Google stock - Time Series Analysis

Ajay Kumar

Sarthak Meliwal

Swanand Jamadagni

Vedantini Bogawat

Pritesh Manankumar Gujarati

Executive Summary:

The analysis aims to predict the average monthly trade volume for Google stock by identifying patterns and trends in historical data sourced from Yahoo Finance over 11 years. The analysis can provide valuable insights to investors, traders, and anyone interested in the performance of Google stock.

The data analysis includes the identification of the predictability of the original dataset based on the coefficients of Autocorrelation. The results suggest that the data exhibits predictability, allowing for informed predictions to be made.

Additionally, the analysis has identified an overall downwards trend in the trade volume of Google stock over the past 11 years. There is also a monthly seasonality pattern observed, indicating that trade volume tends to fluctuate at certain times of the year.

To make informed predictions, the analysis has used several forecasting models, including Optimal Holts Hinter, Two Level forecasting with Trend and seasonality, ARMA, and ARIMA models. These models help to identify trends and patterns in the data, allowing for more accurate predictions of future trade volume.

Overall, the analysis provides valuable insights into the performance of Google stock and can help investors make informed decisions about trading. By utilizing historical data and identifying patterns and trends, the analysis can provide more accurate predictions of future trade volume, allowing investors to maximize their returns.

Data Introduction:

The dataset provides several insights into the performance of Google stock. Firstly, the market capitalization of Google is $1.1 trillion, indicating that it is a significant player in the stock market. The stock value of Google has changed by 28.19% in the last 12 months, which is a significant increase. This suggests that Google stock has performed well in the market over the past year.

The highest stock value recorded in the last year was $144.6, while the lowest stock value was $83.45. This range indicates the volatility of the stock market and the potential risk associated with investing in stocks. Over the last 3 months, the average traded volume of Google stock was $30.19 million dollars, which is a substantial amount of trade volume. This information can be valuable for investors who are interested in trading Google stock.

Finally, the dataset indicates that the quarterly growth rate for Google is 1%, which suggests that the company is experiencing steady growth. This information can be useful for investors who are looking for long-term investment opportunities.

Overall, the dataset provides valuable insights into the performance of Google stock, including its market capitalization, stock value, trade volume, and quarterly growth rate. This information can be beneficial for investors who are looking to make informed decisions about investing in Google stock. By analyzing historical data, investors can better understand the performance of Google stock and make more informed investment decisions.

Step 1: Define Goal:

Our project aims to provide an in-depth analysis and accurate forecasting of the stock data listed on Yahoo Finance. In order to do this, we created a data integration script utilizing Python API that enables us to choose the businesses/cryptocurrencies and types of time series data we want to study. The script will produce a CSV file, which we will import into R for forecasting and time series analysis. Google will be the subject of our investigation, and we will use at least five different time series forecasting techniques on each of them. We will forecast future stock prices using previous data with a daily frequency. We will test and use a variety of forecasting methods, including moving averages, regression models, ARIMA models, two-level models, and ensemble methods, in order to provide reliable findings. Based on performance and accuracy metrics, the chosen model will be the last one.

