Analysis of Stock Prices using some Time Series Models

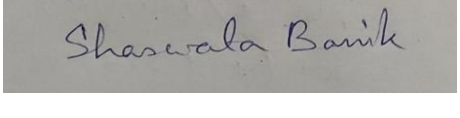
Name: ***Shaswata Banik***

Roll No: ***440***

Supervisor: ***Prof. (Dr.) Surabhi Dasgupta***

<u>Declaration</u>

I, hereby, affirm that I have acknowledged any materials that I've used in my dissertation from outside sources. So, to the best of my knowledge, any parts of my dissertation paper doesn't use any unacknowledged materials.



Shaswata Banilk

Student (3rd year)

Dept. of Statistics

St. Xavier's College (Autonomous), Kolkata

Date: ___/___/2024

Contents:

Introduction:

As, in the present time, there has been a huge impact which is getting more significant by every day, in the effect on economy and socio-economic environment worldwide because of the stock market over few last decades, there has been also significant increase in people interested in it as a researcher, corporate employees, investors, traders, asset managers, stakeholders, policymakers. That's exactly why topics of analyzing stock prices as research purposes have become almost one of the most trending research topics as its insights can be very useful to the understanding of mechanisms of asset prices' movements, important decision making in investments, financial series' behaviors and technical analysis of it which is very important because 88% of millionaires worldwide are self-made and most of their net worth is dependent on investments and also based on several countries' GDPs worldwide, US has over 20% share of global economy as over 50% of global share markets are owned by US and it follows for several developed or 1st World countries as stocks play a very crucial role in their gigantic economies. Generally, these kinds of research contain fitting several kind of Time Series & Financial Time Series models to the data on stock prices and the object of looking into the patten of possible future movements as per the fit. In this project, I have proposed fitting ARIMA (Auto Regressive Integrated Moving Average) and ARCH (Auto Regressive Conditional Heteroscedastic) models on the stock prices (mainly closing & opening stock prices) of Amazon, Microsoft & Netflix over the years.

Data Description:

Here, we will give a short description of the data used in this project for analysis and explain the terminologies present in the data and mainly used in the analysis part in this dissertation.

Θ DATA

Daily data on stock prices on trading days (namely date, opening, high, low, closing, adjusted closing prices & volume) of –

- Amazon (Jan,1999 - Dec,2022)
- Microsoft (Jan,2000 – Nov,2023)
- Netflix (May,2002 – July,2020)

Θ TERMINOLOGIES USED IN DATA ANALYSIS

- Opening Prices: Price at which trade of the scrip starts at the beginning of the trade session

- Closing Prices: Price at which the scrip has traded at the end of the trade session.
- Adjusted Closing Prices: It's the closing prices after paying dividends to shareholders or splitting shares.
- Trading Days: The days on which stock markets don't stay closed.

SOURCES OF DATA

I found these datasets from the website Kaggle. The links for accessing these datasets are given below:

- https://www.kaggle.com/datasets/prajwaldongre/microsoft-stock-price2000-2023
- https://www.kaggle.com/datasets/aayushmishra1512/netflix-stock-data
- https://www.kaggle.com/datasets/sriharshaeedala/amazon-stock-price-from-1999-to-2022

Methodology

Logarithmic returns are important in finance because they provide a more accurate measure of the percentage change in the value of an asset over a period of time. This is particularly important when analyzing financial data because the compounding effect of returns over time can have a significant impact on the value of an asset. Logarithmic returns are also useful because they are additive.

Here, we'll fit ARIMA & ARCH model (if it can be) to the each set of opening and closing stock prices of each companies, namely Amazon, Microsoft & Netflix.

We'll divide the total dataset in the ratio of 8:2 for using as the training data and testing data respectively.

ARIMA(p,d,q) Fitting

In the ARIMA (Auto Regressive Integrated Moving Average) model, the three parameters p, d, q stand for the order of auto-regression, order of differencing and order of moving average respectively.

Here, for each training dataset of log returns of opening and closing stock prices of each companies, we'll follow some steps:

1. Here, we'll use Augmented Dickey-Fuller Test to check for the non-stationarity in for each set of datapoints. In this test, alternative hypothesis stands for the presence of non-stationarity. So, if the test rejected for set of datapoints, then there is non-stationarity present.
   We'll go on with the differencing of the datasets' values until for which the test results in the acceptance of the null hypothesis i.e. the datapoints are stationary.

2. We'll take the order at which the stationarity is obtained as the parameter of order of differencing in the ARIMA model i.e. d.

3. Then we'll plot PACF (Partial Auto Correlation Function) on the true set of datapoints and take the last lag at which the spike is significantly outside the bandwidths in the plot as the parameter of order of Auto-Regression i.e. p.

4. Then we'll plot ACF (Auto Correlation Function) on the obtained set of stationary datapoints and take the last lag at which the spike is outside the bandwidths in the plot as the parameter of order of Moving Averagae i.e. q.

5. We'll fit the each suitably obtained ARIMA models to each the training datasets and the plot the fitted values and true values together in a plot.

6. Then we'll obtain the forecast values and the upper and lower 95% confidence interval linits between which the true datapoints can lie for the each testing datasets based on the fit.

7. We'll plot these values along with true values for the each testing datasets.

8. Then we'll obtain the error created from forecasting the testing values.

9. Then based on those errors obtained, we'll calculate the measures MAD (Mean Absolute Deviation), MPE (Mean Percentage Error), MAPE (Mean Absolute Percentage Error) for each testing datasets and explain the measures.

❖ ARCH(m) fitting

In the ARCH (Auto Regressive Conditional Heteroscedastic) model, the parameter m stand for the order of auto-regression.

Here, for each training dataset of log returns of opening and closing stock prices of each companies, we'll follow some steps:

1. Here, we'll use Augmented Dickey-Fuller Test to check for the non-stationarity in for each set of datapoints. In this test, alternative hypothesis stands for the presence of non-stationarity. So, if the test rejected for set of datapoints, then there is non-stationarity present.
   We'll go on with the differencing of the datasets' values until for which the test results in the acceptance of the null hypothesis i.e. the datapoints are stationary.

2. After that, we'll plot the obtained set of stationary datapoints to check if volatility clustering is present in the data or not

3. Then we'll plot PACF (Partial Auto Correlation Function) on the true set of datapoints and take the last lag at which the spike is significantly outside the bandwidths in the plot as the parameter of order of Auto-Regression i.e. m.

4. Then we'll plot ACF (Auto Correlation Function) on the obtained set of stationary datapoints and take the last lag at which the spike is outside the bandwidths in the plot as the parameter of order of Moving Averagae in ARMA model taken as the mean equation of the log returns series of the stock prices.

5. **Testing for ARCH effects:**
   I. After that, we'll fit suitably obtained ARMA model to the training datasets and obtain the residuals of the fit which will be taken as the shock (a_t) values in the ARCH(m) model

   II. Consequently, we'll fit a linear regression equation of the squared shock values (a_t^2) on the previous m squared shock's lag values ($a_{t-1}^2, a_{t-2}^2, \ldots a_{t-m}^2$) i.e. $a_t^2 = \alpha_0 + \alpha_1 a_{t-1}^2 + \cdots + \alpha_m a_{t-m}^2 + e_t \,; t = m+1, .., T$
   where,
   e_t = error values
   T is the sample size which is here the size of the training data.
   m = some pre-specified integer which is taken as the order obtained from the ACF plots.

   III. Then we'll perform ARCH Lagrange's Multiplier Test which tests if the parameters obtained in the linear regression equation of squared shock values, are all zero (taken in null hypothesis) and takes alternative hypothesis as not the null hypothesis i.e. $H_0\colon \alpha_1 = \alpha_2 = \cdots = \alpha_m \;\; ag. \; H_1\colon not\ H_0$

   IV. Then we'll define some terms
      a) $\bar{w} = \frac{1}{T}\sum_{t=1}^{T} a_t^2$ where $\bar{w}$ is the sample mean of squared shock values.
      b) $SSR_0 = \sum_{t=m+1}^{T}(a_t^2 - \bar{w})^2$
      c) $SSR_1 = \sum_{t=m+1}^{T} \hat{e_t}^2$ where $\hat{e_t}$ is the residual from the prior linear regression equation.
      d) $F = \frac{(SSR_0 - SSR_1)/m}{SSR_1/(T-2m-1)}$ which asymptotically follows χ_m^2 distribution under null hypothesis.

   V. Now, we'll reject H_0 at $\alpha = 0.05$ level of significance if the observed value of F ,based on the training data, is greater than $\chi_{m;\alpha}^2$ where $\chi_{m;\alpha}^2$ is the upper α point of χ_m^2 distribution.

   VI. If it's rejected, we'll conclude that there ARCH effect exists in the data and proceed for the ARCH model fitting and plottin after forecasting.

6. Considering the test for ARCH effect is rejected, we'll fit the ARCH(m), equivalently GARCH(m,0) model to the data and produce the plot of series with 2 conditional SD superimposed to look at the how well the volatility is within the interval of two SDs.

7. After that, we'll find out forecasted volatility series and plot them to how they're varying over the time points.

Results & Discussion

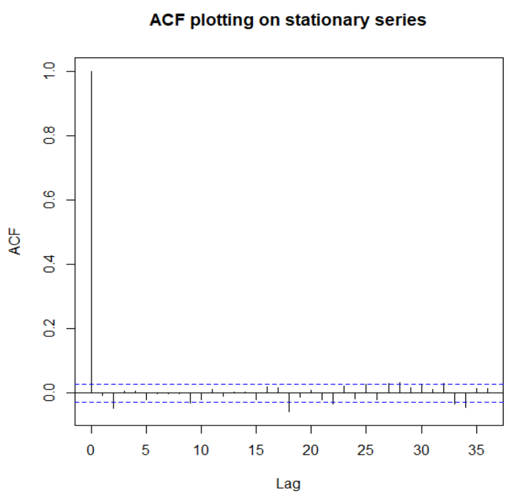
- **ANALYSIS ON LOG RETURNS OF CLOSING PRICES OF AMAZON**
- *Stationarity checking:*

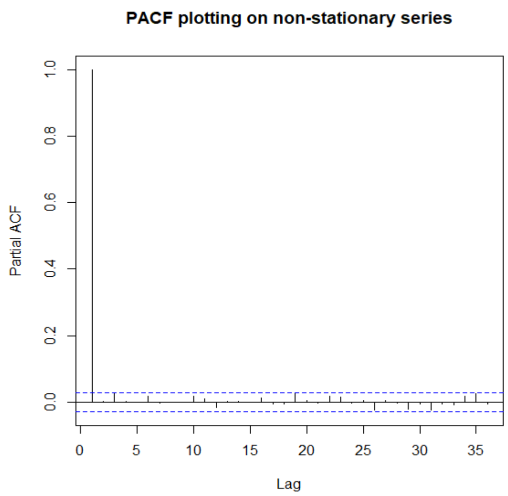
The p-value in the Augmented Dickey-Fuller Test performed

- Based on the true values: 0.2751
- Based on the values after 1st order differencing: 0.01

Implication: Stationarity is obtained after 1st order differencing.So d is taken to be 1.

- *ACF and PACF plots:*





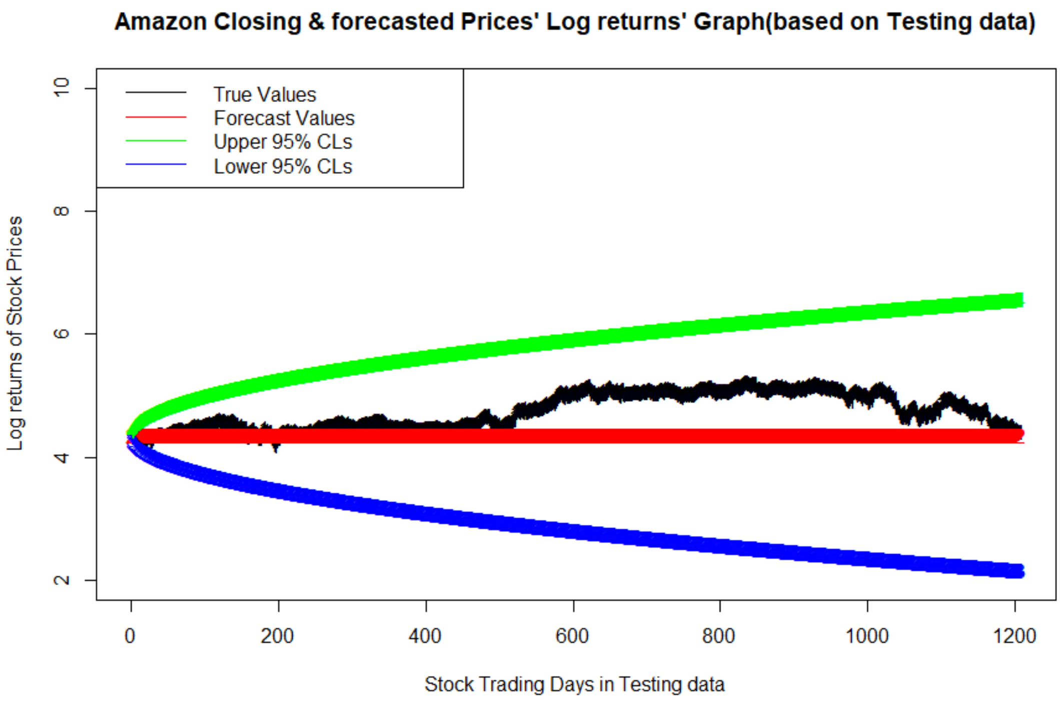
Implication: The ARIMA & ARMA (for the mean equation) parameters p and q are taken as 1 and 2 respectively. ARCH parameter m is taken as 1.

