NVIDIA STOCK ANALYSIS

FINAL REPORT

FIN 620A – Financial Econometrics

Group 3
Aayush Vikram Singh
Dev Vishal Prakash
Priestley Fernandes
Maneet Batra

Table of Content

Overview of Asset and Market

NVIDIA Corp (NVIDIA) designs and develops graphics processing units, central processing units, and system-on-a-chip units for gaming, professional visualization, data center, and automotive markets. Nvidia was founded on April 5, 1993, by Jensen Huang (CEO as of 2022), a Taiwanese American electrical engineer who was previously the director of Core Ware at LSI Logic and a microprocessor designer at AMD; Chris Malachowsky, an engineer who worked at Sun Microsystems; and Curtis Priem, who was previously a senior staff engineer and graphics chip designer at IBM and Sun Microsystems.

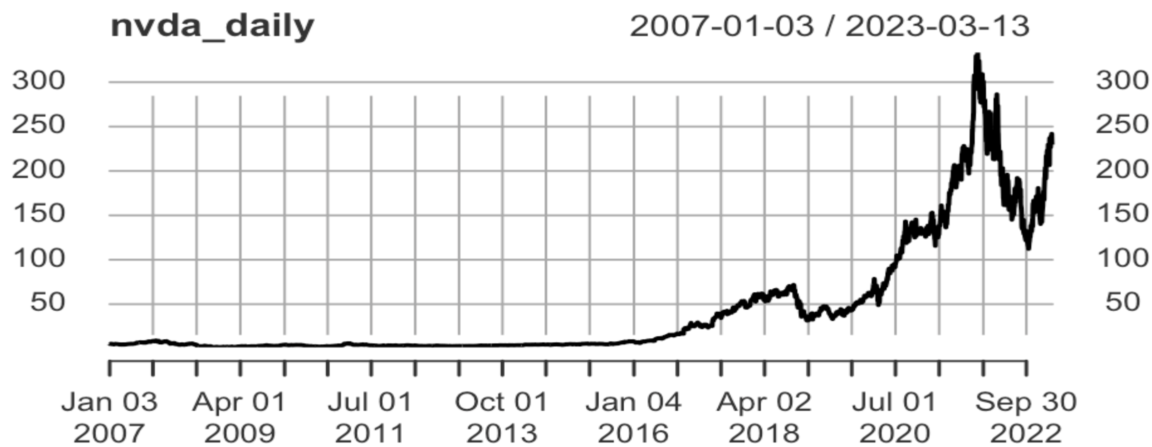
Nvidia Corporation is an American multinational technology company incorporated in Delaware and based in Santa Clara, California. It is a software and fabless company which designs graphics processing units (GPUs), application programming interface (APIs) for data science and high-performance computing as well as system on a chip units (SoCs) for the mobile computing and automotive market. Nvidia is a dominant supplier of artificial intelligence hardware and software. Its professional line of GPUs are used in workstations for applications in such fields as architecture, engineering and construction, media and entertainment, automotive, scientific research, and manufacturing design.

The reason for picking as the asset for our project is the implied volatility that investors and analysts are associating with the options of the company in the past month. Implied volatility shows how much movement the market is expecting in the future. Options with high levels of implied volatility suggest that investors in the underlying stocks are expecting a big move in one direction or the other. It could also mean there is an event coming up soon that may cause a big rally or a huge sell-off. Hence option traders are pricing a big move for Nvidia and given the high volatility of its prices in the recent years there is good reason for it The semiconductor industry market has had a volatile nature in general over the recent years due to chip shortages and relatively low ESG scores for many of the big players in this sector hence Nvidia as one the leaders in the sector presents a good investment opportunity for the investment

Visualization

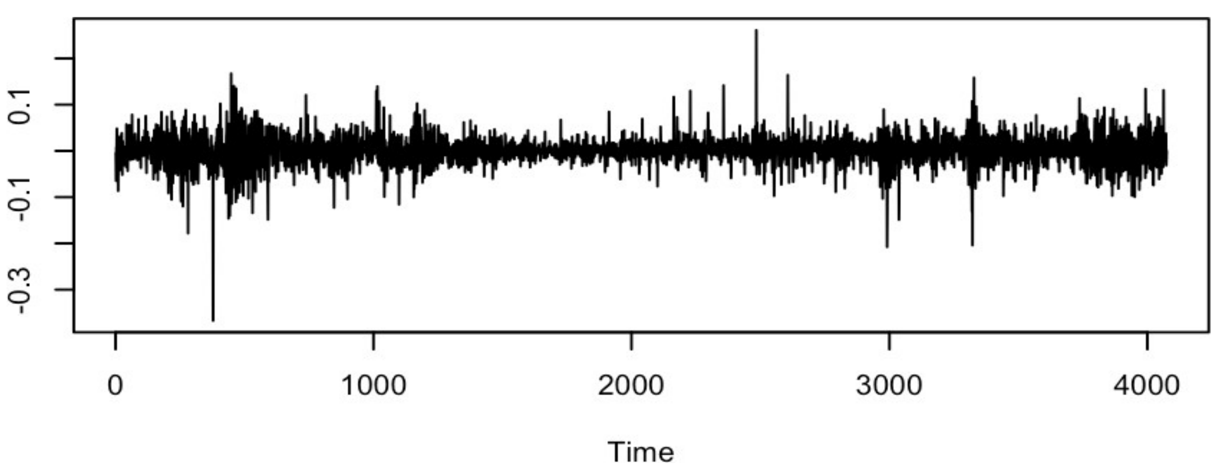
Time Series:

Prices:



The above shows the time series plot for Nvidia for last 15 years the stock shows high periods of volatility for the last five years with high periods of growth and fall in the last five years.

Returns:



As the above graph shows there are periods of high volatility and periods of shocks for the stock.

Descriptive Statistics and AnalysisPrices:

DAILY

	NVDA.Adjusted
nobs	4.076000e+03
NAs	0.000000e+00
Minimum	1.353455e+00
Maximum	3.333508e+02
1. Quartile	3.490134e+00
3. Quartile	5.335527e+01
Mean	4.355859e+01
Median	5.754476e+00
Sum	1.775448e+05
SE Mean	1.051920e+00
LCL Mean	4.149625e+01
UCL Mean	4.562092e+01
Variance	4.510243e+03
Stdev	6.715834e+01
Skewness	1.952460e+00
Kurtosis	3.136425e+00

Returns:

DAILY

	NVDA.Adjusted
nobs	4075.000000
NAs	0.000000
Minimum	-0.367108
Maximum	0.260876
1. Quartile	-0.013656
3. Quartile	0.016276
Mean	0.000915
Median	0.001446
Sum	3.728618
SE Mean	0.000487
LCL Mean	-0.000040
UCL Mean	0.001870
Variance	0.000966
Stdev	0.031086
Skewness	-0.442908
Kurtosis	9.440734

T-test

Prices:

DAILY

```
        One Sample t-test

data:  as.vector(nvda_daily)
t = 41.409, df = 4075, p-value < 2.2e-16
alternative hypothesis: true mean is not equal t
95 percent confidence interval:
 41.49625 45.62092
sample estimates:
mean of x
 43.55858
```

Returns:

DAILY

```
        One Sample t-test

data:  as.vector(nvda_daily_log_return)
t = 1.879, df = 4074, p-value = 0.06032
alternative hypothesis: true mean is not equal t
95 percent confidence interval:
 -3.972295e-05  1.869720e-03
sample estimates:
   mean of x
0.0009149984
```

Volume:

DAILY

```
        One Sample t-test

data:  as.vector(nvda_daily_volume)
t = 106.11, df = 4075, p-value < 2.2e-16
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 53854512 55882085
sample estimates:
mean of x
 54868298
```

We can see the 95% confidence interval, and the mean values of x in each case.

Only in the case of daily log returns, we infer that the p-value is more than 0.05, which means that the result is statistically insignificant. As expected, the mean of returns is close to zero, this is implied since we accept the null hypothesis that true mean is equal to zero.

In all other cases, it is almost equal to zero, or less that 0.05. This shows that the results are statistically significant.
Next, we perform the normality test using the Jarque-Bera method.

Jarque-Bera method

The Jarque-Bera test is a goodness-of-fit test that measures if sample data has skewness and kurtosis that are similar to a normal distribution. The Jarque-Bera test statistic is always positive, and if it is not close to zero, it shows that the sample data do not have a normal distribution.

The null hypothesis of the Jarque-Bera test is a joint hypothesis of the skewness being zero and the excess kurtosis being zero. With a p-value > 0.05, the data are consistent with having skewness and excess kurtosis zero.

We can see from the interpretations that all the tests have p-value < 0.05. This means that the data is not consistent with skewness, and excess kurtosis exists in each test.

Prices:

DAILY

```
Title:
 Jarque - Bera Normalality Test

Test Results:
  STATISTIC:
    X-squared: 196.9747
  P VALUE:
    Asymptotic p Value: < 2.2e-16
```

Returns:

DAILY

```
Title:
 Jarque - Bera Normalality Test

Test Results:
  STATISTIC:
    X-squared: 17.9922
  P VALUE:
    Asymptotic p Value: 0.0001239
```

Volume:

DAILY

```
Title:
 Jarque - Bera Normalality Test

Test Results:
  STATISTIC:
    X-squared: 34042.4723
  P VALUE:
    Asymptotic p Value: < 2.2e-16
```

Skew Kurtosis Test

The skewness is a measure of symmetry or asymmetry of data distribution, and kurtosis measures whether data is heavy-tailed or light-tailed in a normal distribution. Data can be positive-skewed (data-pushed towards the right side) or negative-skewed (data-pushed towards the left side).

When both skewness and kurtosis are close to zero, the pattern of responses is considered a normal distribution.

We can see this is the case for the prices. Hence, prices are in normal distribution. This is also the case with daily data in log returns and volume.
But, the monthly data in log returns and volume are having values less than 1 (non-zero) which mean that although they have excellent skewness value, they have substantial non-normality, and are not in normal distribution.
The following is the table for p values

Prices		
	Daily	Monthly
Skew	0	0
Kurtosis	0	0
Returns		
	Daily	Monthly
Skew	0	0.0115289
Kurtosis	0	0.001075
Prices		
	Daily	Monthly
Skew	0	3.59E-06
Kurtosis	0	0.227677