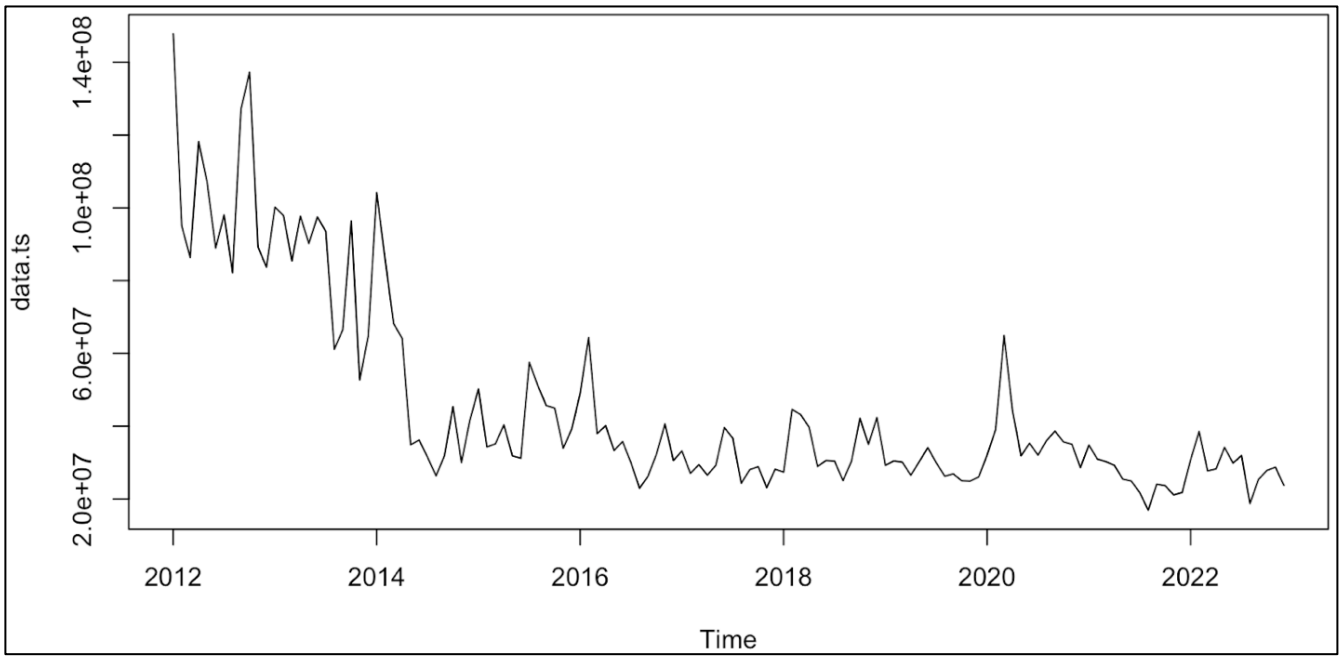
Step 2: Get data

This report focuses on the stock data of Google, which was collected using a python API and Yahoo Finance. The data collected includes the following columns: Date, Open, High, Low, Close, Volume, Dividends, Stock, and Splits. The primary objective of this report is to forecast the Volume column using historical data, which ranges from 2012-01-03 00:00:00-05:00 to 2022-12-30 00:00:00-05:00, with daily frequency.

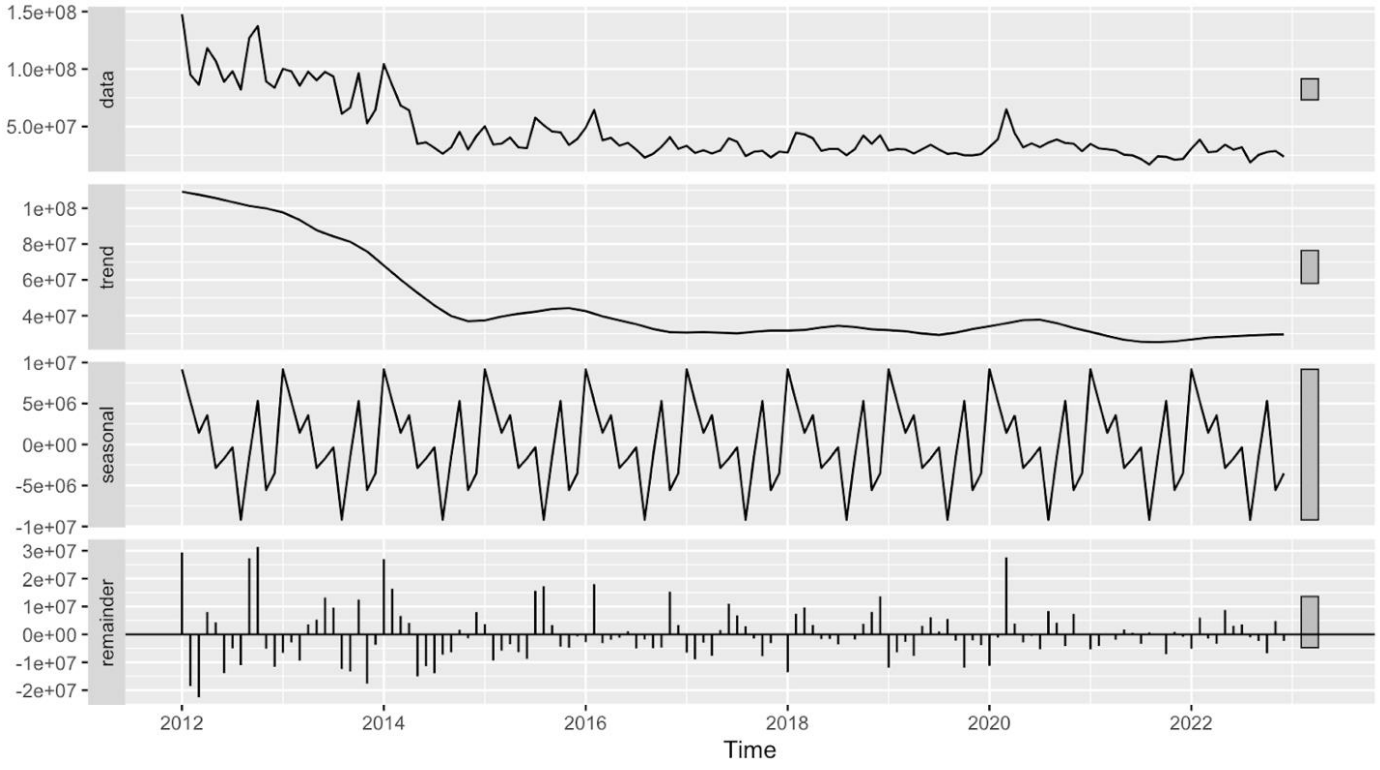
The data collection process involved the use of a python API and Yahoo Finance. The script was designed to retrieve data from Yahoo Finance for Google. The data was collected for the mentioned columns, which were deemed relevant to our analysis. The data was collected daily, and the range of data collected spanned from 2012-01-03 00:00:00-05:00 to 2022-12-30 00:00:00-05:00.