ARIMA Fitting:

- *Fitting ARIMA(1,1,2) on training data and plotting the fitted and true values together:*

The fitted model is given by $(\hat{Y}_t - Y_{t-1}) = 0.6722(Y_{t-1}-Y_{t-2}) + e_t - 0.6787(e_{t-1}) - 0.0237(e_{t-2})$ where Y_t is the Amazon's. closing stock prices' log returns at t.

- *Plotting the forecasted, upper & lower 95% CLs' values along with true values of testing data:*



- *Calculating MAD, MPE & MAPE based on the error obtained from forecasted values:*

i) MAD = 0.396825

ii) MPE = 7.891895

iii) MAPE = 8.010231

Implication:

Here, by fitting ARIMA(1,1,2) model on the training data of log returns of closing stock prices of Amazon, from the measures of the error obtained from forecasting,

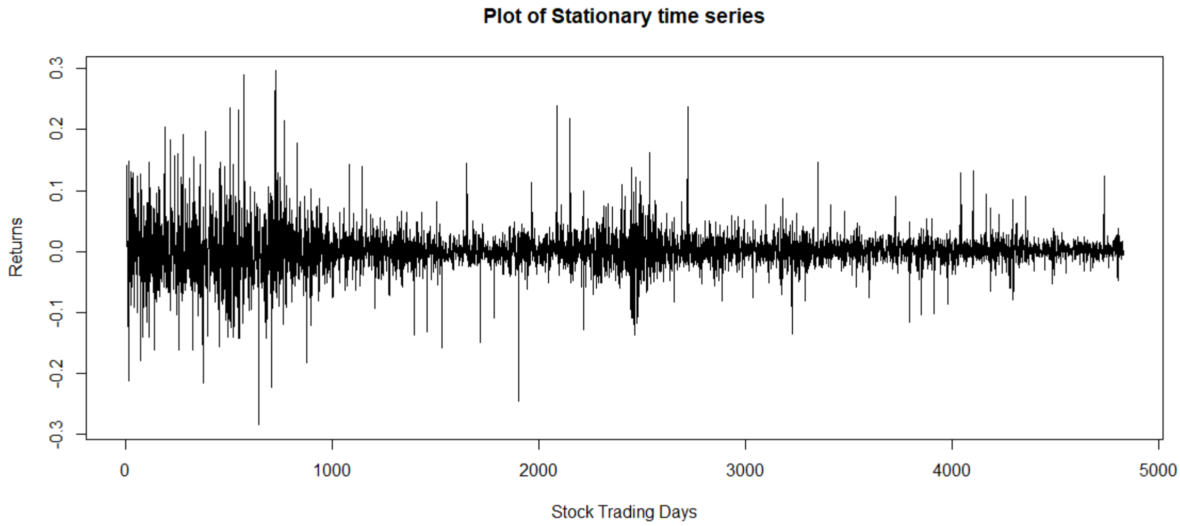
i) The forecasted values is on an average deviated from the real testing values in the absolute magnitude of 0.396825 units.

ii) The average percentage deviation of the forecasted values from the real testing values is of 7.891895.

iii) The average percentage absolute deviation of the forecasted values from the real testing values is of 8.010231.

❖ ARCH fitting

- *Checking for volatility clustering:*



Implication: Volatility Clustering is present here as large changes tend to be followed by large changes and small changes are followed by small changes

- *Fitting ARMA(1,2) as the mean equation of the log return series of stock prices:*

The model is given by, $r_t = \mu_t + a_t$
where,
μ_t is the mean equation at time point t & $\mu_t = 0.0004696 + 1.0001198r_{t-1} - 0.0081412a_{t-1} - 0.0479338a_{t-2}$
r_t is the log return of the stock prices at time point t
a_t is the shock values at time point t

- *Fitting the linear regression equation of squared shock value on its m(=1) previous squared lag values:*

The equation is given by, $a_t^2 = 0.001271 - 0.002512a_{t-1}^2$; t=2,…,4831

- *Performing ARCH LM test for checking for ARCH effect:*

$SSR_0 = 0.09420089$ where $\overline{w} = 0.00127284$
$SSR_1 = 0.09407743$
F_obs = 6.336176 & $\chi^2_{m;\alpha}$ = 3.841459 when α=0.05

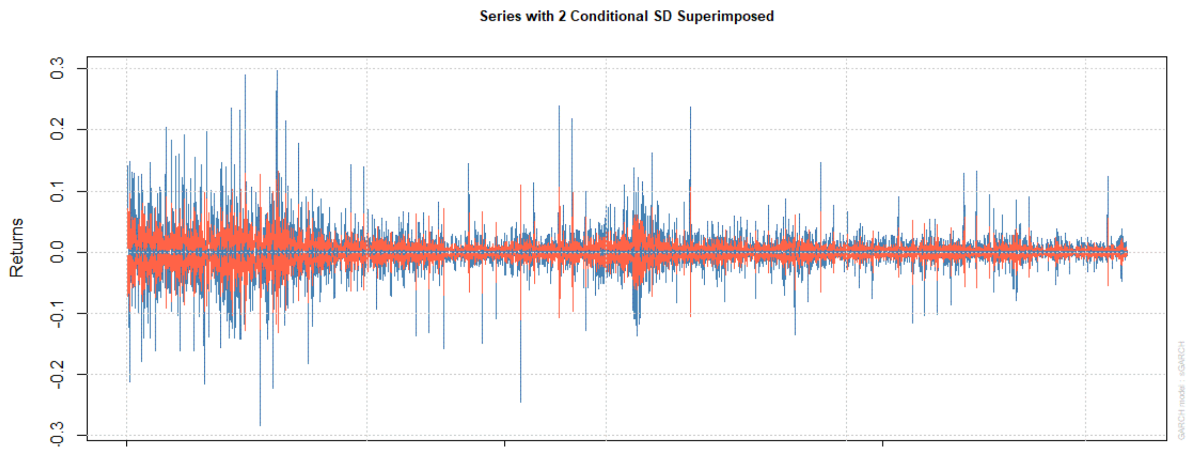
Implication: Here, the null hypothesis is getting rejected at α level of significance as F_obs > $\chi^2_{m;\alpha}$. So, there is ARCH effect present in the data.

- *Fitting ARCH(1) model to the training data and looking the significance of optimal parameters:*

```
Optimal Parameters
------------------------------------
        Estimate  Std. Error   t value Pr(>|t|)
mu      0.000599    0.000000  2400.178        0
ar1     0.666368    0.000185  3606.245        0
ma1    -0.675122    0.000176 -3829.235        0
ma2    -0.024031    0.000024  -990.482        0
omega   0.000001    0.000000    29.309        0
alpha1  0.050000    0.000002 21859.608        0
```

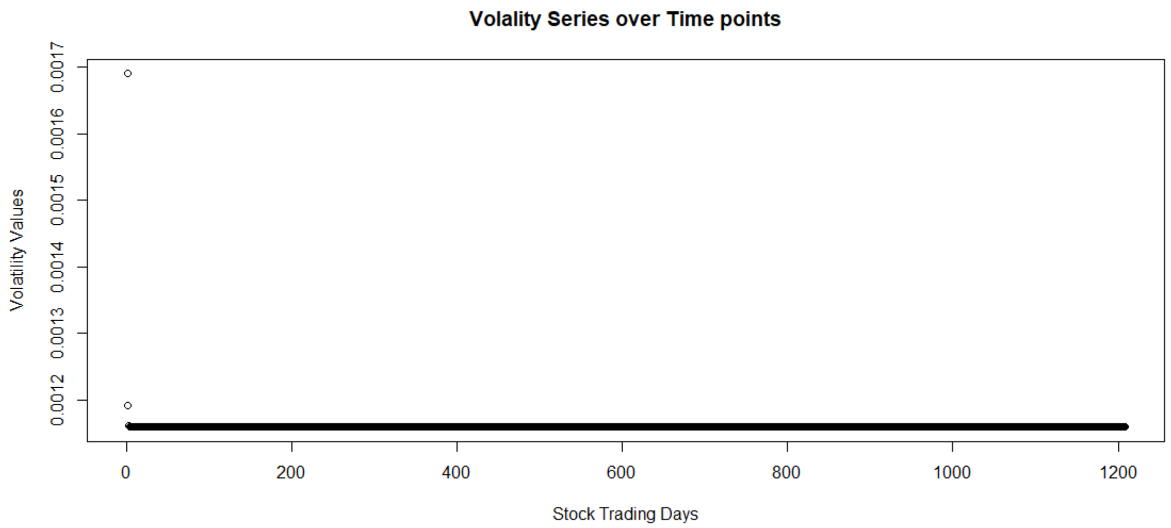
Implication: By looking at the p-values for each parameters, we can see that every one of them is lower than 0.05, so we can say that all the parameters here are significant.

- *Plotting the series with two conditional SDs superimposed:*



Implication: Here we can see that, the most of volatility component is out of the interval formed two SDs.

- *Plotting the volatility series over the time points:*



Implication: When we run the forecast of the volatility of the testing data values, we can see that, the volatility of stock prices decrease for the next 7 days and then stays at the same level for the rest of the part.

➢ ANALYSIS ON LOG RETURNS OF OPENING PRICES OF AMAZON

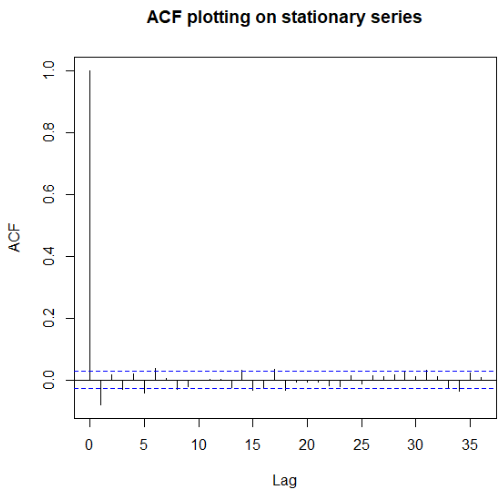
- *Stationarity checking:*

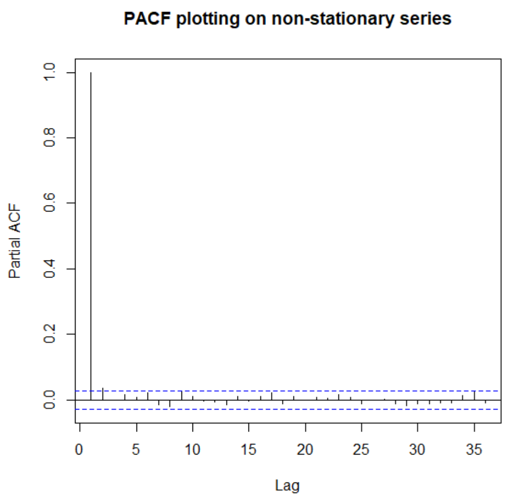
The p-value in the Augmented Dickey-Fuller Test performed

- Based on the true values: 0.2751
- Based on the values after 1st order differencing: 0.01

Implication: Stationarity is obtained after 1st order differencing.So d is taken to be 1.

- *ACF and PACF plots:*



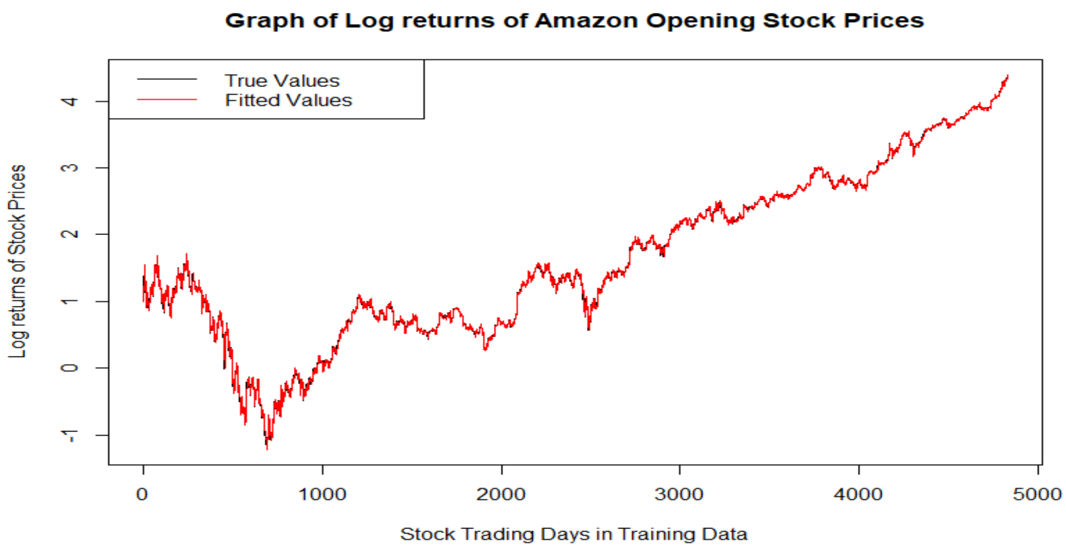


Implication: The ARIMA parameters p,q are taken as 1,1 & ARMA (for the mean eqn.) parameters p, q are taken as 1,0 respectively. ARCH parameter m is taken as 1.