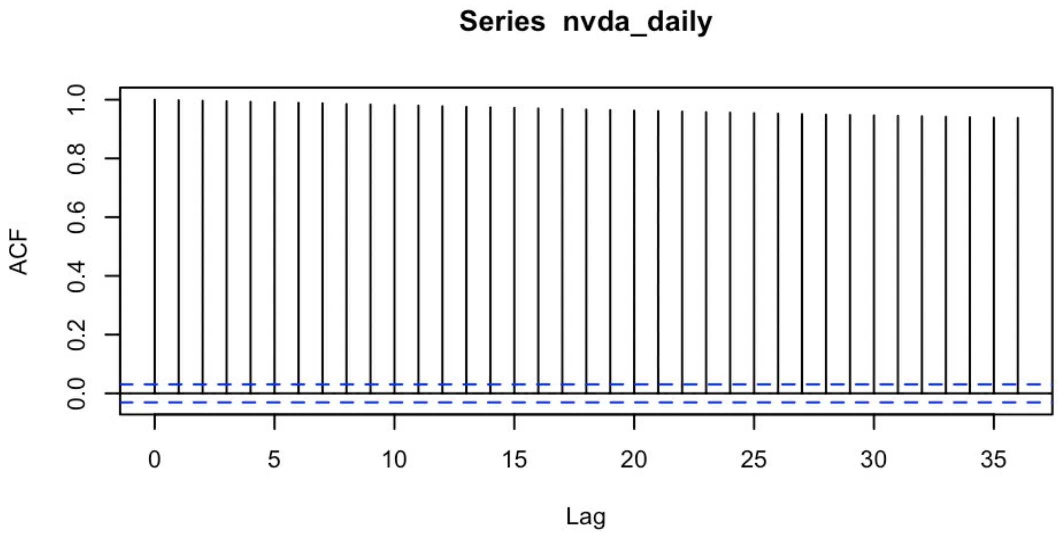
Ljung Box test

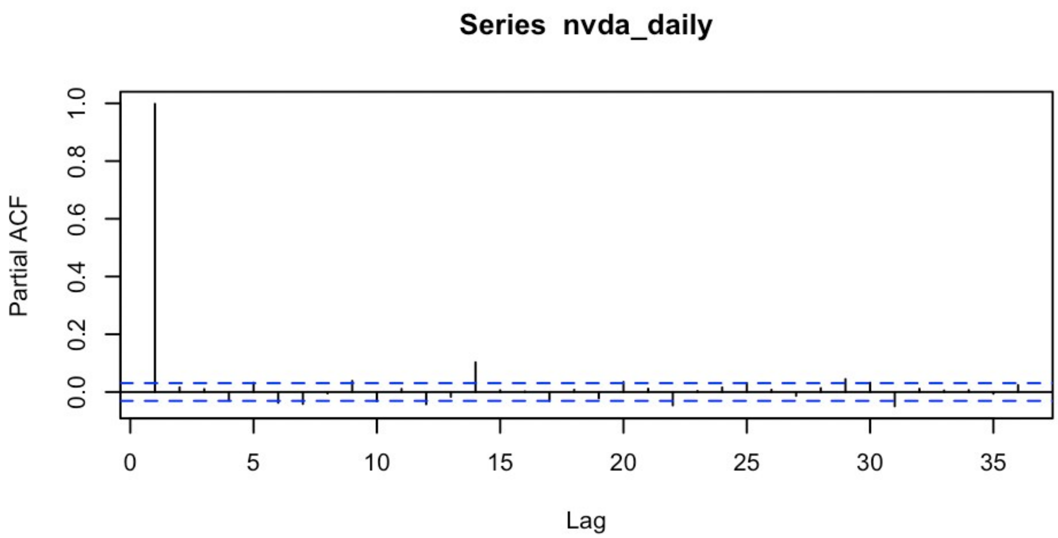
The Ljung Box test is applied to the residuals of a time series after fitting an ARMA(p,q) model to the data. The test examines m autocorrelations of the residuals. If the autocorrelations are very small, we conclude that the model does not exhibit significant lack of fit.
The Ljung-Box test is a method that tests for the absence of autocorrelation in residuals. If the p-value > 0.05, this implies that the residuals of the data are independent. But in the test results for the NVDA data, all the p-values are below a significance level (0.05). This indicates. the values are autocorrelated and the data is not independent.

```
        Box-Ljung test

data:  nvda_monthly
X-squared = 1423, df = 10, p-value < 2.2e-16

        Box-Ljung test

data:  nvda_daily_log_return
X-squared = 33.302, df = 10, p-value = 0.0002423
        Box-Ljung test

data:  nvda_daily
X-squared = 40025, df = 10, p-value < 2.2e-16

        Box-Ljung test

data:  nvda_daily_log_return
X-squared = 33.302, df = 10, p-value = 0.0002423
        Box-Ljung test

data:  nvda_monthly_log_return
X-squared = 10.876, df = 10, p-value = 0.3673
        Box-Ljung test

data:  nvda_monthly_volume
X-squared = 179.05, df = 10, p-value < 2.2e-16

        Box-Ljung test

data:  nvda_daily_volume
X-squared = 8825.1, df = 10, p-value < 2.2e-16
```

ACF and PACF Analysis

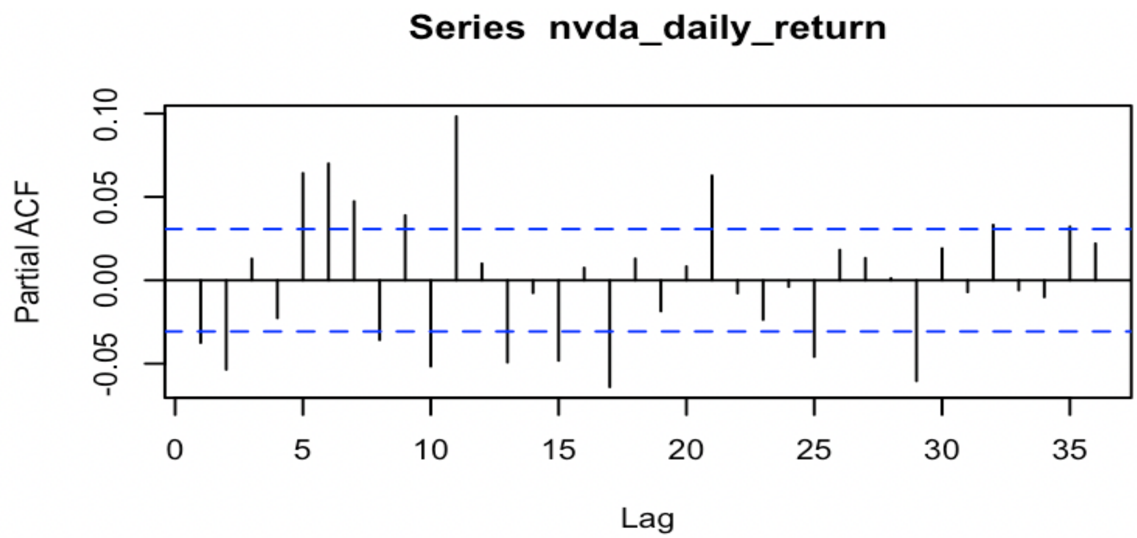
Prices:

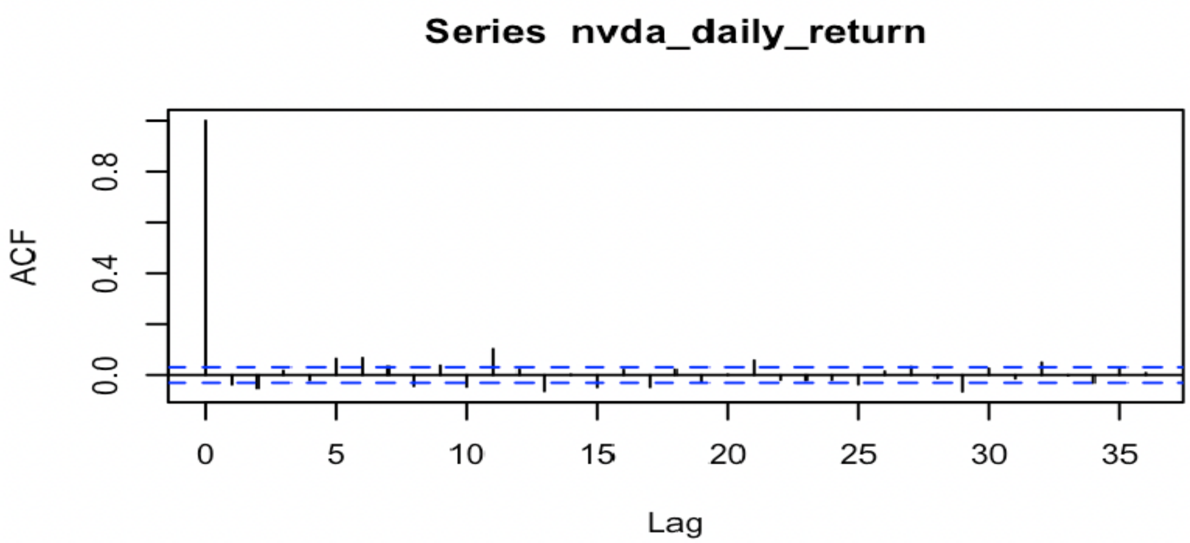




Firstly, we analyze the Autocorrelation and the Partial Autocorrelation function of daily prices and we see that there is high autocorrelation between the daily price of NVidia shares and thus this time series may not be ideal for forecasting. Thus, an AR model might be more useful for this time series.

Returns:

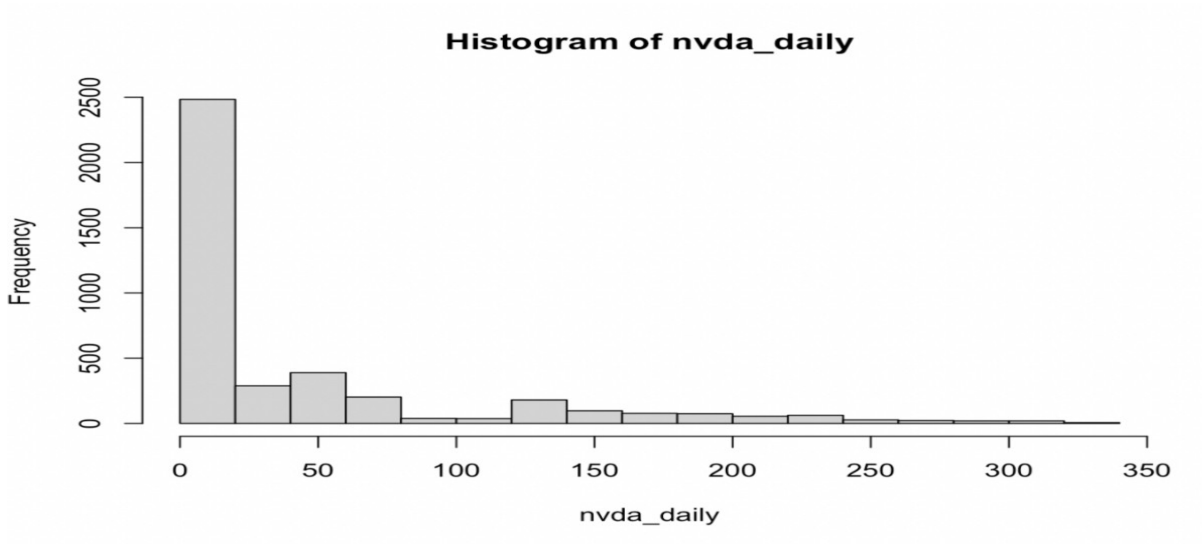




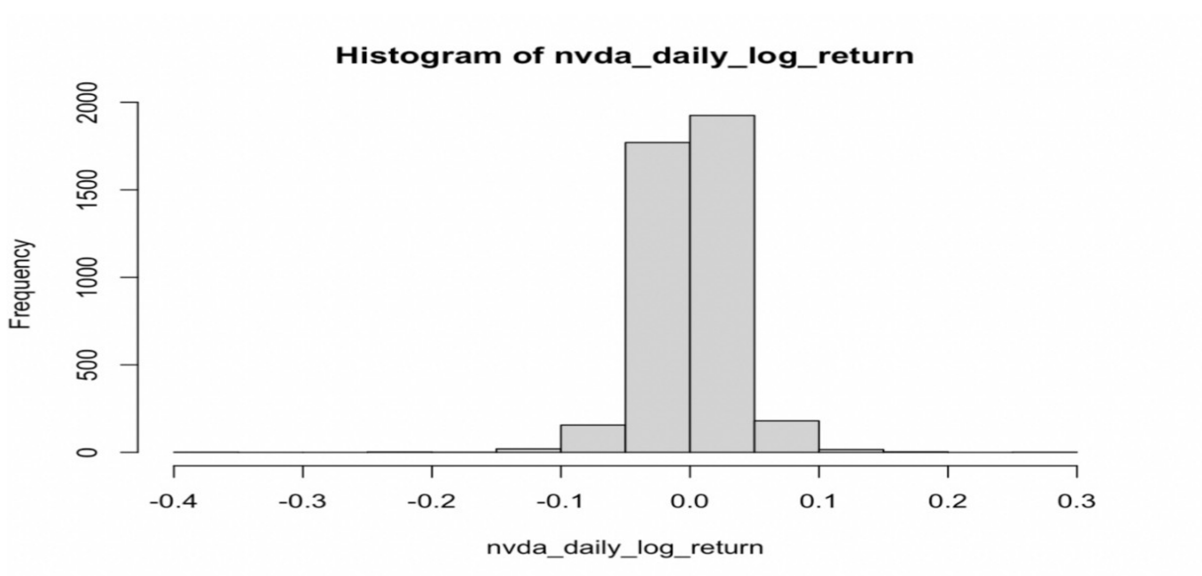
The ACF plot for raw returns of the Nvidia stock shows several significant values. A number of values are significant like AR 3 and AR 5 and AR 6. This time series is better for forecasting than the daily return prices because the significant values are constantly decaying constantly and have been reducing since the first lag. The PACF for this time series though has several significant values and thus will not be very usable for forecasting as the value of PACF has been reducing geometrically until lag 30 and reduces in significance after that.

Histogram

Prices:



Returns:



The histogram of log return is normally distributed and hence being normally distributed it removes the effects on non-stationarity from the time series hence we use both the log returns and raw returns of the prices and as such we the ARIMA models have been created and defined using both the datasets

Unit Root Test

Using ADF or Augmented Dickey Fuller Test for analysis:

Null Hypothesis:

Unit root is present in the time series sample or series is non-stationary.

On applying the ADF test to prices and returns following results are obtained:

Prices:

1. Daily

Dickey-Fuller = -1.6798, Lag order = 15, p-value = 0.7139
Thus, since p value is greater than 0.05, null hypothesis is accepted. It means that the series is non-stationary.

Returns:

1. Daily

Dickey-Fuller = -15.683, Lag order = 15, p-value = 0.01
The p value here less than 0.01. It leads us to reject the null hypothesis and thus we conclude that the series of log returns is stationary.

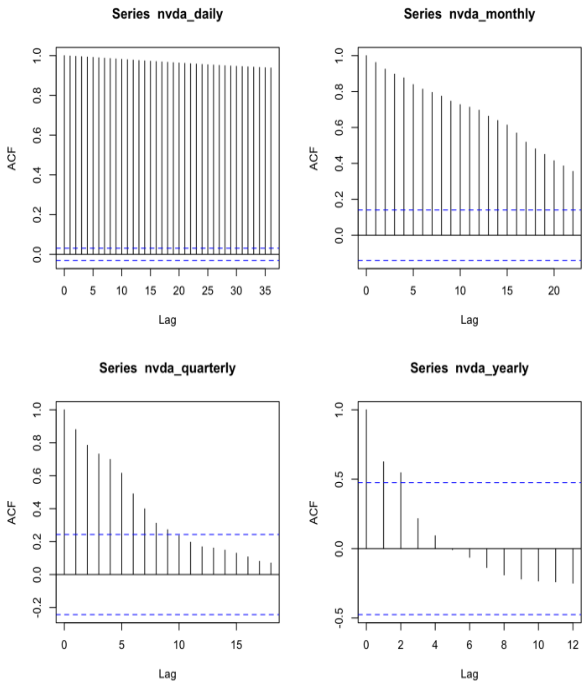
Based on the observations of Ljung Box test, the daily returns have some or the autocorrelation between lags.