Step 3: Explore and Visualize Series:

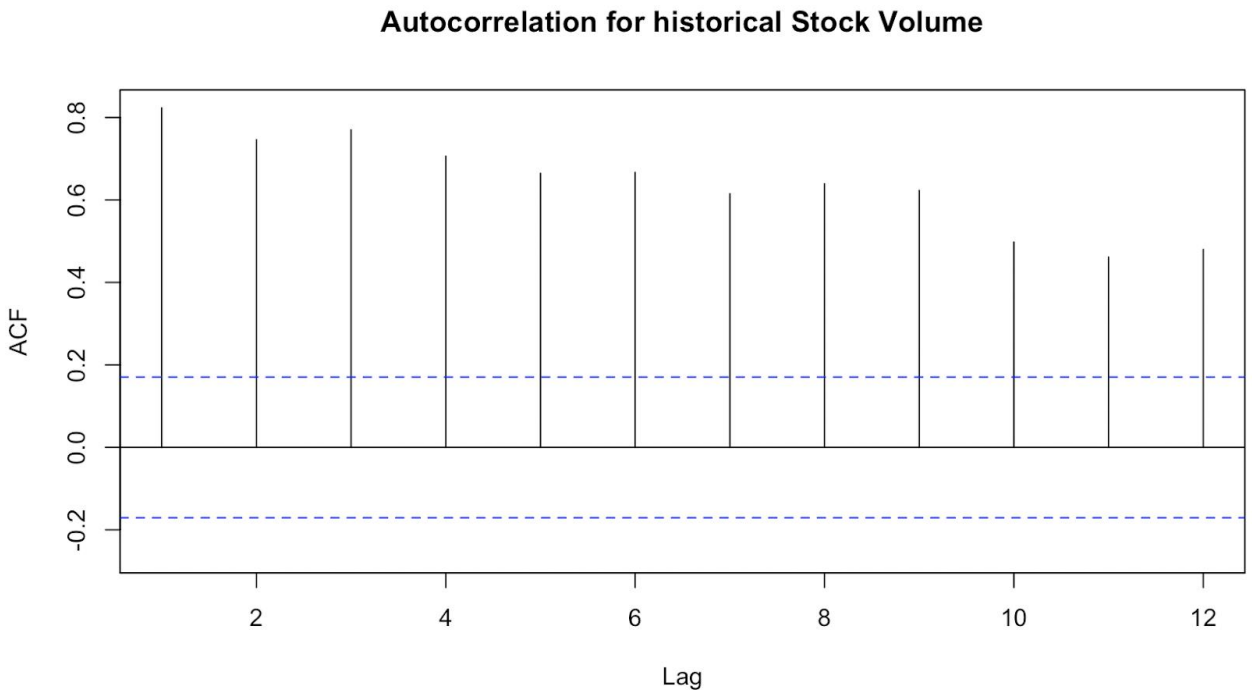
1. Data Trend



Time Series Components visualization



2. Add autocorrelation graph



Step 4: Partition Series:

To facilitate training and validation, the data was partitioned into two subsets. The training subset contained 120 records, while the validation subset contained 24 records. These subsets were utilized for the time period ranging from 2012 to 2022. The partitioning of both subsets is presented in appendix figures 1 and 2.

Training set:

```
> train.ts
           Jan       Feb       Mar       Apr       May       Jun       Jul       Aug       Sep
2012 147837261  95084268  86377464 118196776 107078585  88911637  98054923  82156008 127164548
2013 100193576  97924411  85416767  97732594  90186415  97525384  93522874  61113851  66495310
2014 104220997  85680584  68188667  64148208  34883797  36192810  31464422  26342029  31921209
2015  50228124  34310996  35135108  40359646  31865900  31200455  57562727  51159429  45671429
2016  49011789  64367000  37947636  40146667  33330000  35784091  29914800  22943826  26239429
2017  33200100  27035368  29421826  26546000  29254636  39625091  36679200  24311304  28053800
2018  27369905  44612632  43193619  39729429  28954000  30567619  30419143  25050783  30381474
2019  29251429  30455474  30104095  26528762  30288545  34096800  29953273  26256727  26899400
2020  32087238  39032211  64901909  44128667  31866300  35298545  32042727  36029429  38635810
           Oct       Nov       Dec
2012 137272228  89246029  83703570
2013  96402600  52710032  64704862
2014  45359498  30021778  41695343
2015  44944182  33934800  39182091
2016  32339333  40611619  30568857
2017  28890636  23110000  28163700
2018  42169304  34985810  42376421
2019  25007217  24923700  26054381
2020  35685636  35035400  28605364
```