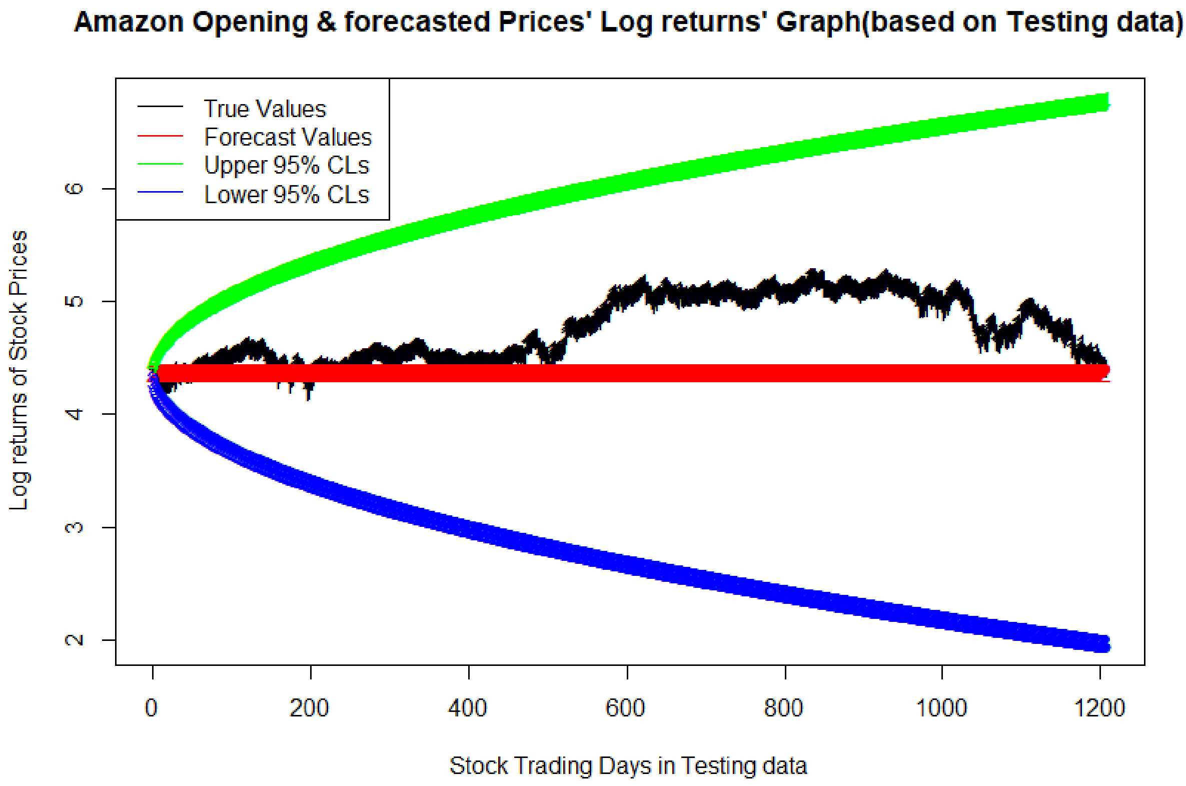
❖ ***ARIMA Fitting:***

- *Fitting ARIMA(1,1,1) on training data and plotting the fitted and true values together:*

The fitted model is given by $(\hat{Y}_t - Y_{t-1}) = -0.6922(Y_{t-1}-Y_{t-2}) + e_t + 0.6304(e_{t-1})$ where Y_t is the Amazon's. opening stock prices' log returns at t.



- *Plotting the forecasted, upper & lower 95% CLs' values along with true values of testing data:*



- *Calculating MAD, MPE & MAPE based on the error obtained from forecasted values:*

i) MAD = 0.389801

ii) MPE = 7.726643

iii) MAPE = 7.862295

Implication:

Here, by fitting ARIMA(1,1,1) model on the training data of log returns of closing stock prices of Amazon, from the measures of the error obtained from forecasting,

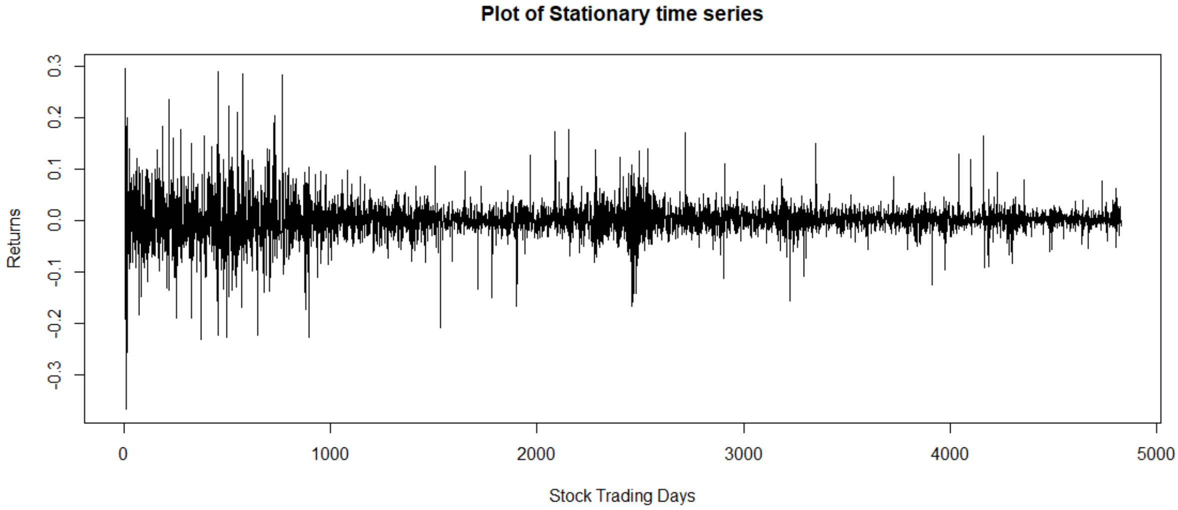
i) The forecasted values is on an average deviated from the real testing values in the absolute magnitude of 0.389801 units.

ii) The average percentage deviation of the forecasted values from the real testing values is of 7.726643.

iii) The average percentage absolute deviation of the forecasted values from the real testing values is of 7.862295.

❖ ARCH fitting

- *Checking for volatility clustering:*



Implication: Volatility Clustering is present here as large changes tend to be followed by large changes and small changes are followed by small changes

- *Fitting ARMA(1,0) as the mean equation of the log return series of stock prices:*

The model is given by, $r_t = \mu_t + a_t$
where,

μ_t is the mean equation at time point t & $\mu_t = 0.0006239 + 1.0000467 r_{t-1}$
r_t is the log return of the stock prices at time point t

- *Fitting the linear regression equation of squared shock value on its m(=1) previous squared lag values:*

The equation is given by, $a_t^2 = 0.001351 - 0.008867 a_{t-1}^2$; t=2,…,4831

- *Performing ARCH LM test for checking for ARCH effect:*

SSR_0 = 0.1213478 where $\bar{w} = 0.001351208$
SSR_1 = 0.1208329
F_obs = 20.5731 & $\chi^2_{m;\alpha}$ = 3.841459 when α=0.05

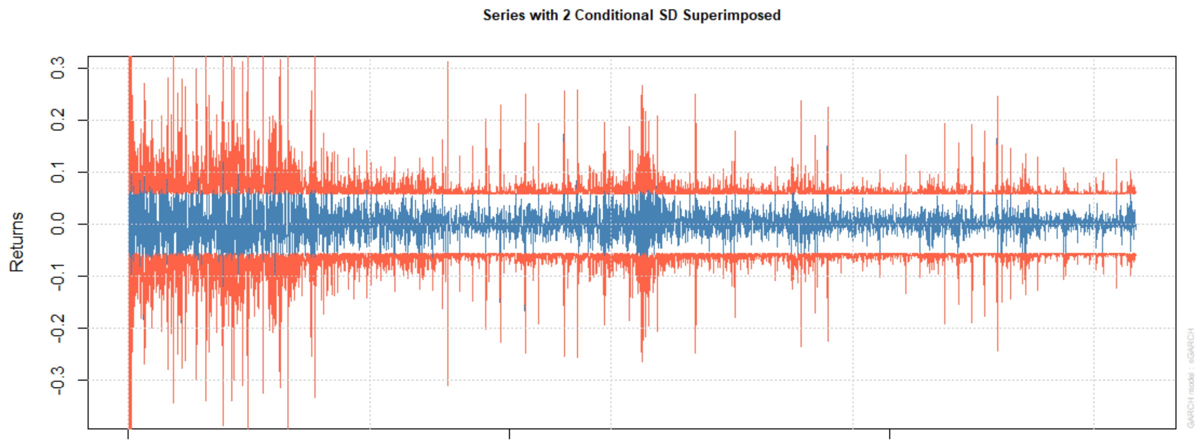
Implication: Here, the null hypothesis is getting rejected at α level of significance as F_obs > $\chi^2_{m;\alpha}$. So, there is ARCH effect present in the data.

- *Fitting ARCH(1) model to the training data and looking the significance of optimal parameters:*

```
Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.001018    0.000398   2.5572 0.010552
ar1    -0.112570    0.019506  -5.7711 0.000000
omega   0.000824    0.000024  33.8746 0.000000
alpha1  0.528625    0.041637  12.6962 0.000000
```

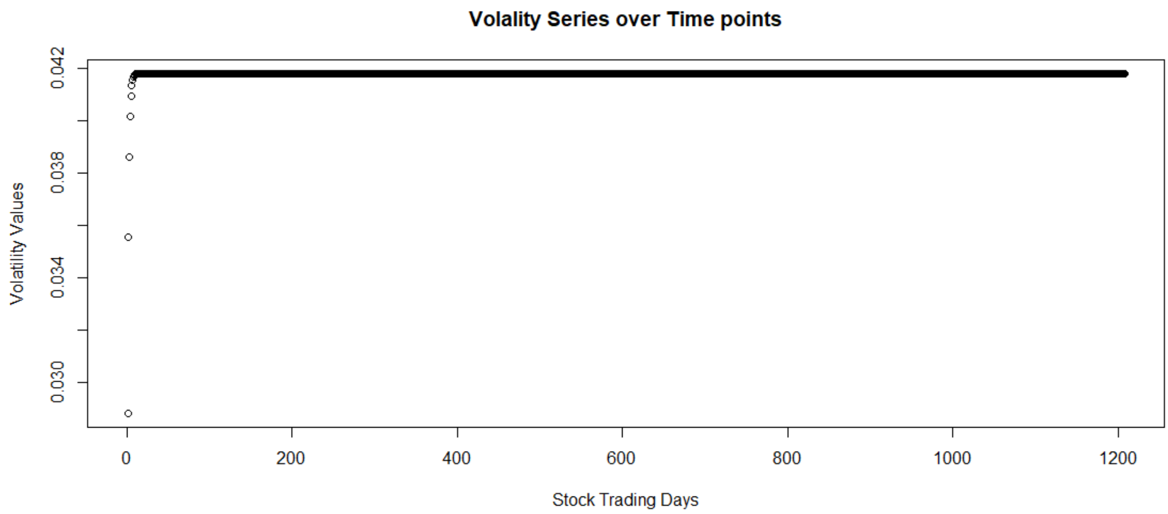
Implication: By looking at the p-values for each parameters, we can see that every one of them except the parameter μ in the ARMA model is lower than 0.05, so we can say that all the parameters here are significant except μ in the ARMA model.

- *Plotting the series with two conditional SDs superimposed:*



Implication: Here we can see that, the most of volatility component is inside the interval formed by two conditional SDs

- *Plotting the volatility series over the time points:*



Implication: When we run the forecast of the volatility of the testing data values, we can see that, the volatility of stock prices increase for the next 13 days and then stays at the same level for the rest of the part.

➢ ANALYSIS ON LOG RETURNS OF CLOSING PRICES OF MICROSOFT

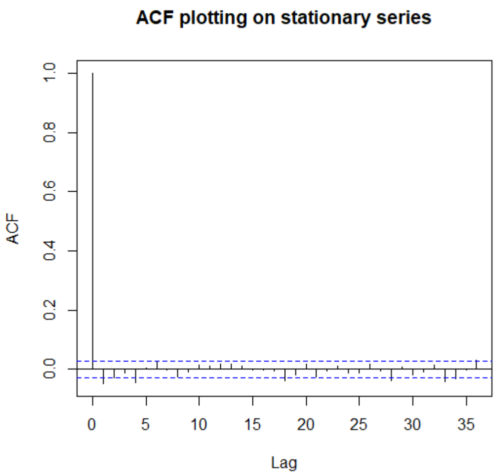
- *Stationarity checking:*

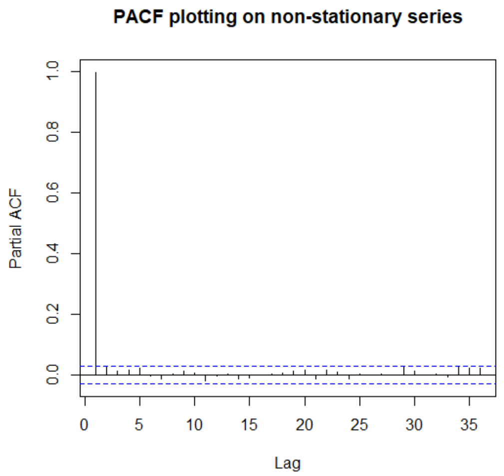
The p-value in the Augmented Dickey-Fuller Test performed

- Based on the true values: 0.4087
- Based on the values after 1st order differencing: 0.01

Implication: Stationarity is obtained after 1st order differencing.So d is taken to be 1.

