.

The data consist of columns “Date”, “Last Price” and “Volume”. From these columns “Last Price” is used to predict the stock trend.

Date	Last Price	Volume
Fr 05/14/21	315.94	19,245,724
Th 05/13/21	305.26	18,079,160
We 05/12/21	302.55	24,641,010
Tu 05/11/21	306.53	18,920,086
Mo 05/10/21	305.97	24,239,170
Fr 05/07/21	319.08	15,916,743
Th 05/06/21	320.02	19,012,241
We 05/05/21	315.02	15,577,605
Tu 05/04/21	318.36	24,032,557
Mo 05/03/21	322.58	18,719,462
Fr 04/30/21	325.08	26,332,423
Th 04/29/21	329.51	56,526,771
We 04/28/21	307.10	33,907,210

Figure 1. Data in Bloomberg Interface

Date	PX_LAST	PX_VOLUME
05/14/21	315.94	19245724
05/13/21	305.26	18079160
05/12/21	302.55	24641010
05/11/21	306.53	18920086
05/10/21	305.97	24239170
05/07/21	319.08	15916743

Figure 2. Data Downloaded in Excel file

The Last Price column indicates the closing price of the stock and the Volume column indicates the number of people trading the stock on that date.

The data is then saved into CSV format in the local disk space. The data is then read into python IDE.

Data Cleaning:

As you can see in the excel file, the data is into the reverse order, meaning the recent data is first whereas older data is at the last. To make predictions we need data in ascending order of date. This is taken care in the Python IDE.

The entire dataset is on object format, so the Date column is converted into “datetime” format, Close and Volume columns are converted into “float” and “int” format respectively.

Date column is then set as the index of the data to make proper plots

	Close	Volume
Date		
2015-05-01	78.990	24136358.0
2015-05-04	78.810	14691991.0
2015-05-05	77.560	22311318.0
2015-05-06	78.100	28663026.0
2015-05-07	78.425	19749428.0

Figure 3. Data into Pandas DataFrame

The figure above shows the final data after cleaning and renaming columns. The figure below shows the trend of the stock over the period of time



Figure 4. Stock Price over Time

Building the Model:

As you can see in figure 4, the blue part of the plot is the training data for the statistical model ARIMA, and the green part is the testing data which the model is tested on.

ARIMA, short for 'Auto Regressive Integrated Moving Average' is actually a class of models that 'explains' a given time series based on its own past values, that is, its own lags and the lagged forecast errors, so that equation can be used to forecast future values.

An ARIMA model is characterized by 3 terms: p, d, q

Where,

1. p is the order of the AR term
2. q is the order of the MA term
3. d is the number of differencing required to make the time series stationary

After the building the model, it is fitted to the training data to predict the future points.

The figure below shows the plot of predicted values along with the actual values. The red color of the plot indicates the Actual values and the Green color indicates the Predicted points.



Figure 5. Predicted and Actual values

The figure 6. below indicates the Actual value of the stock vs the predicted value by the model for the date 04/30/2021.

```
: 1 today_actual = data['Close'].loc['4/30/2021']
  2 today_predicted = predict[0].loc['4/30/2021']
  3 print(today_actual, ' ', today_predicted)
```