Arima Model

Checking seasonality

As a part of this analysis, we plotted different time series plots. The plots below include daily, monthly, quarterly and yearly prices. From the graphs below, it is quite evident that there are no significant seasonal patterns which can direct the course of the analysis. There is an upward trend with considerable variance.

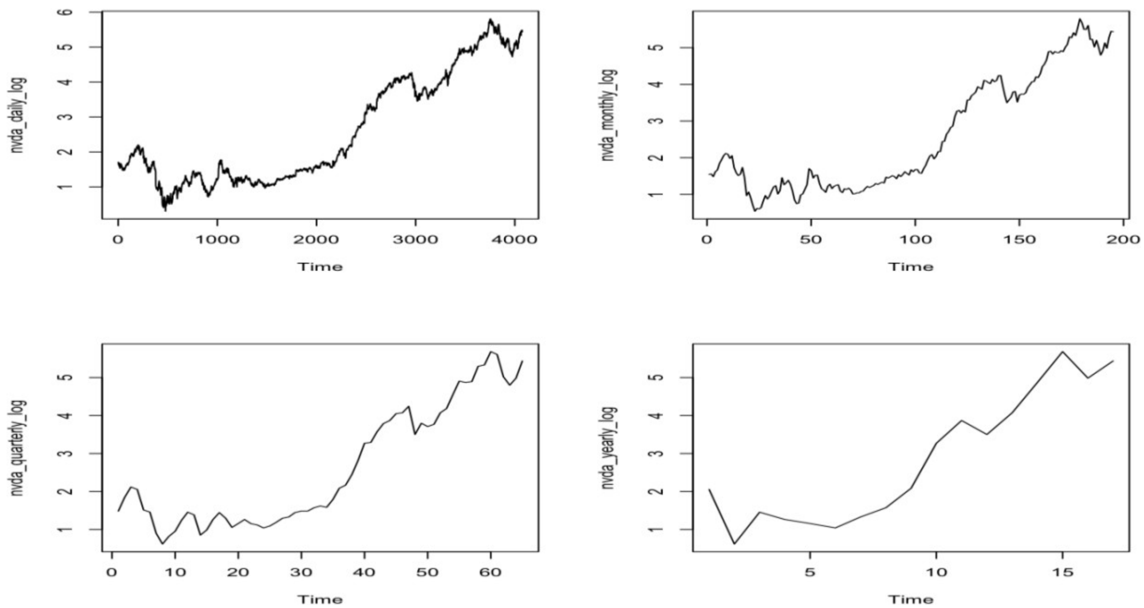
In order to validate our observations, we plotted ACF graph for each of the above cas

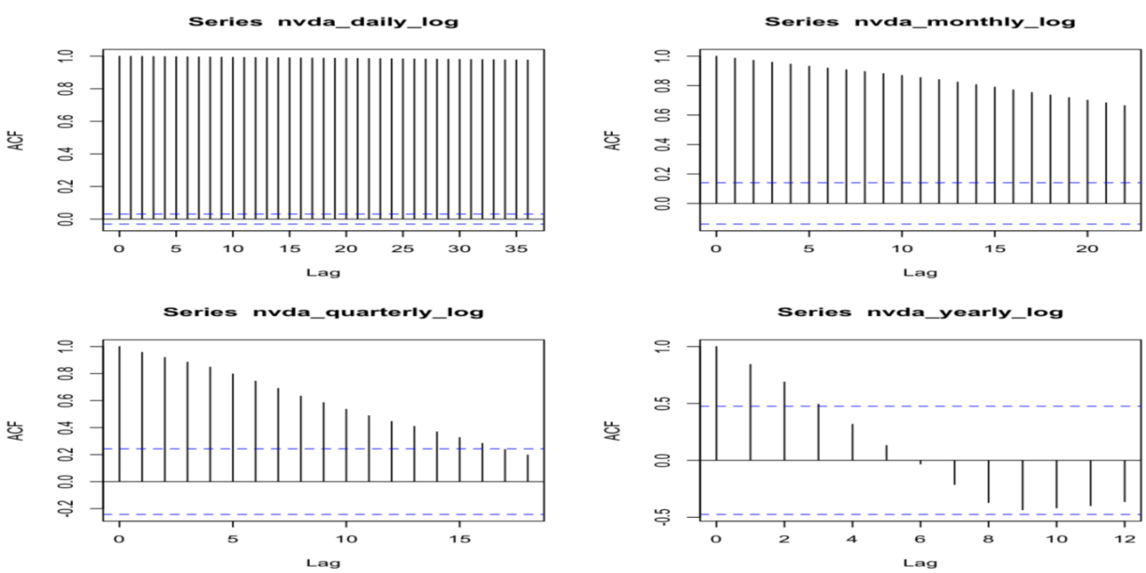


The ACF can serve the purpose of identifying spikes at periodic intervals which prove the existence of seasonality with respect to a particular interval. In the ACF plots below, no such pattern or repetition in autocorrelations is observed. Thus, seasonality is not present.

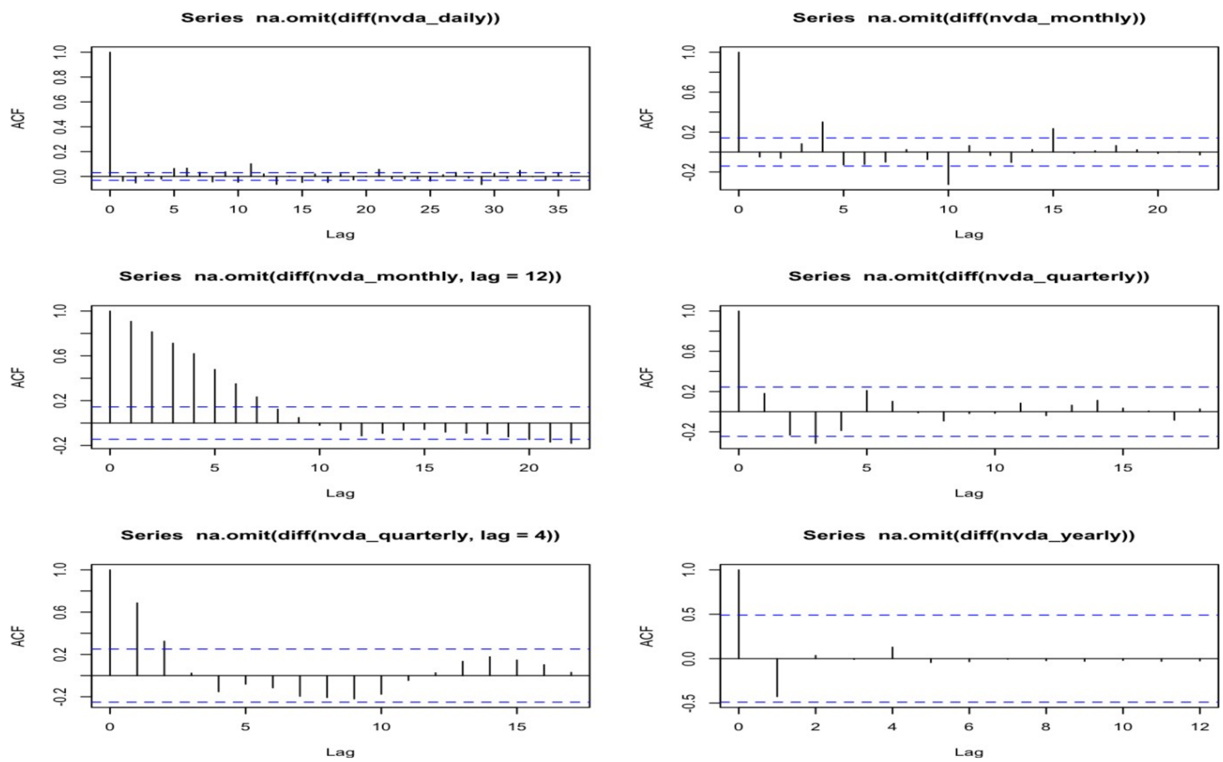
But, an interesting observation can be observed when we tried to take the log of prices to smoothen the curve or eliminate the variance. The four plots below hint towards some pattern, but unfortunately, it seems to be relevant for long term analysis. A vague pattern can be observed from year 8 to 12 and year 12 to 16(almost a 5 year interval). But, it seems to be irrelevant to our analysis which deals with daily price prediction. A long-term pattern can be seen at a later period.

Checking stationarity





From the figure above, it can be seen that prices series(log) is non-stationary. The constant values of autocorrelations in the ACF graph indicate this. Even earlier, using ADF test, we were able to prove that daily prices were non-stationary. Thus, differencing is carried out. Plots show that on differencing once stationarity can be achieved.



.

We further used Ljung Box test to prove that there is no white noise and that there is at least one autocorrelation existing between lags. Similarly, ADF test substantiates stationarity.

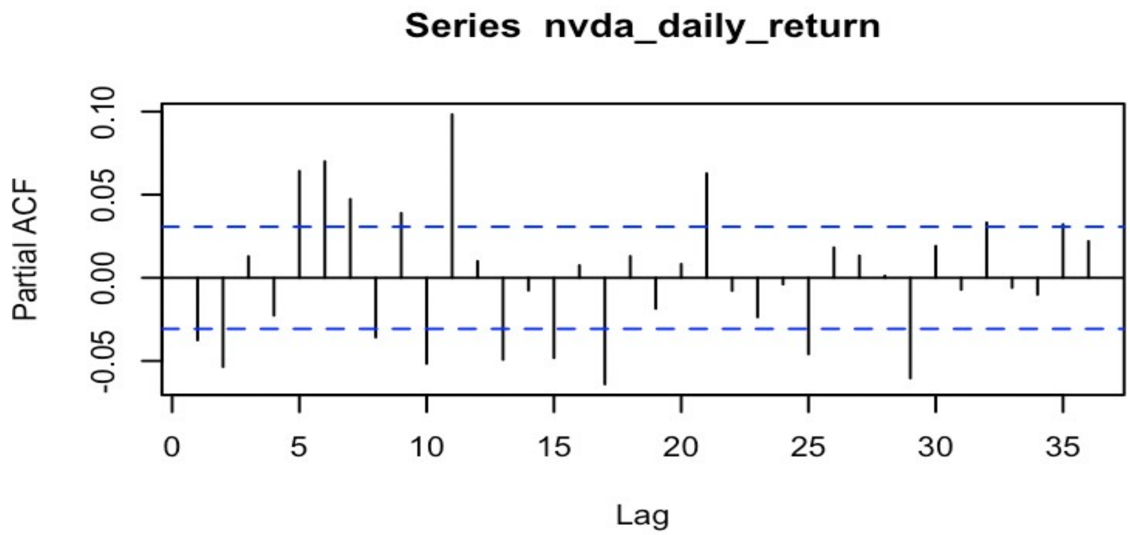
The results are as follows:

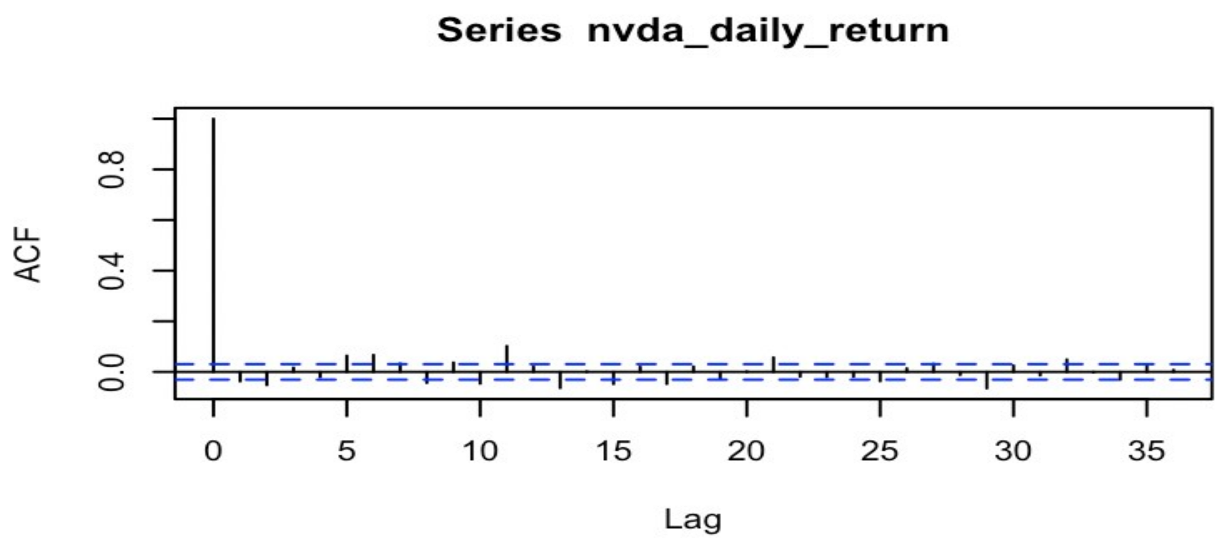
Ljung Box test:

X-squared = 81.697, df = 10, p-value = 2.333e-13
A p-value less than 0.05, indicates that there exists at least one autocorrelation for the lags.

ADF test:

Dickey-Fuller = -15.59, Lag order = 15, p-value = 0.01
A p-value less than 0.05 shows that stationarity exists.