- *ACF and PACF plots:*



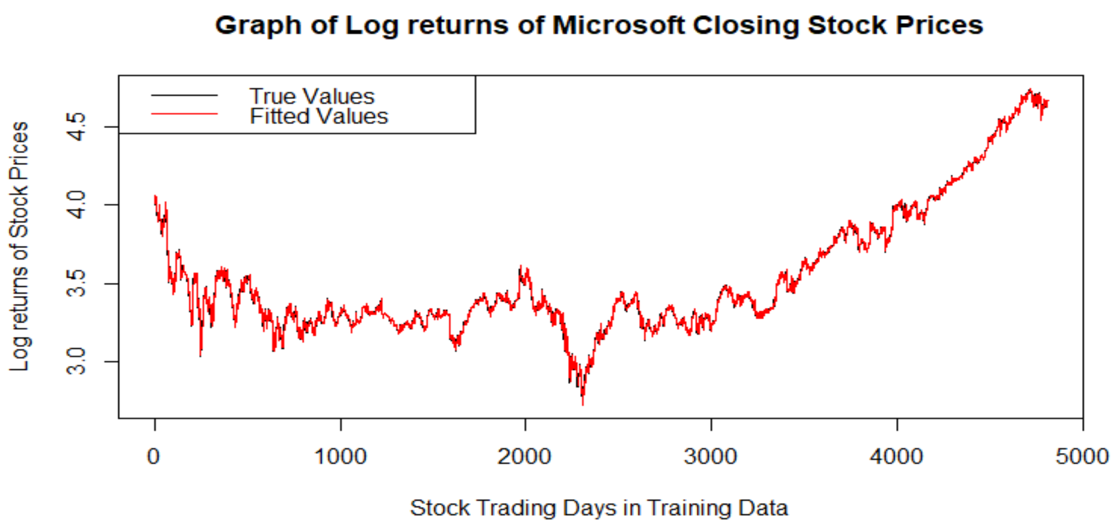


Implication: The ARIMA parameters p,q are taken as 1,1 & ARMA (for the mean eqn.) parameters p, q are taken as 1,0 respectively. ARCH parameter m is taken as 1.

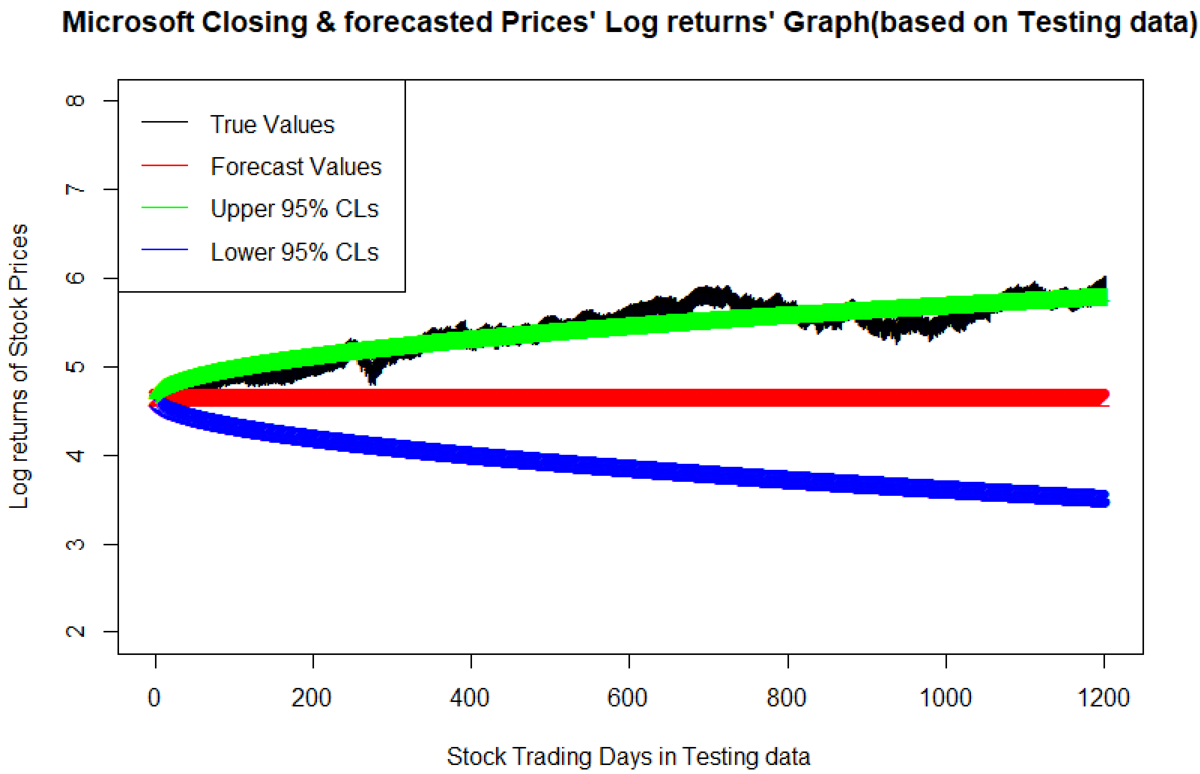
❖ *ARIMA Fitting:*

- *Fitting ARIMA(1,1,1) on training data and plotting the fitted and true values together:*

The fitted model is given by $(\widetilde{Y}_t - Y_{t-1}) = -0.5868(Y_{t-1}-Y_{t-2}) + e_t - 0.6399(e_{t-1})$ where Y_t is the Microsoft's. closing stock prices' log returns at t.



- *Plotting the forecasted, upper & lower 95% CLs' values along with true values of testing data:*



Implication: From this graph, we can see that there may be seasonality present in the true value of training data and that's why it's going outside the interval as we have not considered seasonality in our fitted ARIMA model

- *Calculating MAD, MPE & MAPE based on the error obtained from forecasted values:*

i) MAD = 0.7491078

ii) MPE = 13.51254

iii) MAPE = 13.51254

Implication:

Here, by fitting ARIMA(1,1,1) model on the training data of log returns of closing stock prices of Amazon, from the measures of the error obtained from forecasting,

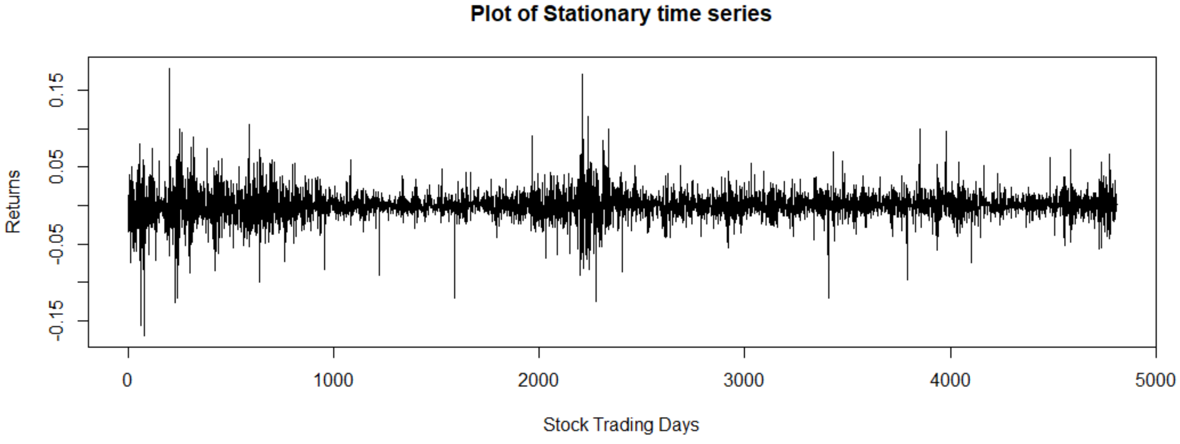
i) The forecasted values is on an average deviated from the real testing values in the absolute magnitude of 0.7491078 units.

ii) The average percentage deviation of the forecasted values from the real testing values is of 13.51254.

iii) The average percentage absolute deviation of the forecasted values from the real testing values is of 13.51254 which is coming to be equal to MPE as here all the true values of data lie above the forecasted values.

❖ ARCH fitting

- *Checking for volatility clustering:*



Implication: Volatility Clustering is present here as large changes tend to be followed by large changes and small changes are followed by small changes

- *Fitting ARMA(1,0) as the mean equation of the log return series of stock prices:*

The model is given by, $r_t = \mu_t + a_t$
where,

μ_t is the mean equation at time point t & $\mu_t = 0.001871 + 0.999507r_{t-1}$
r_t is the log return of the stock prices at time point t

- *Fitting the linear regression equation of squared shock value on its m(=1) previous squared lag values:*

The equation is given by, $a_t^2 = 0.0003721 - 0.0007639a_{t-1}^2$; t=2,…,4810

- *Performing ARCH LM test for checking for ARCH effect:*

$SSR_0 = 0.007727359$ where $\overline{w} = 0.0003722668$
$SSR_1 = 0.007725675$
F_obs = 1.047799 & $\chi^2_{m;\alpha} = 3.841459$ when α=0.05

Implication: Here, the null hypothesis is getting accepted at α level of significance as F_obs < $\chi^2_{m;\alpha}$. So, there is no ARCH effect present in the data. So, it's unnecessary to fit & forecast through ARCH model here.

➢ ANALYSIS ON LOG RETURNS OF OPENING PRICES OF MICROSOFT

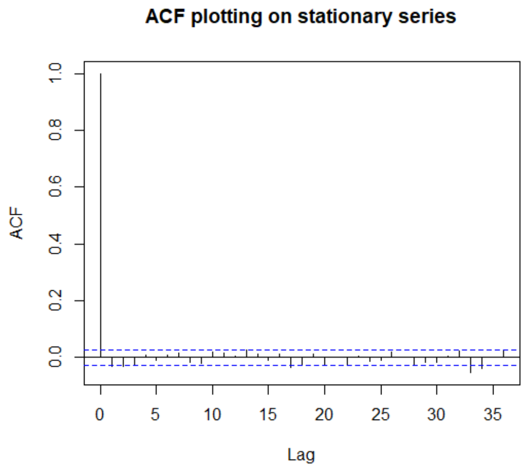
- *Stationarity checking:*

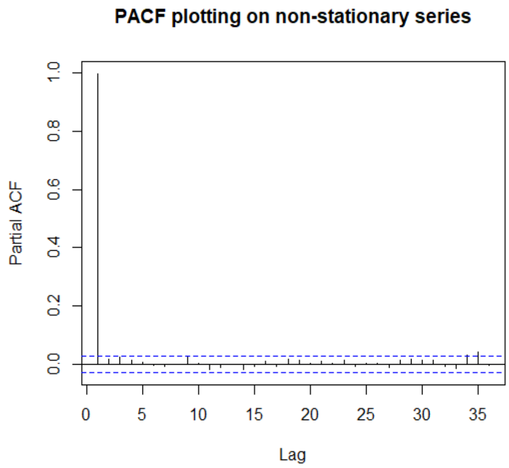
The p-value in the Augmented Dickey-Fuller Test performed

- Based on the true values: 0.372
- Based on the values after 1st order differencing: 0.01

Implication: Stationarity is obtained after 1st order differencing.So d is taken to be 1.

- *ACF and PACF plots:*



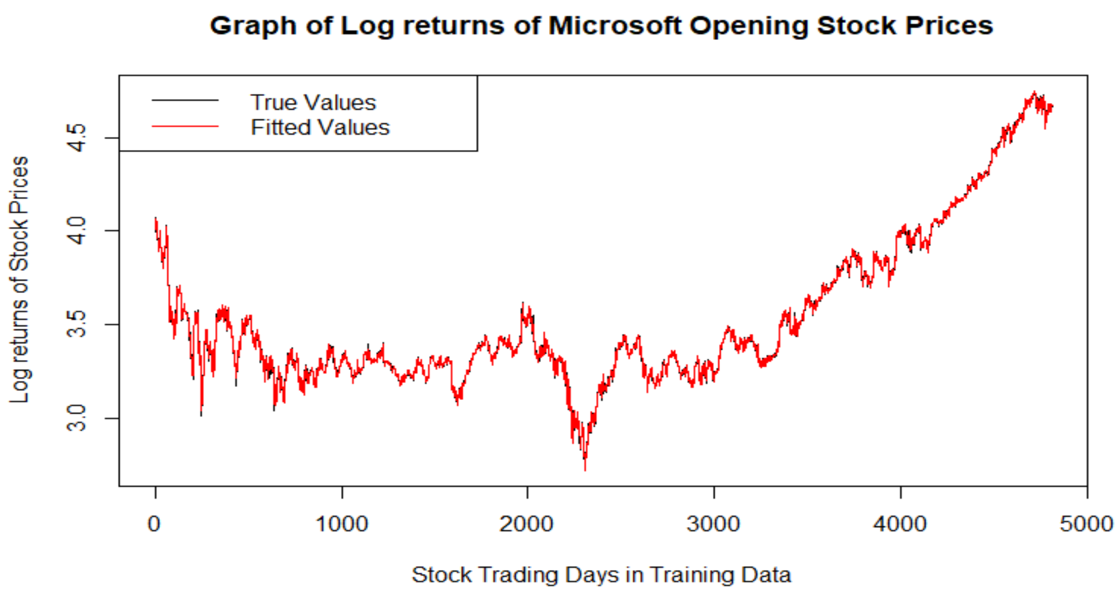


Implication: The ARIMA parameters p,q are taken as 1,4 & ARMA (for the mean eqn.) parameters p, q are taken as 1,0 respectively. ARCH parameter m is taken as 1.