Validation set:

```
> valid.ts
          Jan      Feb      Mar      Apr      May      Jun      Jul      Aug      Sep      Oct
2021 34823895 30977895 30255826 29261619 25455700 24938909 21751238 16923273 24057048 23646286
2022 30775500 38560526 27713913 28247000 34166000 29863429 31938330 18737457 25380624 27835119
          Nov      Dec
2021 21103429 21784545
2022 28735667 23742162
```

Step 5 & 6: Apply Forecasting & Comparing Performance

Data.ts

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
2012	36.68	48	43.55	51.56	54.89	50.88	52.01	52.89	47.45	43.9	44.95	43.32
2013	46.06	44.97	43.07	41.69	42.82	42.65	49	47.72	50.9	53	53.03	57.35
2014	64.73	61.42	60.58	68.86	66.94	70.41	67.87	68.98	76.24	76.25	71.6	67.78
2015	69.71	67.98	67.8	64.58	62.3	66	62.06	62.56	60.54	56.86	53.97	54.99
2016	51.28	52.16	54.08	54.27	52.87	52.84	48.06	48.27	46.39	46.2	50.32	51.69
2017	50.35	50.3	48.44	49.69	48.64	47.21	48.04	48.2	49.14	49.13	52.85	53.67
2018	50.44	52.24	52.14	50.91	52.42	50.92	46.45	46.63	46.47	47.81	50.21	45.99
2019	45.88	45.17	43.87	41.57	42	41.31	45.51	48.91	48.85	47.99	47.52	52.15
2020	58.41	56.29	54.75	57.74	56.35	58.5	58.64	56.05	51.79	51.28	53.52	53.66

2021	50.02	50.66	48.16	44.99	48.72	47.25	48.21	47.52	44.22	45.89	49.37	45.11
2022	47.48	50.81	52.97	48.8	48.28	49.41	51.58	57.26	58.54	58.9	59.37	56.9

"Our project work involved developing and comparing several forecasting models for a time series dataset. The goal of this project was to predict the future values of the time series and determine the accuracy of each model.

To achieve this goal, we first divided the original time series data into a 4:1 ratio for training and validation data. We then used this training data to develop the following models: Trailing MA with k=4,7,10, Holt's Winter (AAA), Optimal Holt's Winter (ZZZ), Linear Trend and seasonality, 2 level TSLM + MA, Auto Regression AR(1), 2 level TSLM + AR(1), and Auto Arima.

Once we had developed these models, we used the forecasted values from the validation data to measure the accuracy of each model. We compared the actual values of the validation data with the predicted values from each model to determine which model was the most accurate. Based on this accuracy measure, we narrowed down the models to the best four to use on the entire dataset (training + validation).

Finally, we used these four models to predict the future values of the time series for the next 12 months. The purpose of this was to determine which model was the most accurate and could be used for future predictions.

Throughout this project, we worked collaboratively to develop and compare these forecasting models. By sharing our expertise and working together, we were able to effectively evaluate each model and determine which was the most accurate.

Overall, this project allowed us to gain a deeper understanding of time series forecasting and the various models used for this purpose. We also developed our collaborative skills and learned how to effectively work together to achieve a common goal."

Optimal Holt's Winter model:

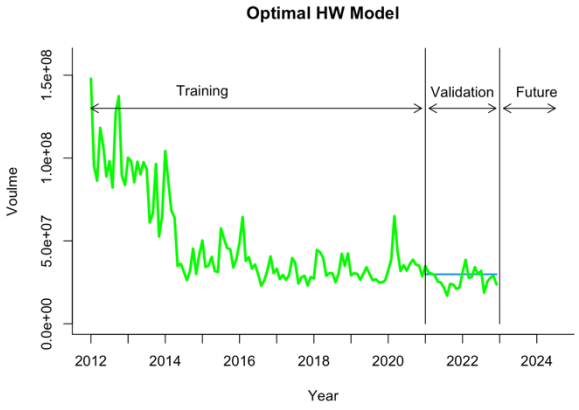
The Holt-Winters method is a popular time series forecasting technique that uses exponential smoothing to predict future values. The method is also known as the triple exponential smoothing method, as it considers three main components of a time series: trend, seasonality, and level.