We followed a step wise approach for log returns.

1. Checking for AR model: (PACF used)
It cannot be a pure AR model since its ACF doesn't seem to be decaying towards zero since there are few significant values at the end and in PACF, none of the lags have a cutoff after their occurrences. Even if there are, its AR(15) and AR(20) which is high lag value.

2. Checking for MA model: (ACF used)
In ACF, there are many insignificant values after the lag 1 and after lag 8. Also, there is a decaying nature observed for PACF. But there are many significant values that can be seen in ACF after lag 1. It leads to ambiguity in the selection of right order for our model and to declare it as either pure AR or pure MA model. Thus, it is decided to move forward with ARMA which requires use of EACF analysis.

EACF Analysis for ARIMA

AR/MA	0	1	2	3	4	5	6	7	8	9	10	11	12	13
0	x	o	o	o	o	o	x	x	o	o	o	o	o	o
1	x	o	o	o	o	o	o	x	x	o	o	o	o	o
2	x	x	o	o	o	o	o	o	o	x	o	o	o	o
3	x	x	x	o	o	o	o	o	o	o	o	o	o	o
4	x	x	x	x	o	o	o	o	o	o	o	o	o	o
5	x	x	x	o	o	o	o	o	o	o	o	o	o	o
6	x	x	o	x	x	x	o	o	o	o	o	o	o	o
7	x	x	x	x	x	x	x	o	o	o	o	o	o	o

Theoretically, the top-left zero of the EACF table for an ARMA (p, q) process should appear in the p-th row and q-th column (with the row and column names both starting at 0).
(In reality, as the sample EACF values contain sampling variability, the sample EACF table will not be as precise as the examples that follow.)

Hence, in this case for the process, we can consider a few models, to state a few:

ARMA (2,2)
ARMA (2,3)
ARMA (2,4)
ARMA (2,5)
ARMA (4,4)
ARMA (3,3)
ARMA (3,4)
ARMA (4,3)
ARMA (4,4)

Similarly, a for loop was used to calculate AIC for various combination of p and r from 1 to 6:

The ARIMA results for various combinations for p and r are as follows:
The AIC is follows:

Here the log returns are used for evaluation. Thus, r =0

The top 5 models with least AIC are as follows:

ARMA (4,5)
-16737.31

ARMA (5,6)
-16737.01

ARMA (4,6)
-16734.2

ARMA (4,3)
-16733.73

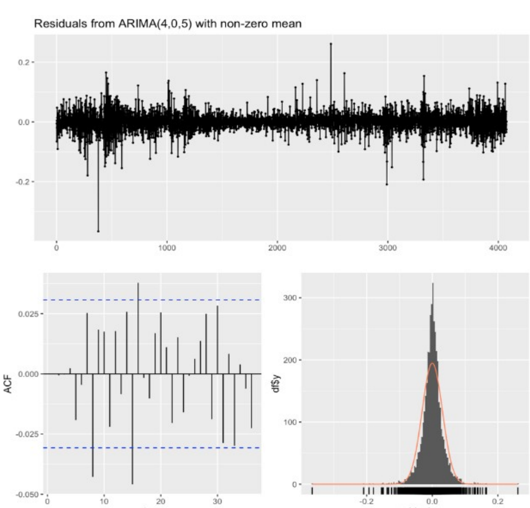
ARMA (6,5)
-16727.67

Few of the relevant results of Ljung box test out of all models selected are shown here:

ARMA (4,5)

Ljung-Box test

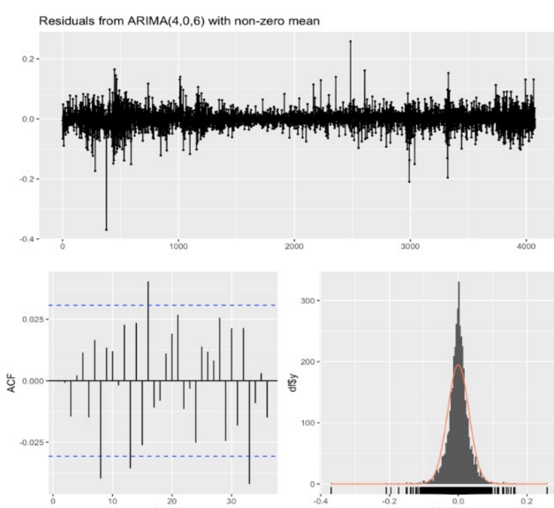
data: Residuals from ARIMA (4,0,5) with non-zero mean
Q^* = 14.338, df = 1, p-value = 0.0001528



ARMA (4,6)

Ljung-Box test

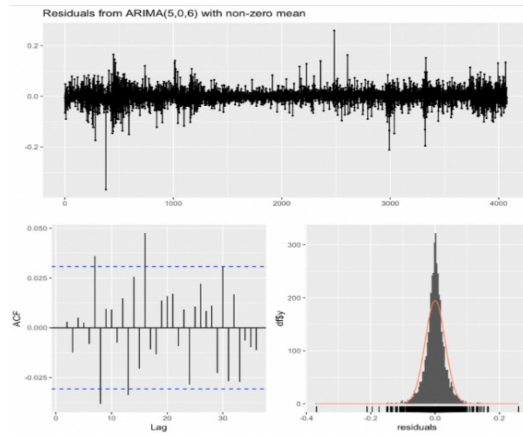
```
        Ljung-Box test

data:  Residuals from ARIMA(4,0,6) with non-zero mean
Q* = 11.226, df = 0, p-value < 2.2e-16

Model df: 10.   Total lags used: 10
```



ARMA (0,1)

<u>**Ljung-Box test**</u>

data: Residuals from ARIMA (0,0,1) with non-zero mean
Q* = 26.304, df = 9, p-value = 0.00182



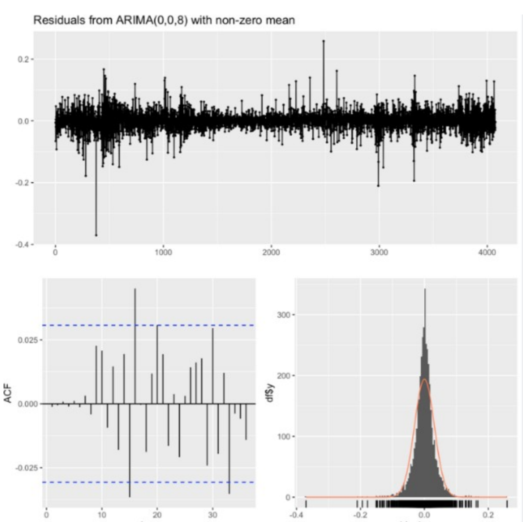
From the above results, it is evident that the none of the residuals exhibit a white noise. The p-value is less than the significant level of 0.05 in all the cases we selected. Thus, none of the models can be selected as our final model as the null hypothesis of zero autocorrelation between all lags of residuals had to be rejected.

Further, we explored orders upto 8 by individually using orders between 6 to 8 in arima function. A similar observation surfaced with most of the residuals failing to exhibit a white noise behavior. But, one of the models ARIMA(0,0,8) for log returns turned out to be valid one.

Following is the Ljung Box test of ARIMA(0,0,8) for log returns:

ARMA (0,8)

<u>**Ljung-Box test**</u>
data: Residuals from ARIMA(0,0,8) with non-zero mean
Q* = 4.0019, df = 2, **p-value = 0.1352**



Since we fail to reject the null hypothesis (p>0.05), ARIMA (0,0,8) for log returns was further analyzed for significance of coefficients.

ARIMA(0,1,8) model:

Call:
arima(x = ts(nvda_daily_log), order = c(0, 1, 8))

Coefficients:

	ma1	ma2	ma3	ma4	ma5	ma6	ma7	ma8
	-0.0297	0.0136	0.0170	-0.0125	0.0049	-0.0048	0.0252	-0.0545
s.e.	0.0156	0.0157	0.0157	0.0157	0.0154	0.0154	0.0155	0.0151

sigma^2 estimated as 0.0009612: log likelihood = 8372.94, aic = -16729.87

Significance testing of coefficients resulted in:
z test of coefficients:

	Estimate	Std. Error	z value	Pr(>|z|)	
ma1	-0.0297042	0.0156304	-1.9004	0.0573796	.
ma2	0.0135868	0.0156584	0.8677	0.3855588	
ma3	0.0169877	0.0156555	1.0851	0.2778807	
ma4	-0.0125458	0.0156518	-0.8016	0.4228073	
ma5	0.0048718	0.0153940	0.3165	0.7516453	
ma6	-0.0048088	0.0153667	-0.3129	0.7543307	
ma7	0.0252298	0.0154648	1.6314	0.1027994	
ma8	-0.0545141	0.0150509	-3.6220	0.0002924	*

Signif. codes: 0 '*' 0.001 '*' 0.01 '' 0.05 '.' 0.1 ' ' 1

Considering 0.05 as the significance level ma8 is the only significant co-efficient. Thus, it can be reduced to a order of 1 (ma8)
A Ljung-box test on the residuals of the fixed equation resulted in the following:

Ljung-Box test

data: Residuals from ARIMA (0,1,8)
Q* = 10.576, df = 2, p-value = 0.005051
Model df: 8. Total lags used: 10

It can be concluded that, since the p-value is less than 0.05, the residuals still have some autocorrelations present among the lags which is not expected. Thus, further investigation is required. Analyzed for all possible models up to order 8. Focused on both low AR and MA orders first with 8 as one of the orders.

Two models were selected:

Ljung-Box test

data: Residuals from ARIMA(8,1,0)
Q* = 3.396, df = 2, p-value = 0.1831
Model df: 8. Total lags used: 10

Ljung-Box test

data: Residuals from ARIMA (1,1,8)
Q* = 2.7713, df = 1, p-value = 0.09597
Model df: 9. Total lags used: 10

From the above two models ARIMA (8,1,0) and ARIMA (1,1,8), the former is selected since the reduced form(significant coefficients) clears the Ljung Box test marginally. The p-value achieved is 0.0499 which is very close to 0.05. Thus, we can say that ARIMA (8,1,0) with significant coefficients has residuals having close to zero autocorrelations for all lags.

Following are the details of the model selected:

z test of coefficients:

	Estimate	Std. Error	z value	Pr(>|z|)	
ar1	-0.0315140	0.0156346	-2.0157	0.04384	*
ar2	0.0111592	0.0156417	0.7134	0.47558	
ar3	0.0194733	0.0156421	1.2449	0.21316	
ar4	-0.0137334	0.0156474	-0.8777	0.38012	
ar5	0.0059603	0.0156491	0.3809	0.70330	
ar6	-0.0071616	0.0156450	-0.4578	0.64713	
ar7	0.0312623	0.0156433	1.9984	0.04567	*
ar8	-0.0609983	0.0156446	-3.8990	9.659e-05	

The reduced model coefficients:

z test of coefficients:

	Estimate	Std. Error	z value	Pr(>|z|)	
ar1	-0.032054	0.015626	-2.0513	0.0402333	*
ar7	0.031141	0.015636	1.9916	0.0464140	*
ar8	-0.060740	0.015644	-3.8827	0.0001033	

Signif. codes: 0 ‘*’ 0.001 ‘*’ 0.01 ‘’ 0.05 ‘.’ 0.1 ‘ ’ 1

The Ljung box test results are:

Ljung-Box test

data: Residuals from ARIMA (8,1,0)
Q* = 5.9953, df = 2, p-value = 0.0499

Model df: 8. Total lags used: 10

Final model selected: (With respect to log prices)

Call:
arima(x = ts(nvda_daily_log), order = c(8, 1, 0), fixed = c(NA, 0, 0, 0, 0,
0, NA, NA))

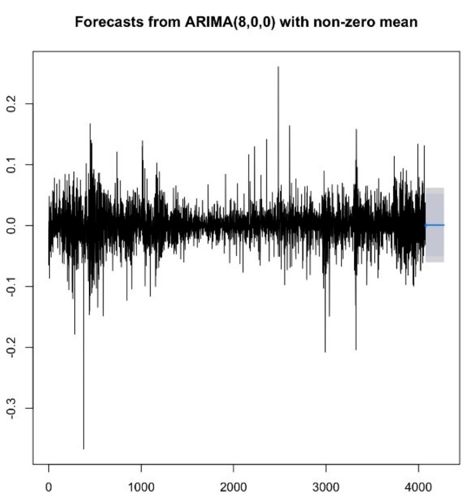
Coefficients:

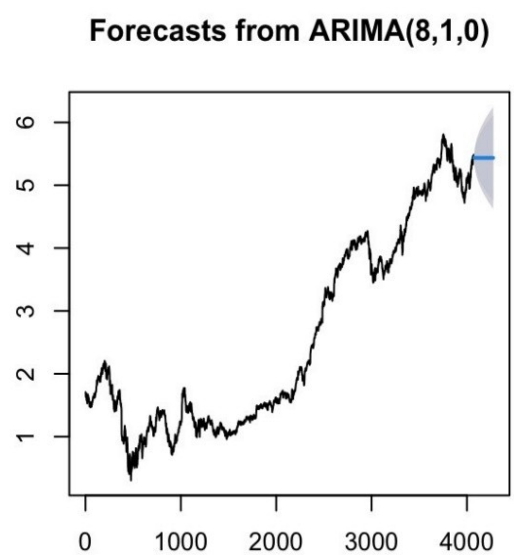
```
         ar1  ar2  ar3  ar4  ar5  ar6     ar7      ar8
     -0.0321    0    0    0    0    0  0.0311  -0.0607
s.e.  0.0156    0    0    0    0    0  0.0156   0.0156
```

sigma^2 estimated as 0.0009612: log likelihood = 8373.07, aic = -16740.13

Forecast ARIMA Model

ARIMA(8,1,0)