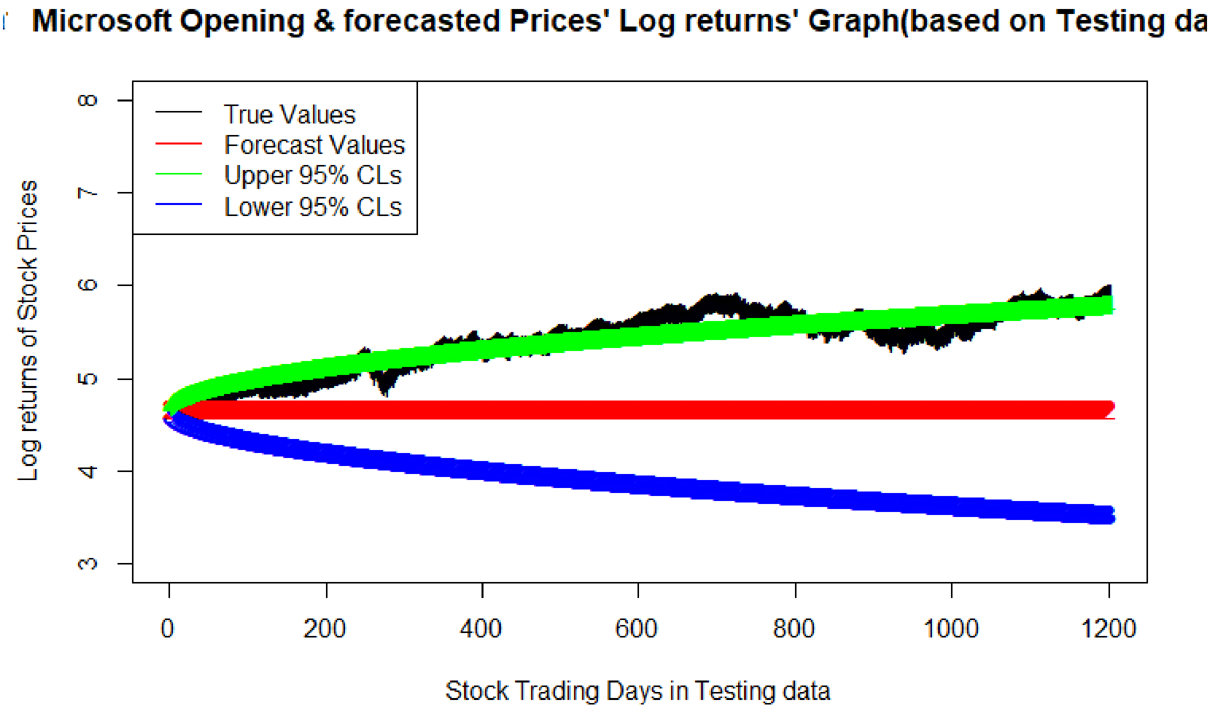
❖ <u>*ARIMA Fitting:*</u>

- *Fitting ARIMA(1,1,4) on training data and plotting the fitted and true values together:*

The fitted model is given by $(\hat{Y}_t - Y_{t-1}) = -0.1208(Y_{t-1}-Y_{t-2})+e_t+0.0701(e_{t-1})-0.0341(e_{t-2}) - 0.0175(e_{t-3}) -0.0479(e_{t-4})$ where Y_t is the Microsoft's. opening stock prices' log returns at t.



- *Plotting the forecasted, upper & lower 95% CLs' values along with true values of testing data:*



Implication: From this graph, we can see that there may be seasonality present in the true value of training data and that's why it's going outside the interval as we have not considered seasonality in our fitted ARIMA model

- *Calculating MAD, MPE & MAPE based on the error obtained from forecasted values:*

i) MAD = 0.7487667

ii) MPE = 13.50697

iii) MAPE = 13.50697

Implication:

Here, by fitting ARIMA(1,1,1) model on the training data of log returns of closing stock prices of Amazon, from the measures of the error obtained from forecasting,

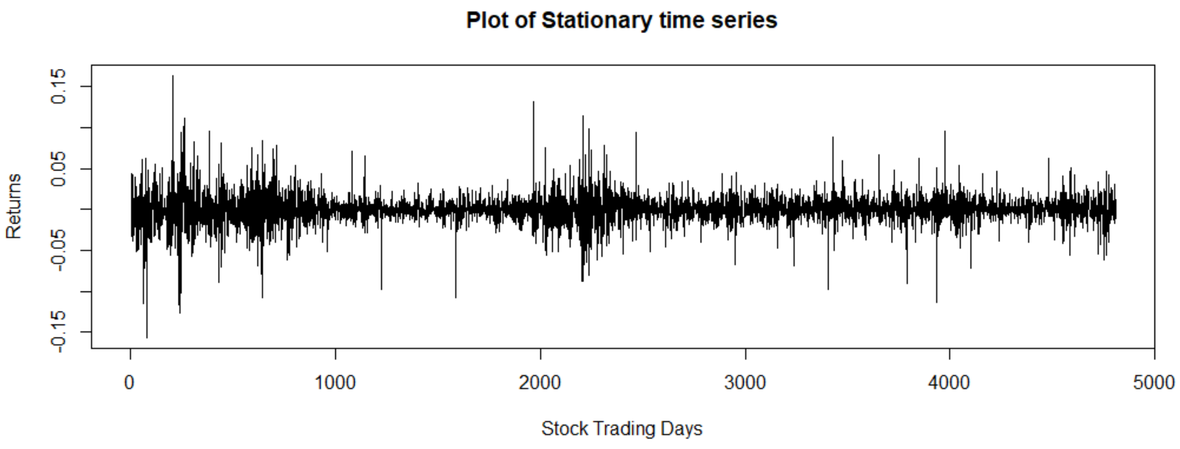
i) The forecasted values is on an average deviated from the real testing values in the absolute magnitude of 0.7487667units.

ii) The average percentage deviation of the forecasted values from the real testing values is of 13.50697.

iii) The average percentage absolute deviation of the forecasted values from the real testing values is of 13.50697 which is coming to be equal to MPE as here all the true values of data lie above the forecasted values.

❖ *ARCH fitting*

- *Checking for volatility clustering:*



Implication: Volatility Clustering is present here as large changes tend to be followed by large changes and small changes tend to be followed by small changes

- *Fitting ARMA(1,0) as the mean equation of the log return series of stock prices:*

The model is given by, $r_t = \mu_t + a_t$
where,

μ_t is the mean equation at time point t & $\mu_t = 0.001593 + 0.999585\, r_{t-1}$
r_t is the log return of the stock prices at time point t

- *Fitting the linear regression equation of squared shock value on its m(=1) previous squared lag values:*

The equation is given by, $a_t^2 = 0.0003444 - 0.0032085 a_{t-1}^2$; t=2,…,4810

- *Performing ARCH LM test for checking for ARCH effect:*

SSR_0 = 0.005744908 where $\overline{w} = 0.000344526$
SSR_1 = 0.005727307
F_obs = 14.77352 & $\chi^2_{m;\alpha}$ = 3.841459 when α=0.05

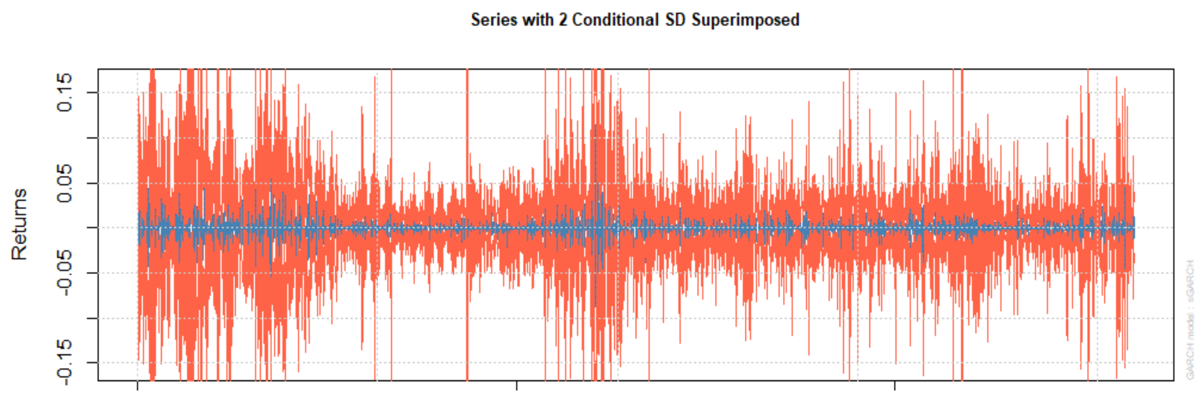
Implication: Here, the null hypothesis is getting rejected at α level of significance as F_obs > $\chi^2_{m;\alpha}$. So, there is ARCH effect present in the data.

- *Fitting ARCH(1) model to the training data and looking the significance of optimal parameters:*

```
Optimal Parameters
------------------------------------
        Estimate  Std. Error    t value Pr(>|t|)
mu     -0.012339    0.000002 -5905.9659    0e+00
ar1     0.884460    0.000402  2199.7013    0e+00
omega   0.000001    0.000000     4.9164    1e-06
alpha1  0.998350    0.000244  4085.3450    0e+00
```

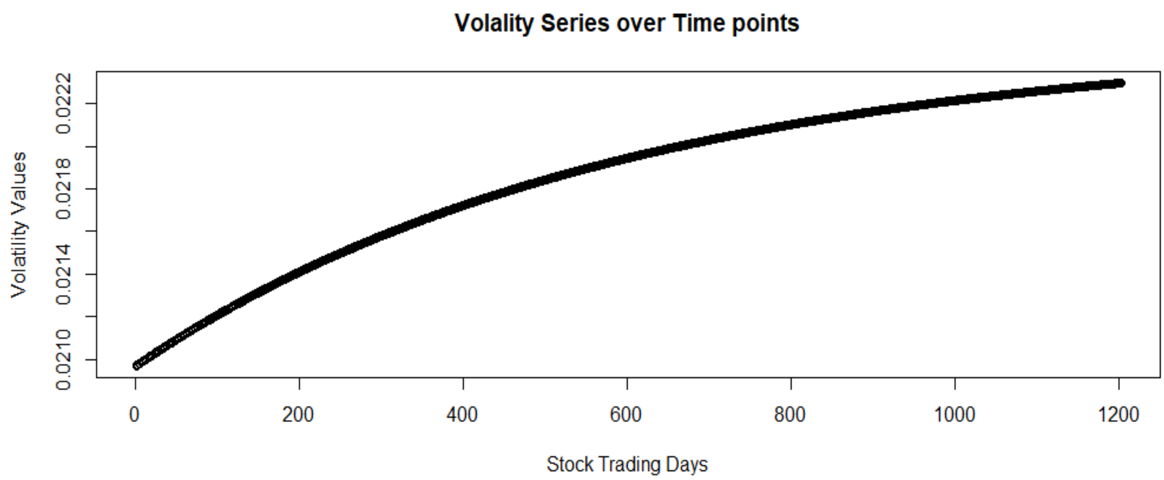
Implication: By looking at the p-values for each parameters, we can see that only one of them i.e. omega(alpha0) in the ARMA model is lower than 0.05, so we can say that only aforementioned parameters here is significant except μ in the ARMA model.

- *Plotting the series with two conditional SDs superimposed:*



Implication: Here we can see that, the most of volatility component is inside the interval formed by two conditional SDs.

- *Plotting the volatility series over the time points:*



Implication: When we run the forecast of the volatility of the testing data values, we can see that, the volatility of stock prices gradually increase over the total forecasting period.