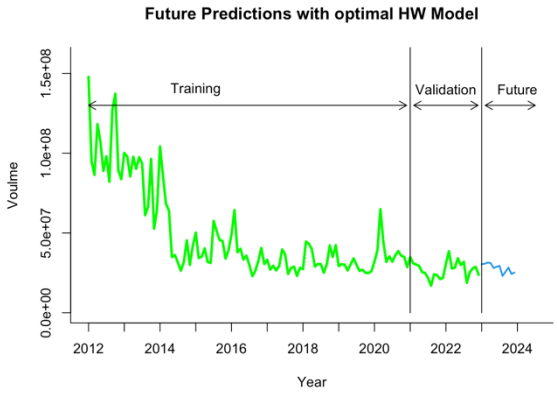
The optimal Holt-Winters model is determined by selecting the model parameters that provide the best fit to the historical data. This typically involves selecting the values for the smoothing constants alpha, beta, and gamma, which control the degree of smoothing applied to the level, trend, and seasonal components, respectively.

The optimal values for these parameters are usually found through a process of trial and error, or by using automated optimization algorithms that search for the best parameter values. Once the

optimal parameters are determined, the model can be used to make forecasts of future values based on the historical data.

Overall, the Holt-Winters method is a flexible and widely used approach for forecasting time series data, and the optimal model parameters are key to achieving accurate and reliable predictions.





Forecast for the validation period using optimal holt's winter model:

```
> hw.opt.pred
         Point Forecast     Lo 0     Hi 0
Jan 2021       29853416 29853416 29853416
Feb 2021       29853416 29853416 29853416
Mar 2021       29853416 29853416 29853416
Apr 2021       29853416 29853416 29853416
May 2021       29853416 29853416 29853416
Jun 2021       29853416 29853416 29853416
Jul 2021       29853416 29853416 29853416
Aug 2021       29853416 29853416 29853416
Sep 2021       29853416 29853416 29853416
Oct 2021       29853416 29853416 29853416
Nov 2021       29853416 29853416 29853416
Dec 2021       29853416 29853416 29853416
Jan 2022       29853416 29853416 29853416
Feb 2022       29853416 29853416 29853416
Mar 2022       29853416 29853416 29853416
Apr 2022       29853416 29853416 29853416
May 2022       29853416 29853416 29853416
Jun 2022       29853416 29853416 29853416
Jul 2022       29853416 29853416 29853416
Aug 2022       29853416 29853416 29853416
Sep 2022       29853416 29853416 29853416
Oct 2022       29853416 29853416 29853416
Nov 2022       29853416 29853416 29853416
Dec 2022       29853416 29853416 29853416
```

Forecast for the future period using optimal holt's winter model:

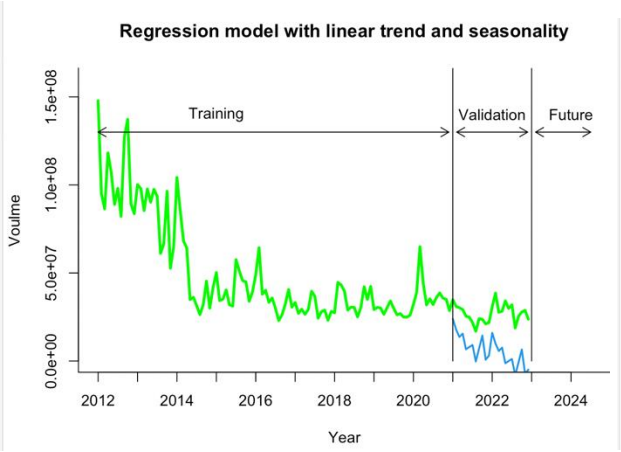
```
> hw.opt.future.pred
         Point Forecast    Lo 0    Hi 0
Jan 2023       30446110 30446110 30446110
Feb 2023       30691717 30691717 30691717
Mar 2023       31429527 31429527 31429527
Apr 2023       31188276 31188276 31188276
May 2023       28038842 28038842 28038842
Jun 2023       28800927 28800927 28800927
Jul 2023       29378510 29378510 29378510
Aug 2023       23094932 23094932 23094932
Sep 2023       25622728 25622728 25622728
Oct 2023       28328392 28328392 28328392
Nov 2023       24301352 24301352 24301352
Dec 2023       25175172 25175172 25175172
```

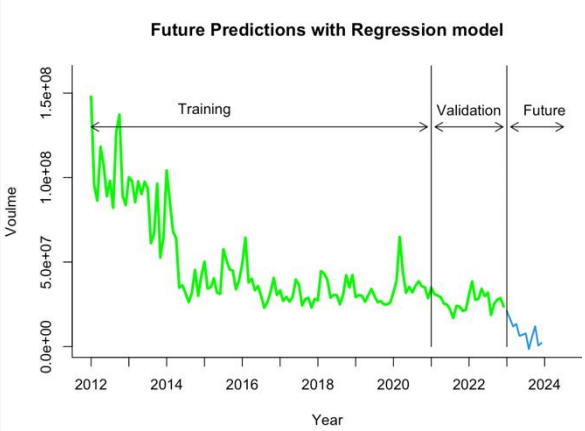
Linear regression model with trend and seasonality:

Linear regression model with trend and seasonality is a type of time series forecasting model that takes into account both the overall trend of a time series and its seasonal patterns. This model assumes that the future values of the time series can be predicted by a linear combination of a time trend variable and seasonal variables.