Multivariate Analysis (VAR)

VAR Analysis

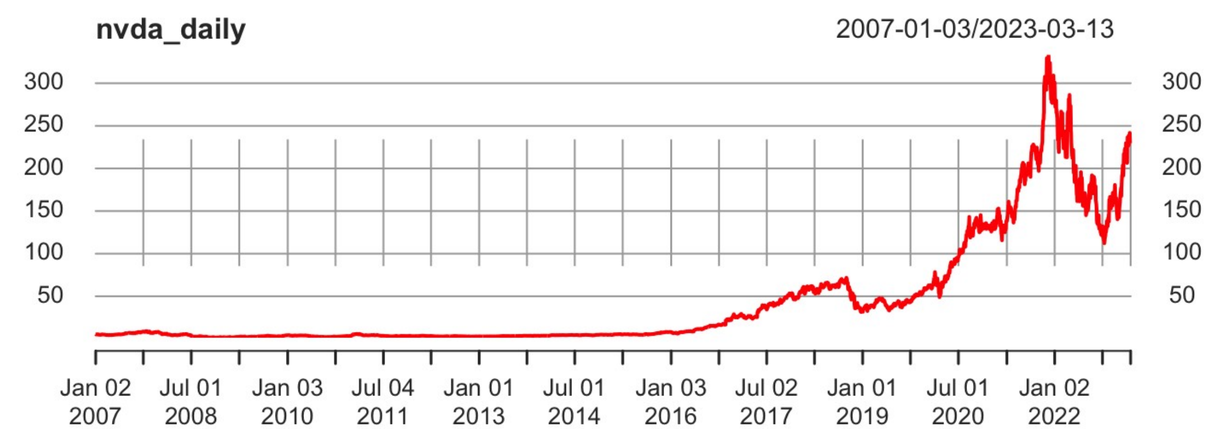
Value at risk (VaR) is a measure of the risk of loss for investments. It estimates how much a set of investments might lose (with a given probability), given normal market conditions, in a set time period such as a day. VaR is typically used by firms and regulators in the financial industry to gauge the amount of assets needed to cover possible losses.

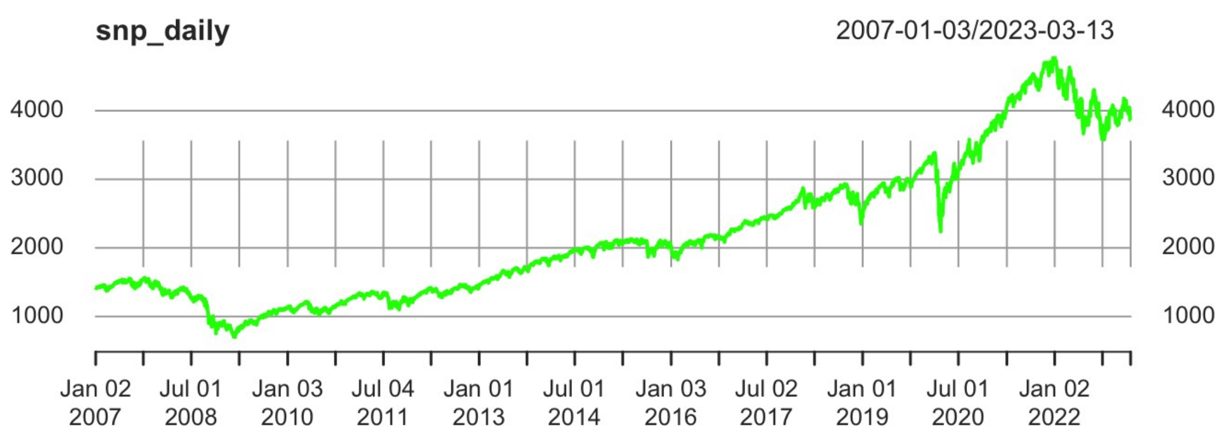
Why joint series analysis?

Price movements in one market can spread easily and instantly to another and market and one must consider them jointly to better understand the dynamic structure of the global finance. The relationship can help in better decision making.

For VAR analysis we will try to compare the Nvidia stock prices with S&P index and check if we can identify any relationship between the two series and finally identify whether a better forecasting can be achieved if we integrate the two.

The below graph shows a times series plot for both Nvidia stocks and S&P index. From the plot, it can be implied that there might be some positive correlation existing.





Regression

Call:

```
lm(formula = unlist(nvda_daily_2) ~ unlist(snp_daily_2))
```

Residuals:

```
   Min      1Q  Median      3Q     Max
-50.976 -21.907  0.741  16.493 143.571
```

Coefficients:

```
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)         -8.749e+01 1.045e+00 -83.72 <2e-16 ***
unlist(snp_daily_2)  5.956e-02  4.306e-04  138.31  <2e-16 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 28.14 on 4074 degrees of freedom
Multiple R-squared:  0.8244,  Adjusted R-squared:  0.8244
F-statistic: 1.913e+04 on 1 and 4074 DF,  p-value: < 2.2e-16
```

From the above results of regression, which was done as preliminary test it can be seen that the prices are very poorly related. Thus, tried to find relation between log returns. Below are the results.

Call:

```
lm(formula = nvda_log_ret ~ snp_log_ret)
```

Residuals:

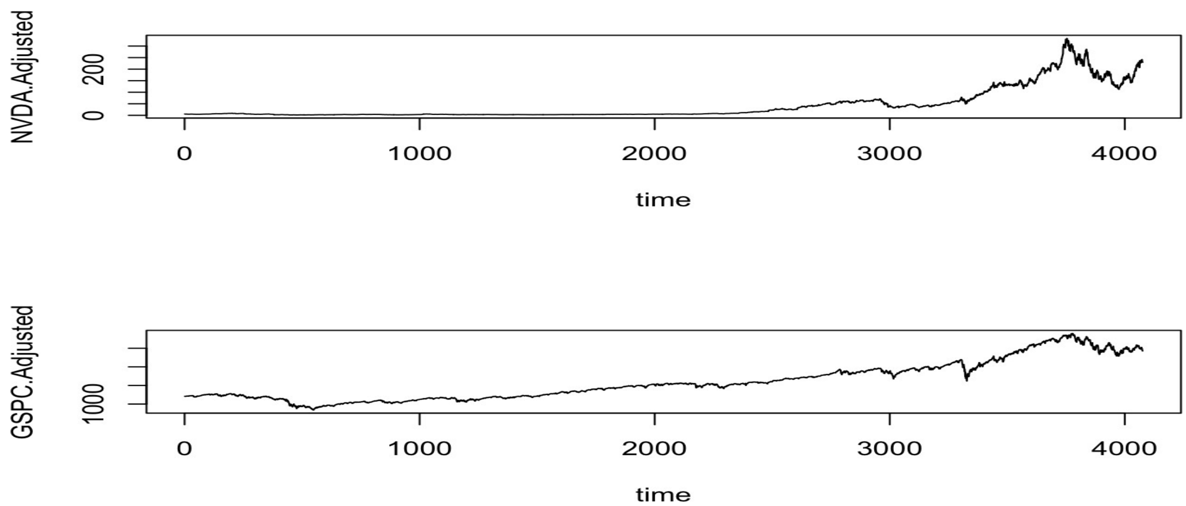
```
    Min       1Q   Median       3Q      Max
-0.36929 -0.01190 -0.00052 0.01114  0.26242
```

Coefficients:

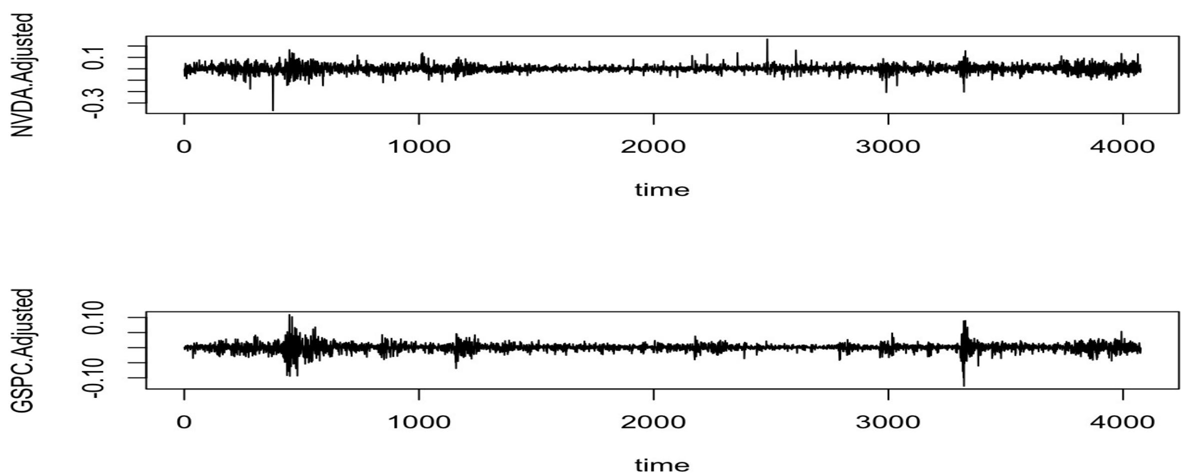
```
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.0005473  0.0003799   1.441     0.15
snp_log_ret 1.4965444  0.0292155  51.224  <2e-16 ***
---
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
Residual standard error: 0.02425 on 4073 degrees of freedom
Multiple R-squared:  0.3918,  Adjusted R-squared:  0.3917
F-statistic:  2624 on 1 and 4073 DF,  p-value: < 2.2e-16
```

There is stronger correlation which can be identified here. We investigate further.
The log daily prices and returns are used to carry out further analysis.

The below two graphs are MTS plots of daily log prices and returns respectively.