➢ ANALYSIS ON LOG RETURNS OF CLOSING PRICES OF NETFLIX

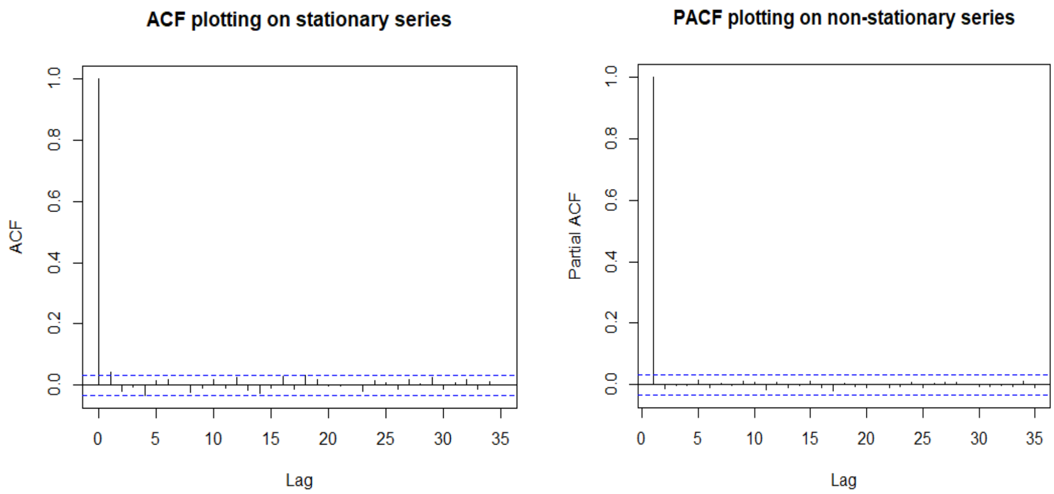
- *Stationarity checking:*

The p-value in the Augmented Dickey-Fuller Test performed

- Based on the true values: 0.3891
- Based on the values after 1st order differencing: 0.01

Implication: Stationarity is obtained after 1st order differencing.So d is taken to be 1.

- *ACF and PACF plots:*

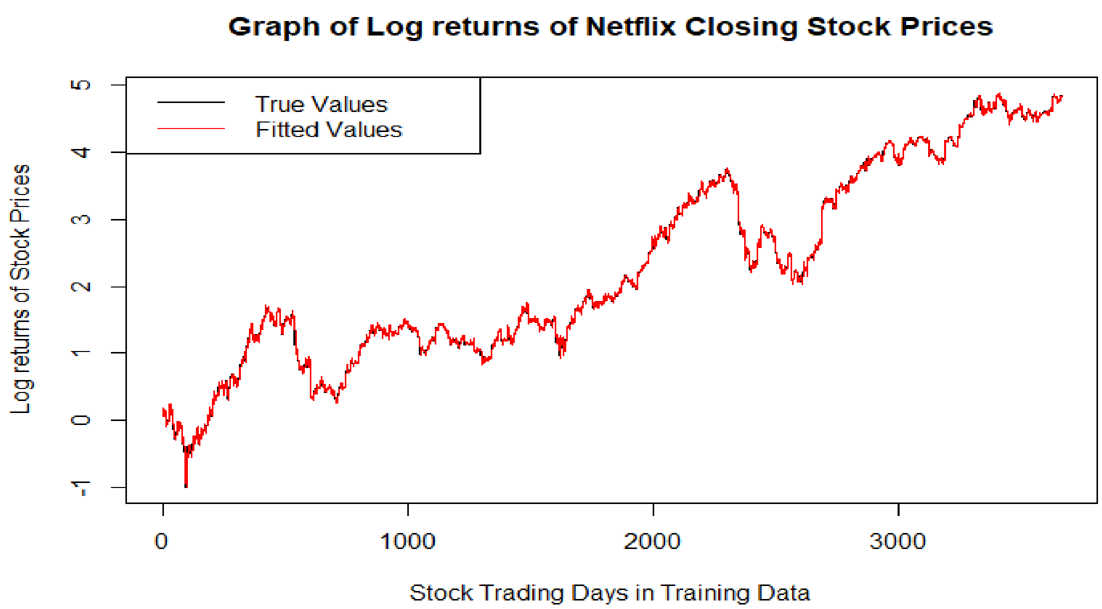


Implication: The ARIMA parameters p,q are taken as 1,0 & ARMA (for the mean eqn.) parameters p, q are taken as 1,0 respectively. ARCH parameter m is taken as 1.

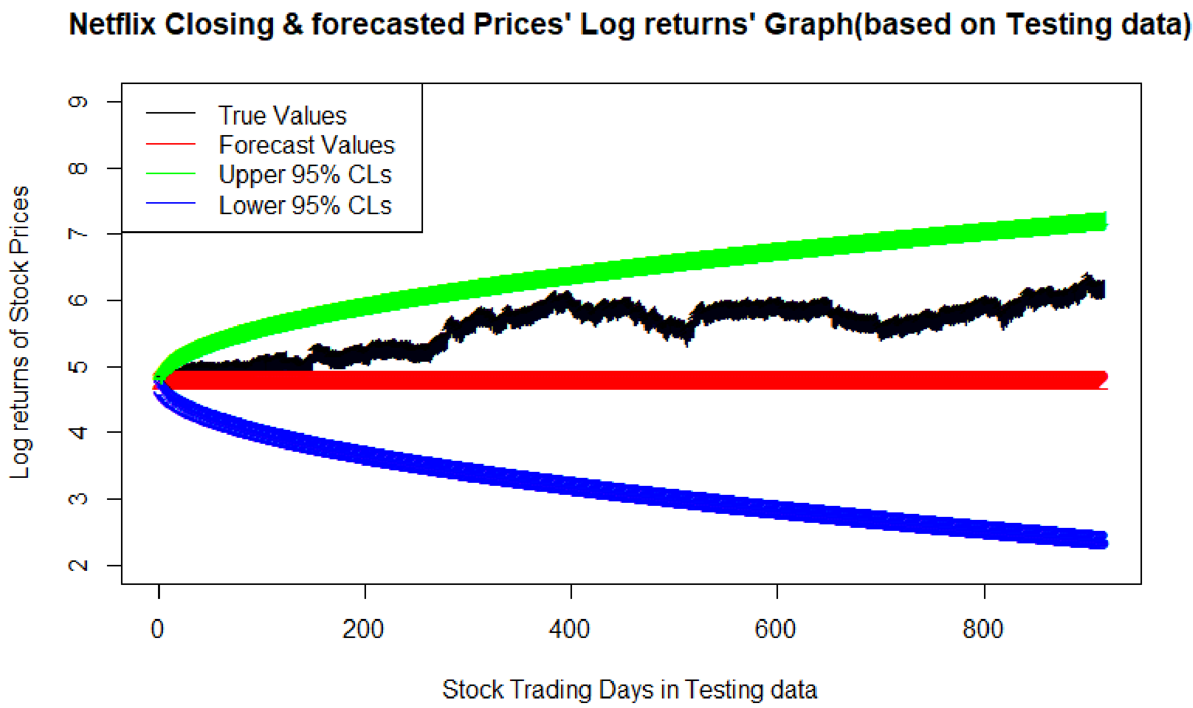
❖ ARIMA Fitting:

- *Fitting ARIMA(1,1,0) on training data and plotting the fitted and true values together:*

The fitted model is given by ($\hat{Y}_t$- Y_{t-1}) = 0.0415(Y_{t-1}-Y_{t-2})+e_t where Y_t is the Netflix's. closing stock prices' log returns at t.



- *Plotting the forecasted, upper & lower 95% CLs' values along with true values of testing data:*



Implication: From this graph, we can see that there may be slight seasonality present in the true values of training data and that's why it's not going outside the interval but it's above the forecasted values as we haven't considered seasonality in fitted ARIMA model

- *Calculating MAD, MPE & MAPE based on the error obtained from forecasted values:*

i) MAD = 0.7990358

ii) MPE = 13.84941

iii) MAPE = 13.84941

Implication:

Here, by fitting ARIMA(1,1,1) model on the training data of log returns of closing stock prices of Amazon, from the measures of the error obtained from forecasting,

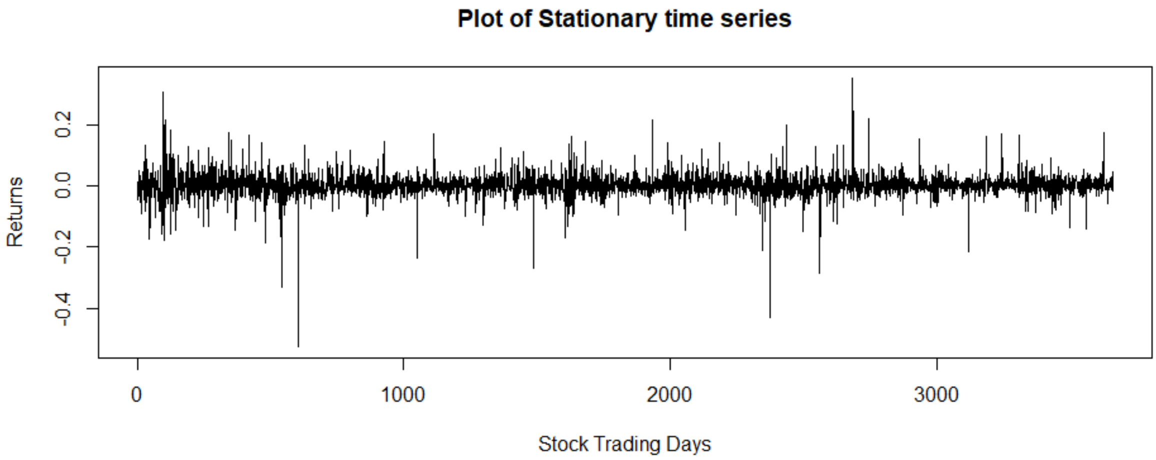
i) The forecasted values is on an average deviated from the real testing values in the absolute magnitude of 0.7990358 units.

ii) The average percentage deviation of the forecasted values from the real testing values is of 13.84941.

iii) The average percentage absolute deviation of the forecasted values from the real testing values is of 13.84941 which is coming to be equal to MPE as here all the true values of data lie above the forecasted values.

❖ *ARCH fitting*

- *Checking for volatility clustering:*



Implication: Volatility Clustering is present here as large changes tend to be followed by large changes and small changes tend to be followed by small changes.

- *Fitting ARMA(1,0) as the mean equation of the log return series of stock prices:*

The model is given by, $r_t = \mu_t + a_t$
where,

μ_t is the mean equation at time point t & $\mu_t = 0.001785 + 0.999771\ r_{t-1}$
r_t is the log return of the stock prices at time point t

- *Fitting the linear regression equation of squared shock value on its m(=1) previous squared lag values:*

The equation is given by, $a_t^2 = 0.0015196 + 0.0003775 a_{t-1}^2$; t=2,…,3665

- *Performing ARCH LM test for checking for ARCH effect:*

$SSR_0 = 0.2045498$ where $\overline{w} = 0.001519209$
$SSR_1 = 0.2045469$
F_obs = 0.05073404 & $\chi^2_{m;\alpha}$ = 3.841459 when α=0.05

Implication: Here, the null hypothesis is getting accepted at α level of significance as F_obs < $\chi^2_{m;\alpha}$. So, there is no ARCH effect present in the data. So, it's unnecessary to fit & forecast through ARCH model here

ANALYSIS ON LOG RETURNS OF OPENING PRICES OF NETFLIX

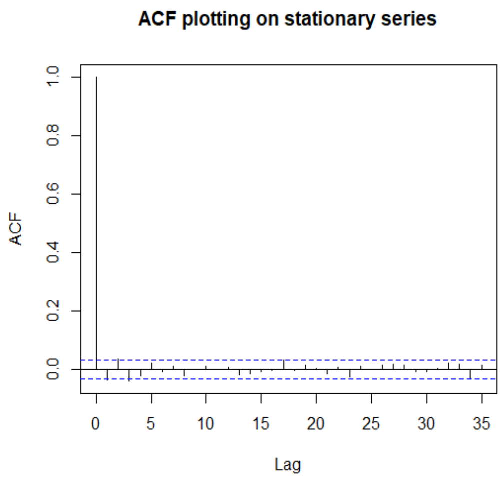
- *Stationarity checking:*

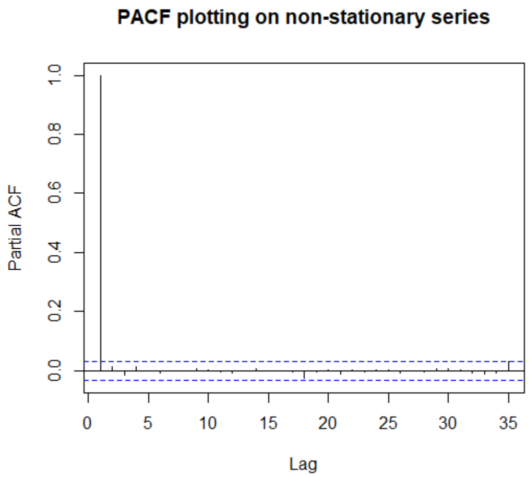
The p-value in the Augmented Dickey-Fuller Test performed

- Based on the true values: 0.3766
- Based on the values after 1st order differencing: 0.01

Implication: Stationarity is obtained after 1st order differencing.So d is taken to be 1.