The trend variable represents the overall upward or downward movement of the time series over time, while the seasonal variables capture the cyclical patterns that occur within each season. These variables are combined using linear regression to create a forecast model that can predict future values of the time series.

The model is particularly useful for time series data that have both a clear trend and seasonal patterns, as it allows for more accurate predictions of future values. By incorporating both trend and seasonality into the model, it can account for both short-term and long-term patterns in the data, making it a powerful tool for time series forecasting.





Forecast for the validation period using regerssion model with linear trend and seasonality:

```
> reg.ts.prediction
         Point Forecast        Lo 0        Hi 0
Jan 2021    23953209.69 23953209.69 23953209.69
Feb 2021    17853490.18 17853490.18 17853490.18
Mar 2021    13651729.00 13651729.00 13651729.00
Apr 2021    15521690.80 15521690.80 15521690.80
May 2021     6654071.95  6654071.95  6654071.95
Jun 2021     7931211.11  7931211.11  7931211.11
Jul 2021     9088061.96  9088061.96  9088061.96
Aug 2021     -273127.36  -273127.36  -273127.36
Sep 2021     7071208.48  7071208.48  7071208.48
Oct 2021    14472122.61 14472122.61 14472122.61
Nov 2021      750848.45   750848.45   750848.45
Dec 2021     3025895.23  3025895.23  3025895.23
Jan 2022    16001620.10 16001620.10 16001620.10
Feb 2022     9901900.59  9901900.59  9901900.59
Mar 2022     5700139.41  5700139.41  5700139.41
Apr 2022     7570101.21  7570101.21  7570101.21
May 2022    -1297517.64 -1297517.64 -1297517.64
Jun 2022      -20378.48   -20378.48   -20378.48
Jul 2022     1136472.37  1136472.37  1136472.37
Aug 2022    -8224716.95 -8224716.95 -8224716.95
Sep 2022     -880381.11  -880381.11  -880381.11
Oct 2022     6520533.02  6520533.02  6520533.02
Nov 2022    -7200741.14 -7200741.14 -7200741.14
Dec 2022    -4925694.36 -4925694.36 -4925694.36
```

Forecast for the future period using regerssion model with linear trend and seasonality:

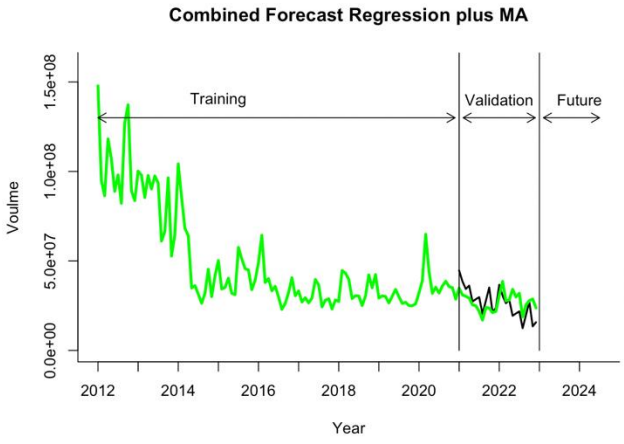
```
> reg.ts.future.prediction
         Point Forecast       Lo 0       Hi 0
Jan 2023     21054940.7 21054940.7 21054940.7
Feb 2023     16422354.4 16422354.4 16422354.4
Mar 2023     11932851.4 11932851.4 11932851.4
Apr 2023     13420900.2 13420900.2 13420900.2
May 2023      6357673.9  6357673.9  6357673.9
Jun 2023      6964482.1  6964482.1  6964482.1
Jul 2023      7809835.6  7809835.6  7809835.6
Aug 2023     -1488304.7 -1488304.7 -1488304.7
Sep 2023      5773146.7  5773146.7  5773146.7
Oct 2023     12014234.0 12014234.0 12014234.0
Nov 2023       638436.1   638436.1   638436.1
Dec 2023      2107802.7  2107802.7  2107802.7
```