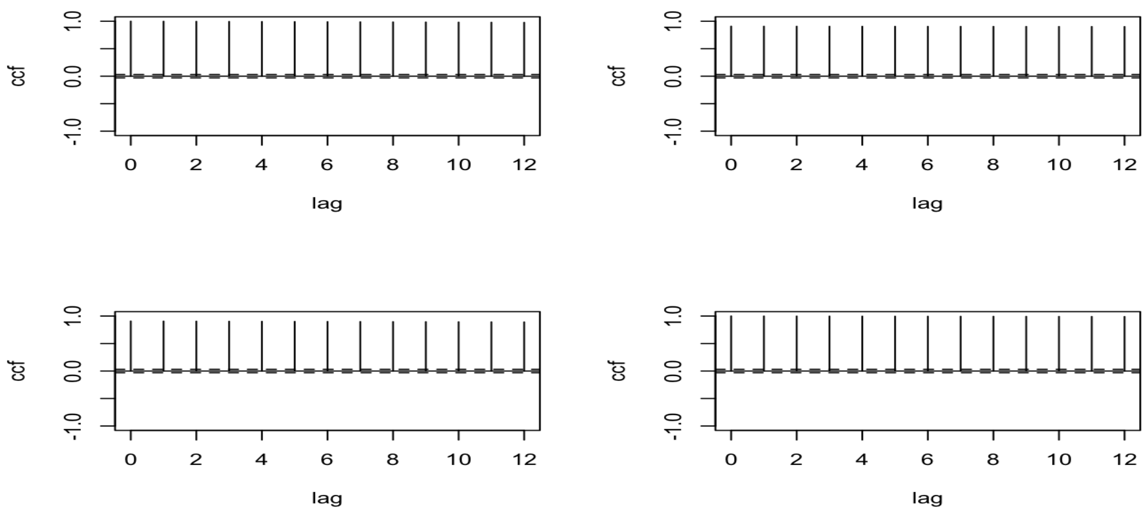
Log daily prices

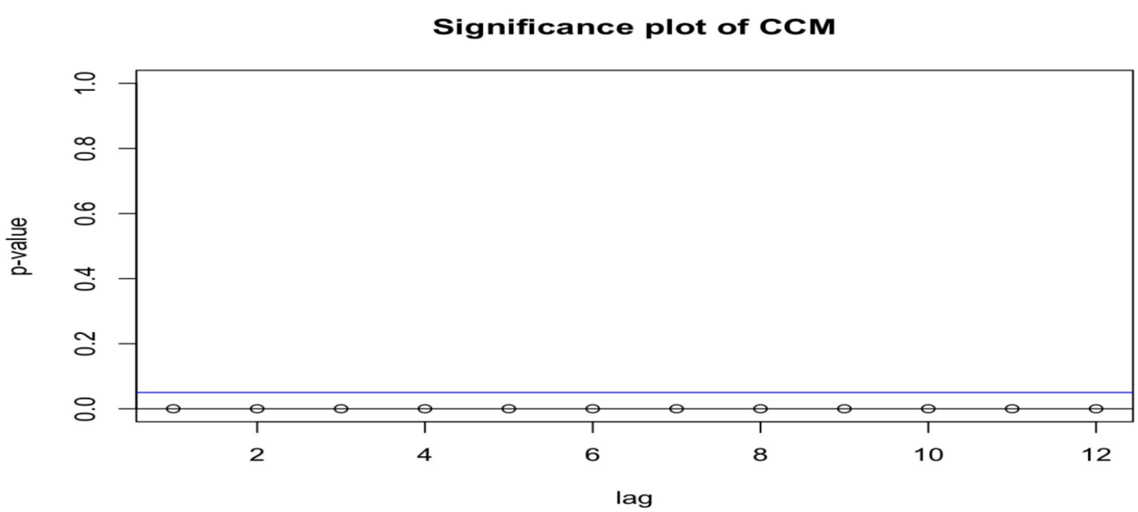


Log daily returns

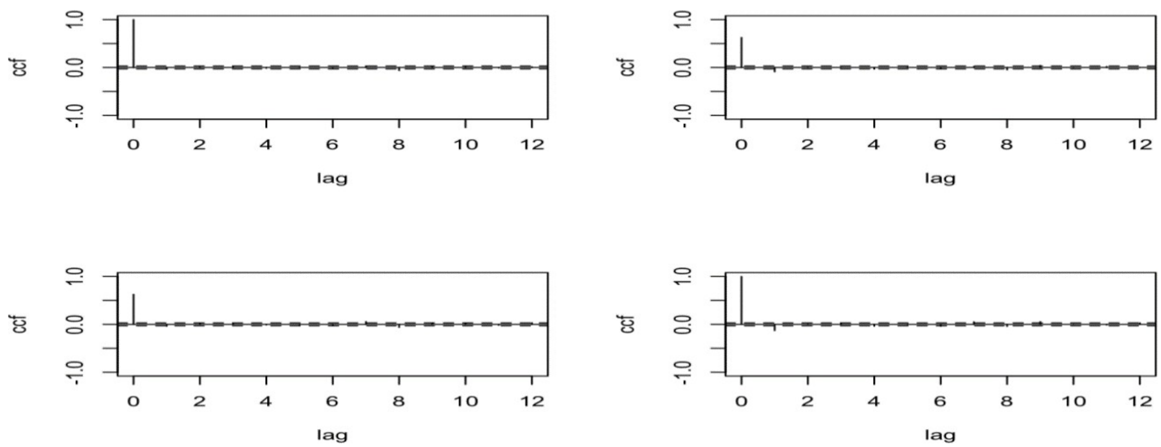
Cross Correlation matrices

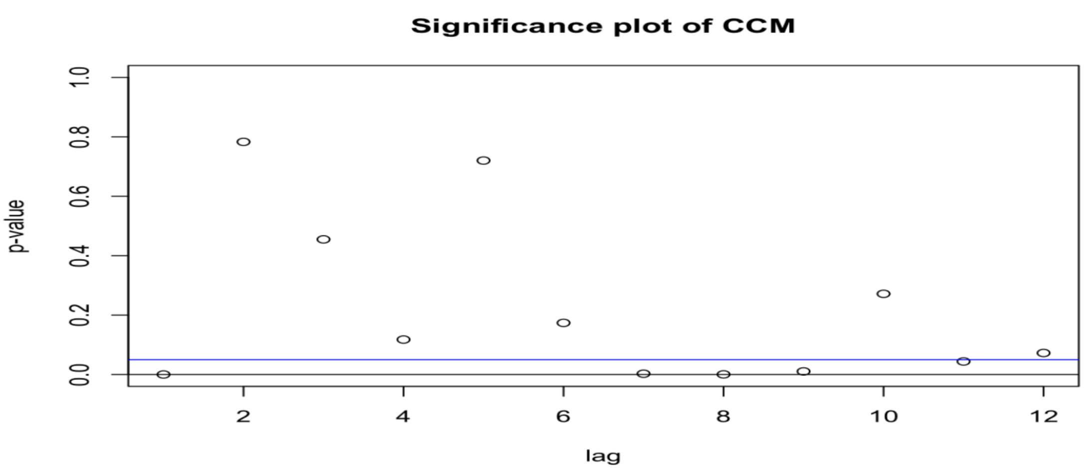
On observing the cross-correlation matrices of both prices it was observed that, all lags had significant autocorrelations individually as well as mutually with respect to each other. The p value plot indicated the same about the significance.





On observing the cross-correlation matrices of both returns it was observed that, not all lags had significant autocorrelations individually as well as mutually with respect to each other. For Nvidia significant correlations can be observed at lags 1,4,7,8 and 9 while for S&P they can be observed at lags 1,4,6,7,8,9 and 12. Thus, further analysis can shed some more light.





```
[1] "Covariance matrix:"
              NVDA.Adjusted GSPC.Adjusted
NVDA.Adjusted      0.000966      0.000253
GSPC.Adjusted      0.000253      0.000169
CCM at lag:  0
      [,1] [,2]
[1,] 1.000 0.626
[2,] 0.626 1.000
```

Simplified matrix:

```
CCM at lag:  1
- -
- -
CCM at lag:  2
. .
. .
CCM at lag:  3
. .
. .
CCM at lag:  4
. -
. -
CCM at lag:  5
. .
. .
CCM at lag:  6
. .
. -
CCM at lag:  7
+ .
+ +
CCM at lag:  8
- -
- -
CCM at lag:  9
. +
. +
CCM at lag:  10
. .
. .
CCM at lag:  11
. .
. .
CCM at lag:  12
. .
+ .
```

For Returns:

On using the VAR order function, based on different criteria such as aic, bic and hq orders 11, 1, and 10 were selected for log returns

selected order: AIC = 11

selected order: BIC = 1

selected order: HQ = 10

Order 11:

AIC = -20.53608
BIC = -20.46792
HQ = -20.51195

Ljung-Box Statistics:

	m	Q(m)	df	p-value
[1,]	1.00000	0.00873	4.00000	1
[2,]	2.00000	0.02419	8.00000	1
[3,]	3.00000	0.05111	12.00000	1
[4,]	4.00000	0.07400	16.00000	1
[5,]	5.00000	0.26864	20.00000	1
[6,]	6.00000	0.57780	24.00000	1
[7,]	7.00000	0.80186	28.00000	1
[8,]	8.00000	1.26115	32.00000	1
[9,]	9.00000	1.57315	36.00000	1
[10,]	10.00000	2.68976	40.00000	1
[11,]	11.00000	3.19393	44.00000	1
[12,]	12.00000	14.56614	48.00000	1
[13,]	13.00000	18.50598	52.00000	1

After Significance testing:

AIC = -20.53547
BIC = -20.50294
HQ = -20.52395

Significant coefficients of model:

	[,1]	[,2]
[1,]	0.00000000	0.000000000
[2,]	0.05844937	0.006547865
[3,]	-1.55188103	-0.183751923
[4,]	0.00000000	0.000000000
[5,]	-0.65283701	0.000000000
[6,]	0.03972279	0.002651206
[7,]	0.00000000	0.000000000
[8,]	-0.07575217	0.000000000
[9,]	0.00000000	-0.032436752
[10,]	-0.03918215	-0.002576822
[11,]	0.00000000	0.000000000
[12,]	0.00000000	0.000000000
[13,]	0.00000000	-0.032153208
[14,]	0.00000000	0.000000000
[15,]	-0.61327946	0.000000000
[16,]	-0.11119386	0.000000000
[17,]	0.77776141	0.000000000
[18,]	0.00000000	-0.003341818
[19,]	0.00000000	0.067637772
[20,]	0.08274979	0.004519428
[21,]	0.00000000	0.000000000
[22,]	-0.06242786	0.000000000
[23,]	0.75677292	0.000000000

Order 1:

AIC = -20.50255
BIC = -20.49636
HQ = -20.50036

Ljung-Box Statistics:

	m	Q(m)	df	p-value
[1,]	1.000	0.299	4.000	0.99

After Significance testing:

AIC = -20.50213
BIC = -20.49594
HQ = -20.49994

Significant coefficients of model:

	[,1]	[,2]
[1,]	0.00000000	0.000000000
[2,]	0.05604962	0.007097692
[3,]	-1.54442404	-0.190176332

Order 10:

AIC = -20.53542
BIC = -20.47346
HQ = -20.51348

Ljung-Box Statistics:

	m	Q(m)	df	p-value
[1,]	1.0000	0.0569	4.0000	1.00
[2,]	2.0000	0.1097	8.0000	1.00

[3,]	3.0000	0.1831	12.0000	1.00
[4,]	4.0000	0.2321	16.0000	1.00
[5,]	5.0000	0.4812	20.0000	1.00
[6,]	6.0000	0.7555	24.0000	1.00
[7,]	7.0000	0.9499	28.0000	1.00
[8,]	8.0000	1.3095	32.0000	1.00
[9,]	9.0000	1.7225	36.0000	1.00
[10,]	10.0000	2.7148	40.0000	1.00
[11,]	11.0000	16.1245	44.0000	1.00
[12,]	12.0000	28.5537	48.0000	0.99
[13,]	13.0000	31.8576	52.0000	0.99

After Significance testing:

AIC = -20.53405
BIC = -20.50462
HQ = -20.52363

Significant coefficients of model:

	[,1]	[,2]
[1,]	0.00000000	0.000000000
[2,]	0.05574196	0.006404221
[3,]	-1.57432424	-0.181141787
[4,]	0.00000000	0.000000000
[5,]	-0.62047575	0.000000000
[6,]	0.04397927	0.002758889
[7,]	0.00000000	0.000000000
[8,]	-0.07519472	0.000000000
[9,]	0.00000000	-0.032910658
[10,]	-0.03947367	-0.002336495
[11,]	0.00000000	0.000000000
[12,]	0.00000000	0.000000000
[13,]	0.00000000	0.000000000
[14,]	0.00000000	0.000000000
[15,]	-0.63068122	0.033022212
[16,]	-0.11275862	0.000000000
[17,]	0.77129137	0.000000000
[18,]	0.00000000	-0.003376775
[19,]	0.00000000	0.067509317
[20,]	0.07957168	0.004449275
[21,]	0.00000000	0.000000000

From all the above models, the selection of final model is done on the basis of AIC and BIC comparison. The best model should be complex enough to capture key details of the data, yet not too complex to overfit the data. AIC tends to pick up more complex models with increase in number of data points since it penalizes by a constant factor of 2. But, BIC penalizes it by a factor of the natural log of the number of data points, i.e. ln(n). For forecast a model capable of capturing details is important. Order 11 and 10 give good AIC scores while order 1 has good BIC score. If models based on order 10 and 11 are reduced with significant coefficients, it can be observed that both models achieve a better BIC score as well. Since the scores are marginally different, we can pick any of the models. But using least AIC score to select the VAR(11) model for demonstration.

The final equation of the model is:

VAR Estimation Results:
=========================

Estimated coefficients for equation y1:
===
Call:

y1 = y1.l1 + y2.l1 + y2.l2 + y1.l3 + y1.l4 + y1.l5 + y2.l7 + y1.l8 + y2.l8 + y1.l10 + y1.l11 + y2.l11

```
     y1.l1       y2.l1       y2.l2       y1.l3       y1.l4       y1.l5       y2.l7
 0.05845063 -1.55189273 -0.65283750 0.03972361 -0.07575304 -0.03918155 -0.61327823
     y1.l8       y2.l8      y1.l10      y1.l11      y2.l11
-0.11119291 0.77775291  0.08274974 -0.06242788  0.75677789
```

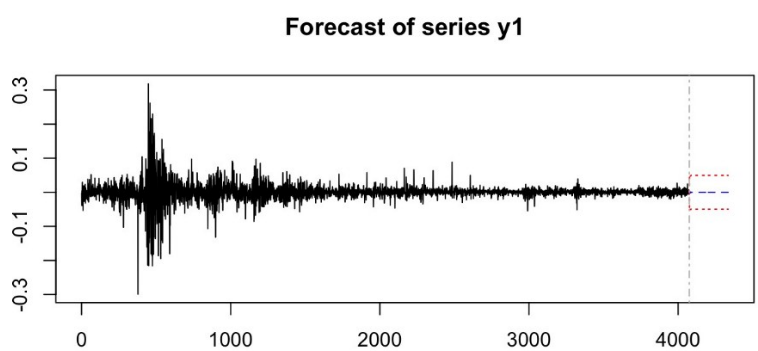
Estimated coefficients for equation y2:
==
Call:
y2 = y1.l1 + y2.l1 + y1.l3 + y2.l4 + y1.l5 + y2.l6 + y1.l9 + y2.l9 + y1.l10

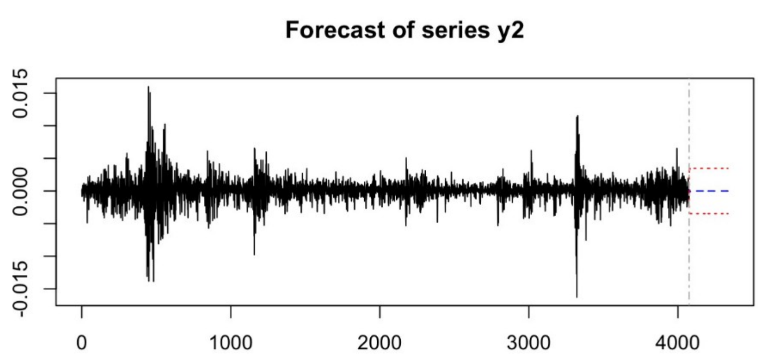
```
      y1.l1        y2.l1        y1.l3        y2.l4        y1.l5        y2.l6        y1.l9
 0.006547900 -0.183752196 0.002651207 -0.032436763 -0.002576802 -0.032153175 -
0.003341842
      y2.l9       y1.l10
 0.067637968  0.004519433
```

Forecast

Returns:

VAR(11)





Co-integration test

A co-integration test is performed to check if combination of two series is stationary. Here, both log of S&P index and log of Nvidia are non-stationary. This is an experiment ton check for co-integrations.