- *ACF and PACF plots:*



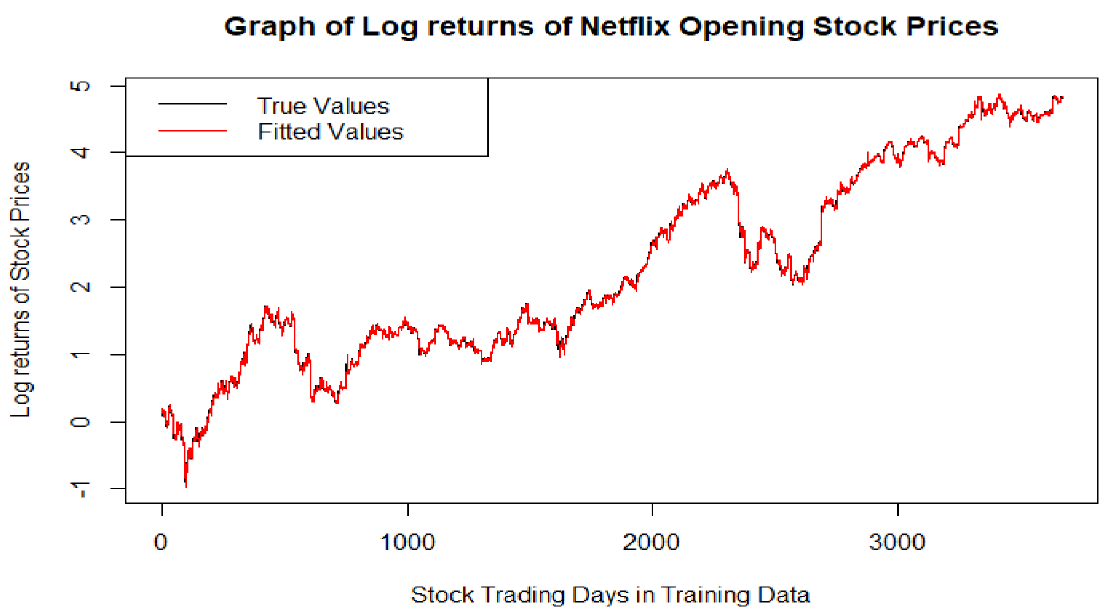


Implication: The ARIMA parameters p,q are taken as 1,2 & ARMA (for the mean eqn.) parameters p, q are taken as 1,0 respectively. ARCH parameter m is taken as 1.

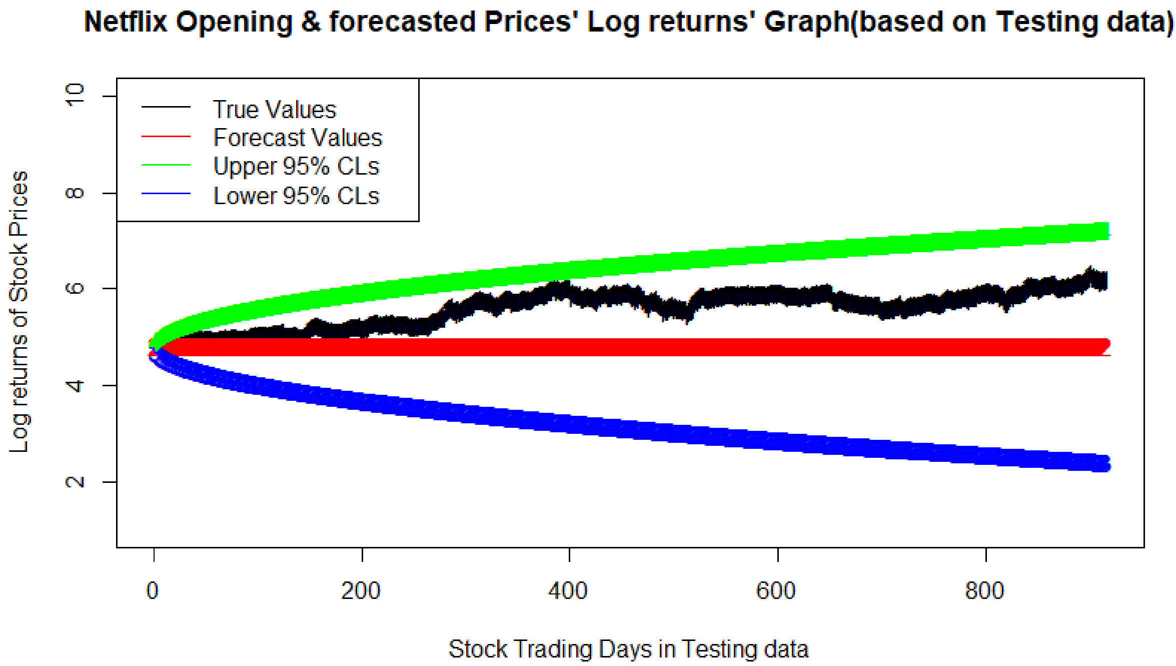
ARIMA Fitting:

- *Fitting ARIMA(1,1,2) on training data and plotting the fitted and true values together:*

The fitted model is given by $(\hat{Y}_t - Y_{t-1}) = -0.3419(Y_{t-1}-Y_{t-2})+e_t+0.3119(e_{t-1})-0.0348(e_{t-2})$ where Y_t is the Netflix's. opening stock prices' log returns at t.



- *Plotting the forecasted, upper & lower 95% CLs' values along with true values of testing data:*



Implication: From this graph, we can see that there may be slight seasonality present in the true values of training data and that's why it's not going outside the interval but it's above the forecasted values as we haven't considered seasonality in fitted ARIMA model

- *Calculating MAD, MPE & MAPE based on the error obtained from forecasted values:*

i) MAD = 0.7914426

ii) MPE = 13.71441

iii) MAPE = 13.84941

Implication:

Here, by fitting ARIMA(1,1,1) model on the training data of log returns of closing stock prices of Amazon, from the measures of the error obtained from forecasting,

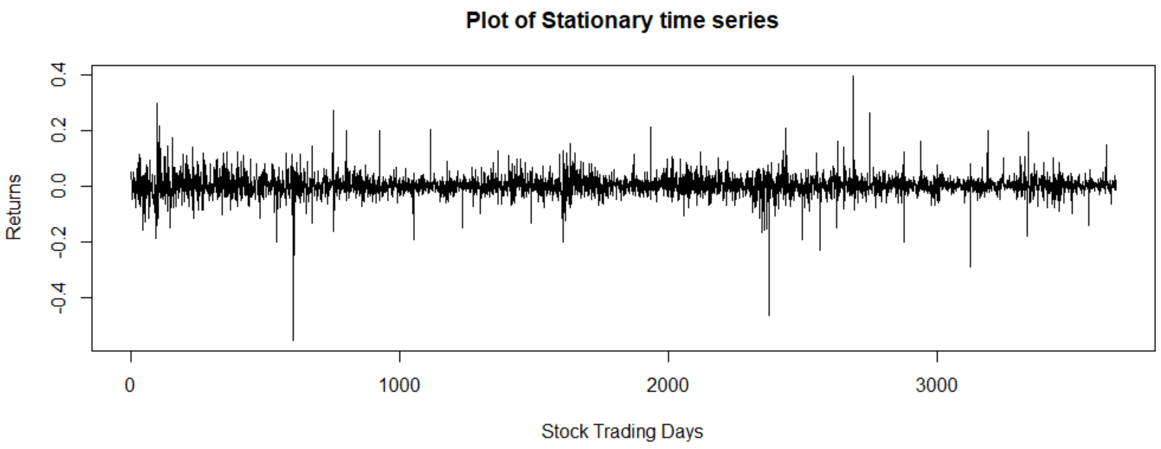
i) The forecasted values is on an average deviated from the real testing values in the absolute magnitude of 0.7914426 units.

ii) The average percentage deviation of the forecasted values from the real testing values is of 13.71441.

iii) The average percentage absolute deviation of the forecasted values from the real testing values is of 13.71509 which is coming to be almost equal to MPE as here almost all the true values of data lie above the forecasted values.

❖ ARCH fitting

- *Checking for volatility clustering:*



Implication: Volatility Clustering is present here as large changes tend to be followed by large changes and small changes tend to be followed by small changes.

- *Fitting ARMA(1,0) as the mean equation of the log return series of stock prices:*

The model is given by, $r_t = \mu_t + a_t$
where,

μ_t is the mean equation at time point t & $\mu_t = 0.001881 + 0.999730\, r_{t-1}$
r_t is the log return of the stock prices at time point t

- *Fitting the linear regression equation of squared shock value on its m(=1) previous squared lag values:*

The equation is given by, $a_t^2 = 0.001645 - 0.004882 a_{t-1}^2$; t=2,…,3665

- *Performing ARCH LM test for checking for ARCH effect:*

SSR_0 = 0.2392814 where $\overline{w} = 0.001644712$
SSR_1 = 0.2391375
F_obs = 2.204403 & $\chi^2_{m;\alpha}$ = 3.841459 when α=0.05

Implication: Here, the null hypothesis is getting accepted at α level of significance as F_obs < $\chi^2_{m;\alpha}$. So, there is no ARCH effect present in the data. So, it's unnecessary to fit & forecast through ARCH model here.

Conclusion

Θ On Closing Stock Prices of Amazon:

ARIMA fitting:

ARIMA(1,1,2) model is a good fit to the training data of log returns of Closing Stock Prices of Amazon and forecasting done through this model on testing data is also good as the error measures' values are fairly small.

ARCH fitting:

ARCH(1) model isn't a very good fit which we can see from the volatility series plot with 2 superimposed conditional SDs as most of the volatility component isn't lying in the intervals and in case of forecasting, it couldn't predict the volatility much ahead of the training data.

Θ On Opening Stock Prices of Amazon:

ARIMA fitting:

ARIMA(1,1,1) model is a good fit to the training data of log returns of Opening Stock Prices of Amazon and forecasting done through this model on testing data is moderate as the error measures' values are fairly small but for later part of the testing data, forecasted values aren't that close to the true values.

ARCH fitting:

ARCH(1) model is a good fit as the volatility components are lying b/w the intervals whih we can see volatility series plot with 2 SDs and in case of forecasting, it couldn't predict volatility much ahead of the training data.

Θ On Closing Stock Prices of Microsoft:

ARIMA fitting:

ARIMA(1,1,1) model is a good fit to the training data of log returns of Closing Stock Prices of Microsoft and forecasting done through this model on testing data is not good as we can see the forecasted values is getting out of the intervals sometimes from the plot where forecasted & true values are plotted together.

ARCH fitting:

Taking ARMA(1,0) as the mean eqn., there is no ARCH effect present in this training data and it's useless to fit ARCH model.

Θ On Opening Stock Prices of Microsoft:

ARIMA fitting:

ARIMA(1,1,4) model is a good fit to the training data of log returns of Opening Stock Prices of Microsoft and forecasting done through this model on testing data is not good as we can see the forecasted values is getting out of the intervals sometimes from the plot where forecasted & true values are plotted together.

ARCH fitting:

ARCH(1) model is a good fit as all the volatility components lie in the intervals and as for forecasting, it predicted gradual increase over all of the testing period unlike in other cases where ARCH predicted constant volatility over the testing period.

Θ On Closing Stock Prices of Netflix:

ARIMA fitting:

ARIMA(1,1,0) model is a good fit to the training data of log returns of Closing Stock Prices of Netflix and forecasting done through this model on testing data is moderate as the error measures' values are fairly small but for later part of the testing data, forecasted values aren't that close to the true values but the forecasted values are well in the interval.

ARCH fitting:

Taking ARMA(1,0) as the mean eqn., there is no ARCH effect present in this training data and it's useless to fit ARCH model.

Θ On Opening Stock Prices of Netflix:

ARIMA fitting:

ARIMA(1,1,2) model is a good fit to the training data of log returns of Closing Stock Prices of Netflix and forecasting done through this model on testing data is moderate as the error measures' values are fairly small but for later part of the testing data, forecasted values aren't that close to the true values but the forecasted values are well in the interval.

ARCH fitting:

Taking ARMA(1,0) as the mean eqn., there is no ARCH effect present in this training data and it's useless to fit ARCH model.

Limitation of Study & Future Scope

Θ Limitations:

- Even though ARIMA models provide a good fit for all the training datas, in case of forecasting, the forecasted values for time points which are a lot ahead of the training data points, aren't that close to the true values meaning ARIMA can't forecast well for the time points which are too ahead in the future.

- If there's even slight seasonality present, ARIMA models can't take those components into account in case of forecasting.

- In most cases, mean equation have to be taken as ARMA(0,1) in ARCH(1) models. Otherwise, it sometimes fails to fit the volatility components properly.

- In most of the cases, ARCH models even though are predicting volatility which are a little ahead from the training data points but it fails to do so properly when we're too ahead in future as ARCH's predicting volatility to stay at the same level which can't be the case in stock prices datasets as stock prices are highly fluctuating.

Θ Future Scope:

- More advanced models for predicting volatility components could have been used to observe better results.

- Even though, there are many advanced time series or financial time series models which has been developed, stock prices are very fluctuating and can be highly unpredictable at times. So, new models which can give better results from the already existing ones are always welcome.

- As analysis of stock prices is useful to many crucial decisions like money investing, stock trading etc., the development in models related to fitting to the stock prices is always trending.