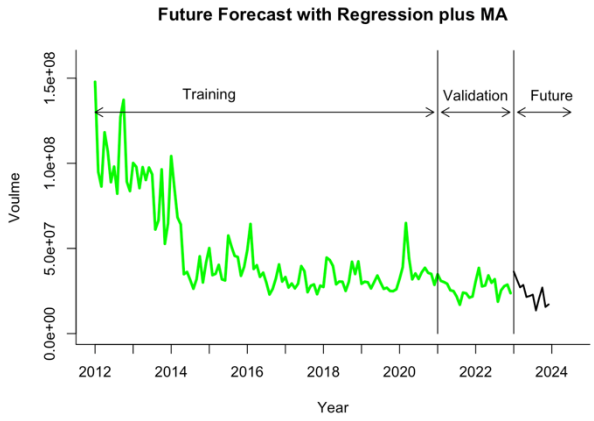
2 level forecasting model using trailing MA

The two-level forecasting model using regression and trailing MA is a time series forecasting approach that involves using two levels of models to generate predictions. In this method, the first level involves computing the trailing moving average (MA) of the historical data, as in the previous explanation.

The second level involves building a regression model using the first level MA as the input variable, along with other relevant predictors. This regression model captures the relationship between the first level MA and the target variable, allowing for more accurate predictions.

By using both a trailing MA and a regression model, the two-level forecasting model is able to capture both short-term and long-term trends in the data, as well as any relevant predictors that may affect the target variable. This approach can be useful for forecasting time series data in industries such as finance, economics, and marketing.





Forecast for the validation period using 2 level forecast using regression and moving average:

```
> reg_ma_2_level_forecast.df
    Pricing Regression_Forecast Trailing MA Residual Two Level Forecast
1  34823895         23953209.69              20689803           44643013
2  30977895         17853490.18              20682939           38536430
3  30255826         13651729.00              20677448           34329177
4  29261619         15521690.80              20673056           36194746
5  25455700          6654071.95              20669541           27323613
6  24938909          7931211.11              20666730           28597941
7  21751238          9088061.96              20664481           29752543
8  16923273          -273127.36              20662681           20389554
9  24057048          7071208.48              20661242           27732451
10 23646286         14472122.61              20660090           35132213
11 21103429           750848.45              20659169           21410018
12 21784545          3025895.23              20658432           23684327
13 30775500         16001620.10              20657843           36659463
14 38560526          9901900.59              20657371           30559272
15 27713913          5700139.41              20656994           26357133
16 28247000          7570101.21              20656692           28226793
17 34166000         -1297517.64              20656450           19358933
18 29863429           -20378.48              20656257           20635879
19 31938330          1136472.38              20656103           21792575
20 18737457         -8224716.95              20655979           12431262
21 25380624          -880381.11              20655880           19775499
22 27835119          6520533.02              20655801           27176334
23 28735667         -7200741.14              20655737           13454996
24 23742162         -4925694.36              20655687           15729992
```

Forecast for the future period using 2 level forecast using regression and moving average:

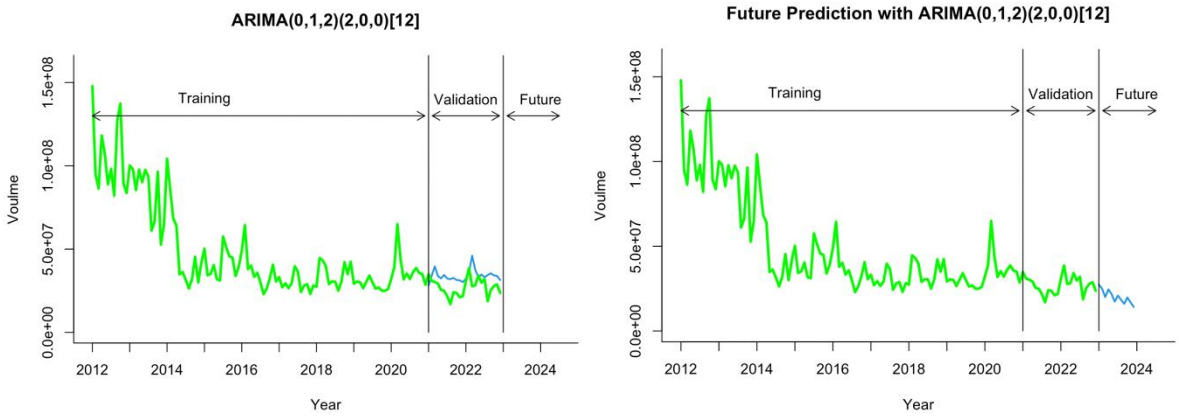
```
> reg_ma_2_level_future_forecast.df
   Regression_Forecast Trailing MA Residual Two Level Forecast
1          21054940.7              15327541           36382482
2          16422354.4              15259116           31681470
3          11932851.4              15204376           27137227
4          13420900.2              15160584           28581484
5           6357673.9              15125550           21483224
6           6964482.1              15097523           22062006
7           7809835.6              15075102           22884937
8          -1488304.7              15057165           13568860
9           5773146.7              15042815           20815961
10         12014234.0              15031335           27045569
11           638436.1              15022151           15660587
12          2107802.7              15014804           17122607
```

Auto Arima model on validation period:

Auto ARIMA is a forecasting algorithm that uses machine learning techniques to automatically select the optimal parameters for an ARIMA model. ARIMA stands for Autoregressive Integrated Moving Average and is a popular time series forecasting method.