Basic idea behind using co-integrating is leveraging the power of stationary series.
The following are the reason:
1. Stationary series is mean reverting.
2. Long term forecasts of the "linear" combination converge to a mean value. In long term they can be linearly related
3. This mean-reverting property has many applications. For instance, pair trading in finance.

The Johansen test is performed for testing co-integration.

```
######################
# Johansen-Procedure #
######################

Test type: maximal eigenvalue statistic (lambda max) , with linear trend

Eigenvalues (lambda):
[1] 0.3701506 0.3334035
```

Values of test statistic and critical values of test:

```
          test 10pct  5pct  1pct
r <= 1 | 1651.89  6.50  8.18 11.65
r = 0  | 1882.84 12.91 14.90 19.19

Eigenvectors, normalised to first column:
(These are the cointegration relations)

                 NVDA.Adjusted.l2 GSPC.Adjusted.l2
NVDA.Adjusted.l2        1.000000        1.0000000
GSPC.Adjusted.l2       -7.465588       -0.8726174

Weights W:
(This is the loading matrix)

                 NVDA.Adjusted.l2 GSPC.Adjusted.l2
NVDA.Adjusted.d         0.1731348       -1.1183954
GSPC.Adjusted.d         0.1775606       -0.1343915
```

From the above results, for null hypothesis r= 0(rank) since the test value is more than critical values at all threshold levels, null hypothesis is rejected. Also, for r<= 1, a similar inference can be made. This implies that, co-integration cannot exist for these two series combined alone. Either more series' can be combined or the series can be replaced with some other. It can be concluded from the performed analysis that, S&P index and Nvidia returns if combined cannot form a stationary series and thus they both cannot be used infer results based on each other.

GARCH Analysis

Modelling done on time series of NVIDIA log returns for the period of 2007 up until the current date. Log returns series taken because of stationarity of the returns and subsequent good results while modelling the series for both the AR and MA models.
The selected model for our ARMA analysis has been model with order (8,0,0)

To further complement the forecasting done on the NVIDIA stock we have done GARCH analysis to add to our ARMA (8,0) model. ARMA models do not take the volatility component into account hence we do GARCH modelling on our time series in order to make sure that the shocks and their subsequent effects are taken into account and that the variance of the stock is used as well when we forecast our time series. will aid us in the accuracy of our forward looking forecast. ARMA and ARCH models can be used together to describe conditional mean and conditional variance respectively. The benefit of the GARCH model over the ARCH model is that the GARCH model takes into account all of the previous returns when forecasting the modelhence the GARCH model is beneficial as it allows us to use all of the previous squared returns plus as well as the most recent ones. The GARCH model allows us to predict both the volatility and the returns of the stock it also enables us to make sure that the stock contains all of the components of a mean prediction model and also takes into account the components of a volatility prediction model.
To set up our GARCH model we used the residuals of the squared and the absolute residuals of our ARMA (8,0) model. We then plotted the ACF and PACF of both the absolute and the squared residuals. The squared and the absolute values do not have the negative instances of the dataset and hence are not impacted by the negative values in the end instance.

We run the GARCH model. We can start with the standard GARCH model where we consider th e conditional error term is a normal distribution.

We used function ugarchspec() for the model specification and ugarchfit() for the model fitting. For the standard
GARCH model, we specify a constant to mean ARMA model, which means that arma0rder =c(1,
We consider the GARCH(1,1) model and the distribution of the conditional error term is the nor mal distribution. Below of a picture of the base GARCH (1,1) model which was used to make the forecast.

We tested our time series for ARCH effect, and found out that ARCH effect was present which prompted us to further use the time series for modelling as if the ARCH effect was not present, our GARCH model would be invalid. Thus, we went ahead and used the ugarchspec() function to set up our base GARCH model for which we used order(1,1) as our base lag order. We then changed the orders multiple times until finally settling on GARCH(3,1) which gave us the lowest AIC value of all the GARCH orders that we used.

We then decided to do a normal GARCH analysis with ARMA order (8,0) and GARCH order (3,2) with our base GARCH model.

We were unsure as to what distribution to use, so we decided to run the baseline GARCH model on all the 4 types of distributions having run the baseline GARCH model on all the 4 types of distributions, we found that our QQ plot was not normal for any of these particular distributions, thus we decided to do a selection according to the AIC of all of these distributions and choose the one with the lowest AIC, which was the student t distribution. Thus, we decided to proceed further with this particular distribution.

We then used all of these 4 GARCH variations with a student t distribution to see which one had the lowest AIC of all of the 4. In our particular model checking, we found that EGARCH had the lowest AIC of -4.4686.

Thus we decided to proceed forward with forecasting the EGARCH(3,2) model with student t distribution. We decided not to use the GARCH M model because the ARCH M was not significant in the GARCH M model. We decided to forecast the returns using the bootstrapping method and our bootstrap forecast was divided into two parts – sigma and series forecast. The sigma forecast was for the prediction of volatility and the series forecast was to predict the returns. Our sigma and series forecast predict the variance for both the returns and volatility of the NVDA stock. Our sigma forecast predicted a variance of 0.20 either side of the mean.

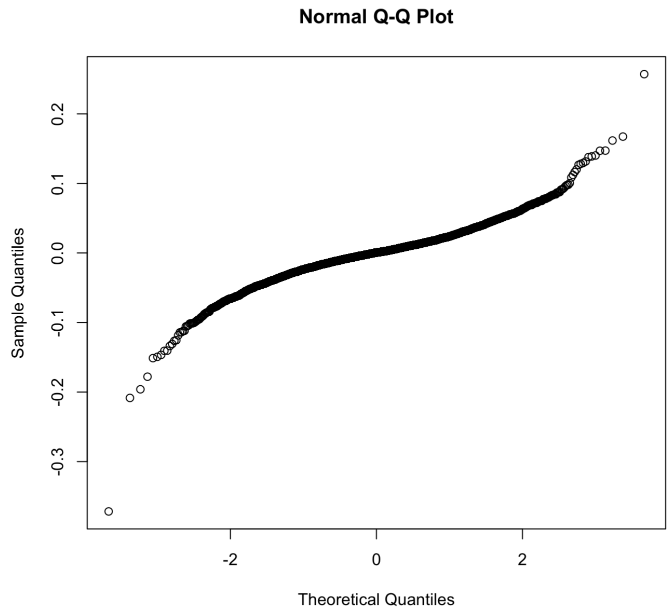
Ljung box test for GARCH

Re-iteration of the ARIMA model chosen.

data: Residuals from ARIMA(8,0,0) with non-zero mean
Q* = 6.0099, df = 2, p-value = 0.04954

Model df: 8. Total lags used: 10

QQ plot of residuals:



ARCH test

The first step in the GARCH analysis is using an ARCH test or Ljung box test on the residuals to identify weather a volatility model can be fitted.
Uncorrelated time series can still be serially dependent due to a dynamic conditional variance process. A time series exhibiting conditional heteroscedasticity—or autocorrelation in the squared series—is said to have *autoregressive conditional heteroscedastic* (ARCH) effects. Engle's ARCH test is a Lagrange multiplier test to assess the significance of ARCH effects. Since the p value is less than 0.05, null hypothesis is rejected and thus ARCH effect exists and we can continue to volatility modelling.

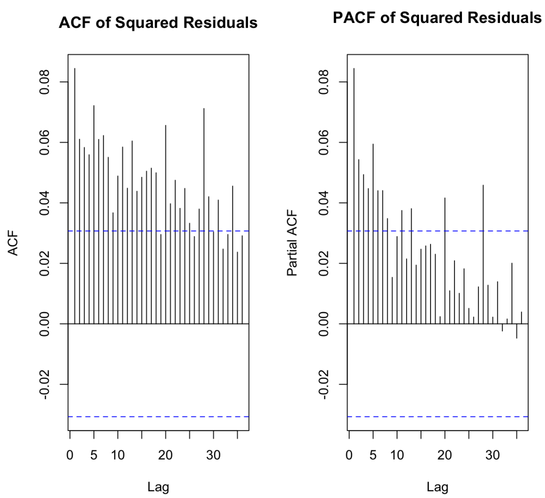
Q(m) of squared series(LM test):
Test statistic: 0.02572404 p-value: 0
Rank-based Test:
Test statistic: 1251.192 p-value: 0

Q(m) of squared series(LM test):
Test statistic: 150.859 p-value: 0
Rank-based Test:
Test statistic: 1251.192 p-value: 0

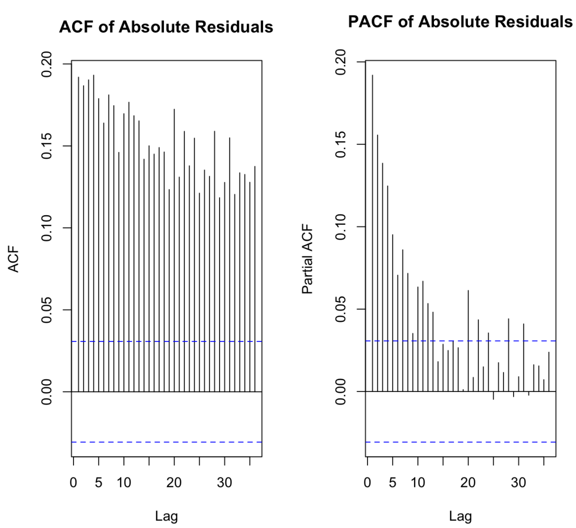
ACF and PACF tests

To set up our GARCH model we used the residuals of the squared and the absolute residuals of our ARMA (8,0) model. We then plotted the ACF and PACF of both the absolute and the squared residuals. The squared and the absolute values do not have the negative instances of thedataset and hence are not impacted by the negative values in the end instance. It can be observed that there is no clear picture of the order to be selected for the GARCH model since even though the ACF of residuals show non zero autocorrelations, looking at the 2 graphs, there is a lot of ambiguity regarding the specific order for the GARCH or ARCH. Below are the generated plots.

ACF and PACF of squared residuals:



ACF and PACF of absolute residuals:



EACF test for GARCH

Since we were not able to infer the order from the above analysis, we move further with the EACF plots.

Squared residuals:

```
AR/MA
  0 1 2 3 4 5 6 7 8 9 10 11 12 13
0 x x x x x x x x x x x  x  x  x
1 x o o o o o o o o o o  o  o  o
2 x o o o o o o o o o o  o  o  o
3 x o x o o o o o o o o  o  o  o
4 x x x x o o o o o o o  o  o  o
5 x x x x x o o o o o o  o  o  o
6 x x x x x x o o o o o  o  o  o
7 x x x x x x o o o o o  o  o  o
```

Absolute residuals:

```
AR/MA
  0 1 2 3 4 5 6 7 8 9 10 11 12 13
0 x x x x x x x x x x x  x  x  x
1 x o o o o o o o o o o  o  o  o
2 x x o o o o o o x o o  o  o  o
3 x o x o o o o o o o o  o  o  o
4 x x x x o o o o o o o  o  o  o
5 x x x x x o o o o o o  o  o  o
6 x x x x o x o o o o o  o  o  o
7 x x x x o x x o o o o  o  o  o
```

Models selected based on the above considerations:

We tested the AIC for the top 5 models which were of orders 1,1/3,1/2,1/2,2/3,2/3,1 and we saw that the least AIC was for the two models listed below. We thus settled on the two models listed below and found that the GARCH 3,1 and 1,1 orders had the lowest AIC and we proceeded to model further with these two orders.