References

- https://www.businessnewsdaily.com/2871-how-most-millionaires-got-rich.html
- https://www.visualcapitalist.com/cp/stock-market-and-gdp-shares/
- https://help.indiainfoline.com/portal/en/kb/articles/what-are-open-high-low-prev-close-prices
- https://www.idrisstsafack.com/post/garch-models-with-r-programming-a-practical-example-with-tesla-stock
- https://saturncloud.io/blog/what-are-logarithmic-returns-and-how-to-calculate-them-in-pandas-dataframe/#:~:text=Logarithmic%20returns%20are%20important%20in,over%20a%20period%20of%20time.
- https://en.wikipedia.org/wiki/Volatility_clustering#:~:text=In%20finance%2C%20volatility%20clustering%20refers,be%20followed%20by%20small%20changes.%22
- https://www.youtube.com/watch?v=Li95a2biFCU
- https://www.youtube.com/watch?v=x81kTJ1afA8&list=PLzAfHlPtM1I4bPhNE5FKOi0S7erZqm5pg
- Tsay, R. S. (2005) Analysis of Financial Time Series, 2nd ed., Wiley, New York.(https://cpb-us-w2.wpmucdn.com/blog.nus.edu.sg/dist/0/6796/files/2017/03/analysis-of-financial-time-series-copy-2ffgm3v.pdf)

Appendix

❖ Snapshots of data used for analysing:

➢ Data on Amazon Stock Prices:

	A	B	C	D	E	F	G
1	Date	Open	High	Low	Close	Adj Close	Volume
2	04-01-1999	2.730729	2.966667	2.665625	2.957813	2.957813	785844000
3	05-01-1999	2.739063	3.24375	2.6625	3.1125	3.1125	1257464000
4	06-01-1999	3.409375	3.509375	3.35	3.45	3.45	723532000
5	07-01-1999	3.428125	4.00625	3.325	3.971875	3.971875	945492000
6	08-01-1999	4.60625	4.978125	3.8	4.00625	4.00625	1333244000
7	11-01-1999	4.6	4.625	4.190625	4.615625	4.615625	778460000
8	12-01-1999	4.5125	4.55	3.975	4.084375	4.084375	488864000
9	13-01-1999	3.125	4.0875	3.125	3.7	3.7	597528000
10	14-01-1999	3.75	3.98125	3.4	3.45	3.45	418248000
11	15-01-1999	3.5	3.751563	3.440625	3.509375	3.509375	402652000
12	19-01-1999	3.7625	3.7625	3.40625	3.495313	3.495313	228108000
13	20-01-1999	3.376563	3.425	2.74375	2.825	2.825	610704000
14	21-01-1999	2.6125	2.759375	2.314063	2.65	2.65	940964000
15	22-01-1999	2.4875	3.146875	2.46875	3.075	3.075	875316000

AMZN_data_1999_2022

➢ Data on Microsoft Stock Prices:

	A	B	C	D	E	F	G
1	Date	Open	High	Low	Close	Adj Close	Volume
2	01-03-2000	58.6875	59.3125	56	58.28125	36.13225	53228400
3	01-04-2000	56.78125	58.5625	56.125	56.3125	34.91171	54119000
4	01-05-2000	55.5625	58.1875	54.6875	56.90625	35.27982	64059600
5	01-06-2000	56.09375	56.9375	54.1875	55	34.09802	54976600
6	01-07-2000	54.3125	56.125	53.65625	55.71875	34.54363	62013600
7	01-10-2000	56.71875	56.84375	55.6875	56.125	34.79548	44963600
8	01-11-2000	55.75	57.125	54.34375	54.6875	33.90427	46743600
9	01-12-2000	54.25	54.4375	52.21875	52.90625	32.79995	66532400
10	1/13/2000	52.1875	54.3125	50.75	53.90625	33.41991	83144000
11	1/14/2000	53.59375	56.96875	52.875	56.125	34.79548	73416400
12	1/18/2000	55.90625	58.25	55.875	57.65625	35.74479	81483600
13	1/19/2000	55.25	55.75	53	53.5	33.16807	97568200
14	1/20/2000	53.53125	54.84375	52.9375	53	32.85807	56349800
15	1/21/2000	53.5	53.625	51.625	51.875	32.16062	68416200

MSFT(2000-2023)

➢ Data on Netflix Stock Prices:

	A	B	C	D	E	F	G
1	Date	Open	High	Low	Close	Adj Close	Volume
2	23-05-2002	1.156429	1.242857	1.145714	1.196429	1.196429	104790000
3	24-05-2002	1.214286	1.225	1.197143	1.21	1.21	11104800
4	28-05-2002	1.213571	1.232143	1.157143	1.157143	1.157143	6609400
5	29-05-2002	1.164286	1.164286	1.085714	1.103571	1.103571	6757800
6	30-05-2002	1.107857	1.107857	1.071429	1.071429	1.071429	10154200
7	31-05-2002	1.078571	1.078571	1.071429	1.076429	1.076429	8464400
8	03-06-2002	1.08	1.149286	1.076429	1.128571	1.128571	3151400
9	04-06-2002	1.135714	1.14	1.110714	1.117857	1.117857	3105200
10	05-06-2002	1.110714	1.159286	1.107143	1.147143	1.147143	1531600
11	06-06-2002	1.15	1.232143	1.148571	1.182143	1.182143	2305800
12	07-06-2002	1.177857	1.177857	1.103571	1.118571	1.118571	1369200
13	10-06-2002	1.135	1.175	1.134286	1.156429	1.156429	484400
14	11-06-2002	1.156429	1.188571	1.128571	1.153571	1.153571	1003800
15	12-06-2002	1.153571	1.182143	1.089286	1.092857	1.092857	1799000

NFLX

❖ Codes in R for analysis:

Similar Process for Analysis of ARIMA & ARCH fitting and forecasting through them, have been done on each set of the stock prices of each companies in R. There's a little change in codes for analysis of the stock prices where ARCH effect was absent. So some of the codes for ARIMA fitting & forecasting, the cases where ARCH effect was present and consequently ARCH fitting & forecasting and the case where ARCH effect was absent and consequently abstaining from fitting are given below.

➢ *Codes for setting up the training & testing data:*

```
rm(list=ls())

data1<- read.csv("C:/Users/User/OneDrive/Documents/Dissertation/AMZN_data_1999_2022.csv",header=T)
data2<-read.csv("C:/Users/User/OneDrive/Documents/Dissertation/MSFT(2000-2023).csv",header=T)
data3<-read.csv("C:/Users/User/OneDrive/Documents/Dissertation/NFLX.csv",header=T)

colnames(data1)

closing_price_amazon=log(data1$Close)
opening_price_amazon=log(data1$Open)
closing_price_microsoft=log(data2$Close)
opening_price_microsoft=log(data2$Open)
closing_price_netflix=log(data3$Close)
opening_price_netflix=log(data3$Open)

n1=length(closing_price_amazon);n1
n2=length(closing_price_microsoft);n2
n3=length(closing_price_netflix);n3

tl_amazon=round(n1*0.8);tl_amazon
tl_microsoft=round(n2*0.8);tl_microsoft
tl_netflix=round(n3*0.8);tl_netflix

tsl_amazon=n1-tl_amazon;tsl_amazon
tsl_microsoft=n2-tl_microsoft;tsl_microsoft
tsl_netflix=n3-tl_netflix;tsl_netflix


library(tseries)
library(forecast)
library(rugarch)
library(FinTS)
library(e1071)
library(zoo)
```

➢ *Codes for just ARCH effect checking on Netflix Closing Prices (as it's absent and consequently abstaining from fitting):*

```
####ARCH Fitting

adf.test((closing_price_netflix[1:tl_netflix]))
adf.test(diff(closing_price_netflix[1:tl_netflix]))

plot((diff(closing_price_netflix[1:tl_netflix])),type="l",main="Plot of Stationary time series",xlab="Stock Trading Days",ylab="Returns")
acf(diff(closing_price_netflix[1:tl_netflix]))
pacf((closing_price_netflix[1:tl_netflix]))
mean_eq_cn=arma((closing_price_netflix[1:tl_netflix]),order=c(1,0));mean_eq_cn
resi_mean_eq_cn=na.omit(residuals(mean_eq_cn))
n_cn=length(resi_mean_eq_cn)
resi_1st_cn=resi_mean_eq_cn[1:(n_cn-1)]
resi_2nd_cn=resi_mean_eq_cn[2:n_cn]
lm((resi_2nd_cn^2)~(resi_1st_cn^2))
summary(lm((resi_2nd_cn^2)~(resi_1st_cn^2)))
et_cn=residuals(lm((resi_2nd_cn^2)~(resi_1st_cn^2)))
w_bar_cn=mean(resi_mean_eq_cn^2);w_bar_cn
SSR0_cn=sum((resi_mean_eq_cn^2 - w_bar_cn)^2);SSR0_cn
SSR1_cn=sum(et_cn^2);SSR1_cn
F_obs_cn=(((SSR0_cn - SSR1_cn)/1)/(SSR1_cn/(tl_netflix-(2*1)-1)));F_obs_cn
qchisq(0.95,1)
#as we can see there's no ARCH effect present, we won't proceed for ARCH fitting
```

➢ *Codes for ARCH ftting & forecasting on Amazon Closing Prices (where ARCH effect is present):*

```
####ARCH Fitting

adf.test((closing_price_amazon[1:tl_amazon]))
adf.test(diff(closing_price_amazon[1:tl_amazon]))

plot((diff(closing_price_amazon[1:tl_amazon])),type="l",main="Plot of Stationary time series",xlab="Stock Trading Days",ylab="Returns")
acf(diff(closing_price_amazon[1:tl_amazon]))
pacf((closing_price_amazon[1:tl_amazon]))
mean_eq_ca=arma((closing_price_amazon[1:tl_amazon]),order=c(1,2));mean_eq_ca
resi_mean_eq_ca=na.omit(residuals(mean_eq_ca))
n_ca=length(resi_mean_eq_ca)
resi_1st_ca=resi_mean_eq_ca[1:(n_ca-1)]
resi_2nd_ca=resi_mean_eq_ca[2:n_ca]
lm((resi_2nd_ca^2)~(resi_1st_ca^2))
summary(lm((resi_2nd_ca^2)~(resi_1st_ca^2)))
et_ca=residuals(lm((resi_2nd_ca^2)~(resi_1st_ca^2)))
w_bar_ca=mean(resi_mean_eq_ca^2);w_bar_ca
SSR0_ca=sum((resi_mean_eq_ca^2 - w_bar_ca)^2);SSR0_ca
SSR1_ca=sum(et_ca^2);SSR1_ca
F_obs_ca=(((SSR0_ca - SSR1_ca)/1)/(SSR1_ca/(tl_amazon-(2*1)-1)));F_obs_ca
qchisq(0.95,1)
ArchTest(diff(closing_price_amazon[1:tl_amazon]),lag=1)  ##by ArchTest function
garch(diff(closing_price_amazon[1:tl_amazon]),c(0,1),grad="numerical",trace=F)

ca_arch=ugarchspec(variance.model=list(garchOrder=c(1,0)),mean.model=list(armaOrder=c(1,2)))
ca_arch_fit=ugarchfit(ca_arch,data=diff(closing_price_amazon[1:tl_amazon]))
ca_arch_fit
plot(ca_arch_fit,xlim=c(1:tl_amazon))

ca_arch_f=(ugarchforecast(ca_arch_fit,n.ahead=tsl_amazon))
ca_arch_f
plot(sigma(ca_arch_f),main="Volality Series over Time points",xlab="Stock Trading Days",ylab="Volatility Values")
```

➢ *Codes for ARIMA fitting & forecasting on Amazon Closing Prices:*

```
####Arima Fitting

adf.test((closing_price_amazon[1:tl_amazon]))
adf.test(diff(closing_price_amazon[1:tl_amazon]))

plot(((closing_price_amazon[1:tl_amazon])),type="l")
acf(diff(closing_price_amazon[1:tl_amazon]),main="ACF plotting on stationary series")
pacf((closing_price_amazon[1:tl_amazon]),main="PACF plotting on non-stationary series")
fit1=arima((closing_price_amazon[1:tl_amazon]),order=c(1,1,2))
fit1
plot(
	((closing_price_amazon[1:tl_amazon])),
	type="l",
	main="Graph of Log returns of Amazon Closing Stock Prices",
	xlab="Stock Trading Days in Training Data",
	ylab="Log returns of Stock Prices"
	)
lines(fitted(fit1),col="red")
legend(
	"topleft",
	legend=c("True Values","Fitted Values"),
	col=c("black","red"),
	lty=1
	)
summary(fit1)
fcast_ca=data.frame(forecast(fit1,h=tsl_amazon))
matplot(
	c(1:length(fcast_ca$Point.Forecast)),
	cbind(closing_price_amazon[(tl_amazon+1):n1],fcast_ca$Point.Forecast,fcast_ca$Lo.95,fcast_ca$Hi.95),
	col=c("black","red","blue","green"),
	main="Amazon Closing & forecasted Prices' Log returns' Graph(based on Testing data)",
	xlab="Stock Trading Days in Testing data",
	ylab="Log returns of Stock Prices",
	ylim=c(2,10)
	)
legend(
	"topleft",
	legend=c("True Values","Forecast Values","Upper 95% CLs","Lower 95% CLs"),
	col=c("black","red","green","blue"),
	lty=1
	)
e_ca=closing_price_amazon[(tl_amazon+1):n1]-fcast_ca$Point.Forecast
MAD_ca=mean(abs(e_ca));MAD_ca
MPE_ca=mean((e_ca/closing_price_amazon[(tl_amazon+1):n1])*100);MPE_ca
MAPE_ca=mean(abs(e_ca/closing_price_amazon[(tl_amazon+1):n1])*100);MAPE_ca
```