Auto ARIMA works by analyzing historical data and identifying patterns and trends. It then selects the best combination of ARIMA parameters (i.e., p, d, q) based on the Bayesian Information Criterion (BIC) or the Akaike Information Criterion (AIC). This process is done automatically, without the need for manual intervention, making it a time-efficient and effective tool for forecasting.

Auto ARIMA is widely used in industries such as finance, healthcare, and retail for predicting trends, identifying anomalies, and making informed decisions.



Forecast for the validation period using optimal ARIMA model:

```
> train.auto.arima.pred
         Point Forecast     Lo 0     Hi 0
Jan 2021       28257057 28257057 28257057
Feb 2021       34034880 34034880 34034880
Mar 2021       39605783 39605783 39605783
Apr 2021       33798736 33798736 33798736
May 2021       32400377 32400377 32400377
Jun 2021       34472228 34472228 34472228
Jul 2021       32323285 32323285 32323285
Aug 2021       31922525 31922525 31922525
Sep 2021       32718224 32718224 32718224
Oct 2021       31414887 31414887 31414887
Nov 2021       31242934 31242934 31242934
Dec 2021       30218973 30218973 30218973
Jan 2022       32228054 32228054 32228054
Feb 2022       35900476 35900476 35900476
Mar 2022       46070160 46070160 46070160
Apr 2022       37610498 37610498 37610498
May 2022       33063368 33063368 33063368
Jun 2022       34705880 34705880 34705880
Jul 2022       33107401 33107401 33107401
Aug 2022       34397528 34397528 34397528
Sep 2022       35473711 35473711 35473711
Oct 2022       34166969 34166969 34166969
Nov 2022       33904328 33904328 33904328
Dec 2022       31455976 31455976 31455976
```

Forecast for the future period using optimal ARIMA model:

```
> auto.arima.future.pred
         Point Forecast     Lo 0     Hi 0
Jan 2023       27477039 27477039 27477039
Feb 2023       28711124 28711124 28711124
Mar 2023       26073223 26073223 26073223
Apr 2023       25868235 25868235 25868235
May 2023       25942818 25942818 25942818
Jun 2023       24821565 24821565 24821565
Jul 2023       24245349 24245349 24245349
Aug 2023       19751470 19751470 19751470
Sep 2023       23541737 23541737 23541737
Oct 2023       23952078 23952078 23952078
Nov 2023       23325219 23325219 23325219
Dec 2023       22440182 22440182 22440182
```

Models	RMSE	MAPE
Optimal Holt's Winter	11455966	16.334
Regression with linear trend and seasonality	18029623	35.615
2 level regression with MA	10896372	18.428
Auto ARIMA	10903360	17.425
Naive	13497578	19.301
Seasonal Naive	20582762	38.669

Conclusion:

- The above table suggests that out of the four models considered, Optimal Holt's winter and Auto Arima have the lowest Mean Absolute Percentage Error (MAPE), making them the best forecasting models. It also mentions that the decision has been made to pick Auto Arima for forecasting, as it not only has the least MAPE but also the lowest Root Mean Square Error (RMSE).

- It is important to note that performance metrics, such as MAPE and RMSE, are useful for comparing models and identifying the best-performing model. However, they should not be the only criteria for selecting a model. Other factors, such as the interpretability of the model and its computational complexity, should also be considered.

- Moreover, even the best-performing model may not always produce accurate predictions as forecasting is inherently uncertain. Therefore, it is essential to validate the chosen model using out-of-sample data or sensitivity analysis to assess the model's robustness.

- In conclusion, while selecting Auto Arima as the forecasting model based on its low MAPE and RMSE values may be a reasonable decision, it is important to consider other relevant factors and evaluate the model's performance using appropriate validation techniques.

Appendix :

Link : https://github.com/vedantini/Time_Series_Analytics

We have modularized our code and called the required files.

The files and their description are below:

- **GetData.py** : Used to fetch data in the required time range.
- **Check_Predictability.R** : Checks if the time series is predictable.
- **DataPreProcessing.R** : Aggregates the stock volume per month by taking average.
- **Generate_Time_Series.R** : Converts the data into time series format.
- **Model_Plot.R** : Used to plot the graph of training, validation and future.
- **Main.R** : This is the main code and utilizes the all modules as needed.
- **Project Veda.R, Project Sarthak.R, Project Pritesh.R** : Have the Time series models mentioned in this document.