325.08

328.35937071333734

Figure 6. Actual value vs Predicted value for the date /02/20/2021

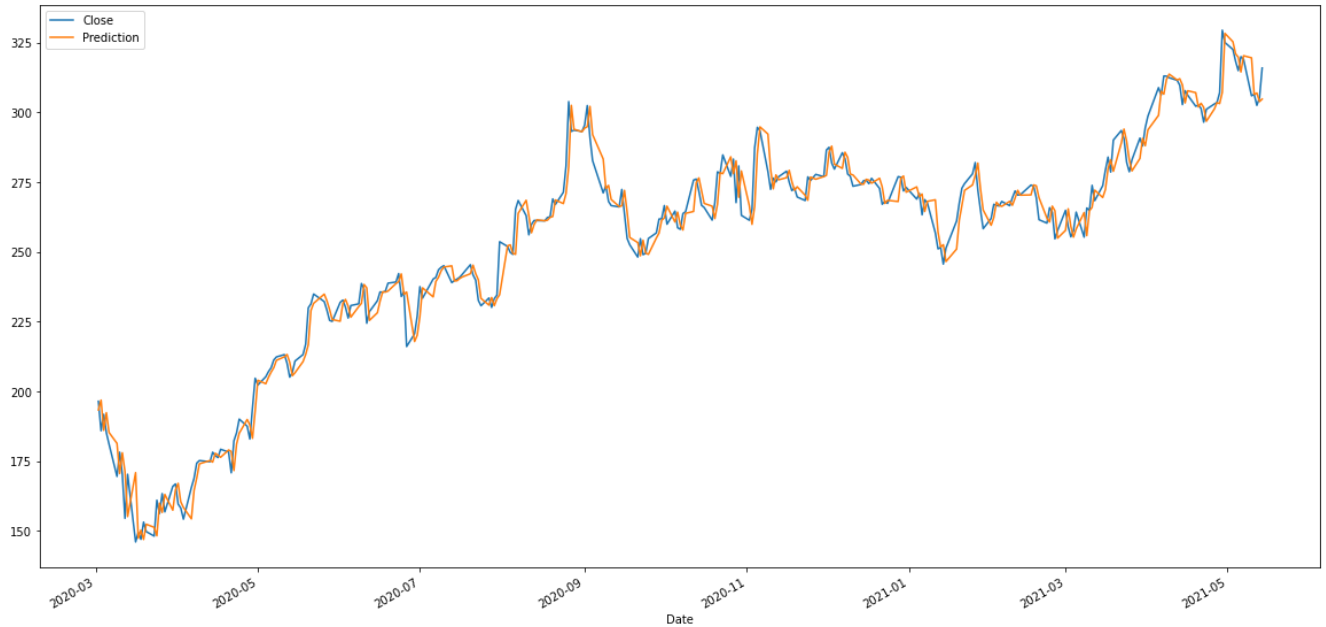


Figure 7. Predicted vs Actual price Plot

Trading Strategy:

Now Z score is calculated which is a numerical measurement used in statistics of a value's relationship to the mean (average) of a group of values, measured in terms of standard deviations from the mean. If a Z-score is 0, it indicates that the data point's score is identical to the mean score.

	Z Score	Close	Prediction
Date			
2020-03-02	0.524225	196.44	193.33
2020-03-03	-2.068257	185.89	196.86
2020-03-04	1.053455	191.76	186.04
2020-03-05	-1.409518	185.17	192.38
2020-03-06	-0.869118	181.09	185.26

Figure 8. Z score table for first 5 predicted data points

Summary:

Using the Z score we can create a Trading strategy by setting the threshold for z-score
When to Buy, Sell or Hold the stock?

Buy when stock is underpriced $\text{testZScores['Z Score']} \leq -1.00$

Sell when stock is overpriced $\text{testZScores['Z Score']} \geq 1.00$

Hold otherwise $(\text{testZScores['Z Score']} < 1.00) \& (\text{testZScores['Z Score']} > -1.00)$

	Close	Action
Date		
2020-03-02	196.44	Sell
2020-03-03	185.89	Buy
2020-03-04	191.76	Sell
2020-03-05	185.17	Buy
2020-03-06	181.09	Buy
2020-03-09	169.50	Buy
2020-03-10	178.19	Sell
2020-03-11	170.24	Buy
2020-03-12	154.47	Buy
2020-03-13	170.28	Sell

Figure 9. Action table indicating whether one should Buy, Sell or Hold the stock

The Figure 9. Indicates whether we should buy, sell or hold the stock. Thus, in this project I have successfully built the model to predict stock price and also created trading strategy.