Two models:

1. Model:

GARCH(1,1)

Residuals:

```
      Min        1Q    Median        3Q       Max
-12.44868  -0.52505   0.02308   0.55173  10.52852
```

Coefficient(s):

```
   Estimate  Std. Error  t value Pr(>|t|)
a0 1.680e-05   1.283e-06     13.1   <2e-16 ***
a1 6.919e-02   3.460e-03     20.0   <2e-16 ***
b1 9.169e-01   4.135e-03    221.8   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Diagnostic Tests:

Jarque Bera Test

data: Residuals
X-squared = 26014, df = 2, p-value < 2.2e-16

Box-Ljung test

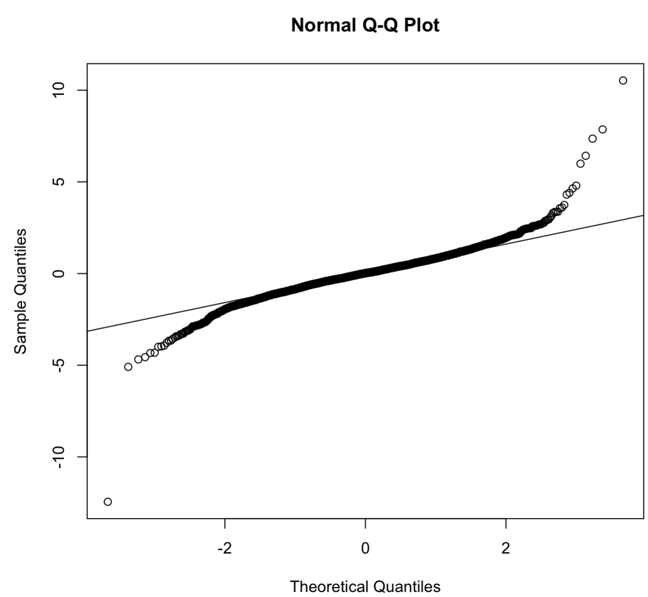
data: Squared.Residuals
X-squared = 0.056655, df = 1, p-value = 0.8119

z test of coefficients:

```
    Estimate Std. Error z value  Pr(>|z|)
a0 1.6805e-05 1.2826e-06  13.102 < 2.2e-16 ***
a1 6.9192e-02 3.4597e-03  20.000 < 2.2e-16 ***
b1 9.1690e-01 4.1347e-03 221.758 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Box-Ljung test

data: residuals(m1)
X-squared = 8.7847, df = 12, p-value = 0.7212



2. Model:

GARCH(3,1)

Residuals:

```
     Min        1Q    Median       3Q       Max
-11.83128  -0.51857   0.02333   0.55488   9.73156
```

Coefficient(s):

```
   Estimate  Std. Error  t value Pr(>|t|)
a0 2.078e-05   1.776e-06   11.703  < 2e-16 ***
a1 1.173e-01   5.097e-03   23.011  < 2e-16 ***
b1 1.054e-01   2.268e-02    4.648 3.35e-06 ***
b2 1.174e-02   1.730e-02    0.678    0.497
b3 7.503e-01   1.995e-02   37.608  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Diagnostic Tests:
Jarque Bera Test

data: Residuals
X-squared = 18440, df = 2, p-value < 2.2e-16

Box-Ljung test

data: Squared.Residuals
X-squared = 0.049189, df = 1, p-value = 0.8245

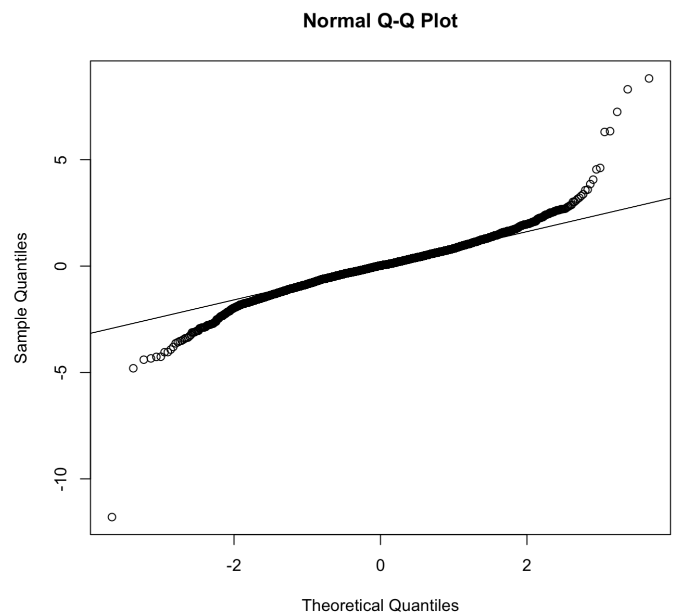
> coeftest(m4)

z test of coefficients:

```
    Estimate Std. Error z value  Pr(>|z|)
a0 2.0779e-05 1.7755e-06 11.7031 < 2.2e-16 ***
a1 1.1729e-01 5.0974e-03 23.0107 < 2.2e-16 ***
b1 1.0543e-01 2.2682e-02  4.6484 3.346e-06 ***
b2 1.1739e-02 1.7303e-02  0.6784    0.4975
b3 7.5025e-01 1.9949e-02 37.6084 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Box-Ljung test

data: residuals(m4)
X-squared = 7.7161, df = 12, p-value = 0.8069



Baseline:

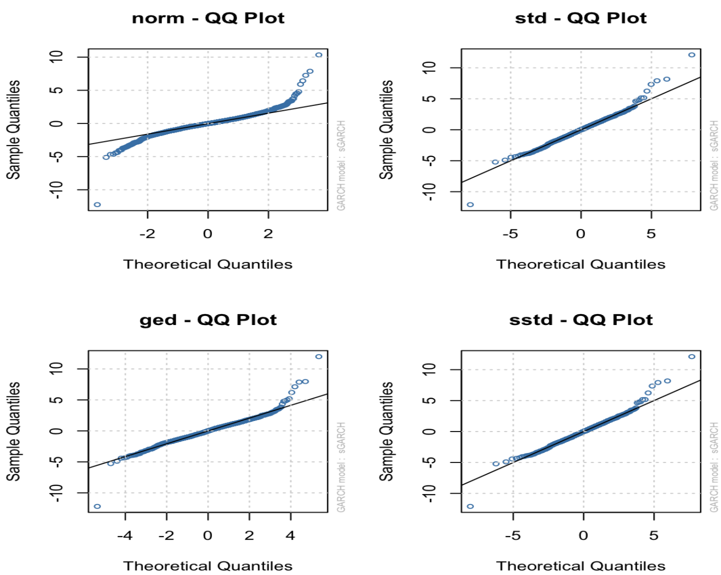
A baseline is a fixed point of reference that is used for comparison purposes. In business, the success of a project or product is often measured against a baseline number for costs, sales, or any number of other variables. A project may exceed a baseline number or fail to meet it.

```
*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model     : sGARCH(3,1)
Mean Model      : ARFIMA(8,0,0)
Distribution    : std

Optimal Parameters
-----------------------------------
        Estimate  Std. Error    t v
mu      0.001298    0.000308   4.21
ar1    -0.026311    0.015009  -1.75
ar2     0.014485    0.014926   0.97
ar3     0.003377    0.014986   0.22
ar4    -0.008180    0.014852  -0.55
ar5     0.003526    0.014634   0.24
ar6    -0.007737    0.014507  -0.53
ar7     0.001553    0.014624   0.16
ar8    -0.052737    0.014543  -3.62
omega   0.000003    0.000002   1.34
alpha1  0.047193    0.017219   2.74
alpha2  0.000000    0.028670   0.00
alpha3  0.000000    0.023946   0.00
beta1   0.951807    0.006142 154.96
shape   4.481322    0.315786  14.19
       Pr(>|t|)
mu      0.000025
ar1     0.079609
ar2     0.331841
ar3     0.821703
ar4     0.581820
ar5     0.809579
ar6     0.593808
ar7     0.915437
ar8     0.000287
omega   0.177759
alpha1  0.006129
alpha2  0.999996
alpha3  0.999997
beta1   0.000000
shape   0.000000

LogLikelihood : 90

Information Criter
------------------

Akaike       -4.45
Bayes        -4.42
Shibata      -4.45
Hannan-Quinn -4.44
```

```
Asymptotic Critical Values (10% 5% 1%)
Joint Statistic:         3.26 3.54 4.07
Individual Statistic:    0.35 0.47 0.75

Sign Bias Test
------------------------------------
                     t-value    prob sig
Sign Bias          1.3761273 0.16886
Negative Sign Bias 1.1897994 0.23419
Positive Sign Bias 0.0001524 0.99988
Joint Effect       8.6766880 0.03391  **
```

QQ plots of residuals for different distributions

Distribution: In this part of our project we plotted the qqplot with baseline garch model for different distributions to arrive at the right distribution that we needed a plot that was centered around the line of the mean but we found that none of the qqplots gave us the right fit around the line of the mean and hence we used AIC as our parameter to judge our distributions as we can see from the table with different AICs that we have the lowest AIC in the distribution with the student T Distribution. Hence we proceeded to model our time series with that particular distribution



normal
Akaike -4.2873
Bayes -4.2687
Shibata -4.2874
Hannan-Quinn -4.2808

std
Akaike -4.4528
Bayes -4.4327
Shibata -4.4528
Hannan-Quinn -4.4457

ged
Akaike -4.4299
Bayes -4.4098
Shibata -4.4300
Hannan-Quinn -4.4228

sstd
Akaike -4.4526
Bayes -4.4309
Shibata -4.4526
Hannan-Quinn -4.4449

Different models of std distribution

After picking the student T distribution we decided to model our time series with the one with the best AIC out of all of the four variations of the GARCH model we then plotted the AIC all of the EGARCH, TGARCH, IGARCH and MGARCH models and as you can see in our AIC table the AIC for the EGARCH model is the lowest at -4.86 we also further analyzed the models to find other redundancies and found faults in other models like the nonexistence of the ARCH-M effect in the M-GARCH model and hence through a process of elimination we decided on the EGARCH model for further modeling

```
*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model     : fGARCH(3,1)
fGARCH Sub-Model        : TGARCH
Mean Model      : ARFIMA(8,0,0)
Distribution    : std

Optimal Parameters
------------------------------------
        Estimate  Std. Error   t value Pr(>|t|)
mu      0.001075    0.000317   3.38820 0.000704
ar1    -0.022317    0.014973  -1.49044 0.136109
ar2     0.018093    0.009204   1.96574 0.049328
ar3     0.008742    0.014717   0.59403 0.552491
ar4    -0.005116    0.014620  -0.34991 0.726410
ar5     0.005691    0.014584   0.39020 0.696387
ar6    -0.004241    0.013717  -0.30914 0.757217
ar7     0.003979    0.014360   0.27711 0.781698
ar8    -0.049328    0.014197  -3.47445 0.000512
omega   0.000263    0.000082   3.21962 0.001284
alpha1  0.071750    0.008222   8.72619 0.000000
alpha2  0.000000    0.000022   0.00077 0.999385
alpha3  0.008243    0.003138   2.62711 0.008611
beta1   0.937885    0.001058 886.65115 0.000000
eta11   0.379279    0.040490   9.36724 0.000000
eta12   0.212292    0.597093   0.35554 0.722183
eta13  -0.999998    0.297262  -3.36402 0.000768
shape   4.569412    0.318437  14.34951 0.000000


Information Criteria
----------------------

Akaike       -4.4663
Bayes        -4.4384
Shibata      -4.4663
Hannan-Quinn -4.4564
```

Jarque Bera Test

data: res_spec1
X-squared = 18671, df = 2, p-value < 2.2e-16

```
*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model     : eGARCH(3,1)
Mean Model      : ARFIMA(8,0,0)
Distribution    : std

Optimal Parameters
------------------------------------
        Estimate  Std. Error     t value Pr(>|t|)
mu      0.001134    0.000414    2.739583 0.006152
ar1    -0.025950    0.015476   -1.676770 0.093588
ar2     0.019812    0.014907    1.329050 0.183831
ar3     0.008245    0.014982    0.550334 0.582090
ar4    -0.006858    0.010433   -0.657340 0.510962
ar5     0.002321    0.016098    0.144194 0.885347
ar6    -0.006443    0.009726   -0.662455 0.507680
ar7     0.001690    0.021271    0.079462 0.936665
ar8    -0.051947    0.014190   -3.660730 0.000251
omega  -0.052624    0.006229   -8.448627 0.000000
alpha1 -0.096532    0.027152   -3.555274 0.000378
alpha2  0.000535    0.037533    0.014247 0.988633
alpha3  0.064336    0.028104    2.289203 0.022068
beta1   0.992726    0.000898 1106.072038 0.000000
gamma1  0.184908    0.046450    3.980810 0.000069
gamma2 -0.046754    0.058410   -0.800449 0.423450
gamma3 -0.014092    0.039810   -0.353967 0.723364
shape   4.612106    0.338601   13.621067 0.000000
Information Criteria
----------------------

Akaike       -4.4686
Bayes        -4.4408
Shibata      -4.4687
Hannan-Quinn -4.4588
```

Jarque Bera Test

data: res_spec2
X-squared = 15591, df = 2, p-value < 2.2e-16

Garch M:

```
*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model     : eGARCH(3,1)
Mean Model      : ARFIMA(8,0,0)
Distribution    : std

Optimal Parameters
------------------------------------
        Estimate  Std. Error     t value Pr(>|t|)
mu      0.000722    0.000406    1.777441 0.075496
ar1    -0.024944    0.018123   -1.376402 0.168697
ar2     0.020979    0.013068    1.605379 0.108410
ar3     0.008337    0.014516    0.574304 0.565762
ar4    -0.006714    0.017493   -0.383838 0.701098
ar5     0.002757    0.008950    0.308062 0.758035
ar6    -0.006677    0.010342   -0.645622 0.518524
ar7     0.001754    0.010032    0.174886 0.861169
ar8    -0.051605    0.012183   -4.235854 0.000023
archm   0.017876    0.014755    1.211536 0.225690
omega  -0.054409    0.005727   -9.500534 0.000000
alpha1 -0.095973    0.026994   -3.555338 0.000377
alpha2  0.000122    0.037173    0.003273 0.997389
alpha3  0.064303    0.027899    2.304845 0.021175
beta1   0.992485    0.000823 1206.476468 0.000000
gamma1  0.184530    0.044747    4.123851 0.000037
gamma2 -0.046571    0.057459   -0.810505 0.417650
gamma3 -0.013869    0.039631   -0.349947 0.726379
shape   4.606449    0.336585   13.685836 0.000000
```

```
LogLikelihood : 9122.985

Information Criteria
----------------------------

Akaike       -4.4682
Bayes        -4.4388
Shibata      -4.4683
Hannan-Quinn -4.4578
```

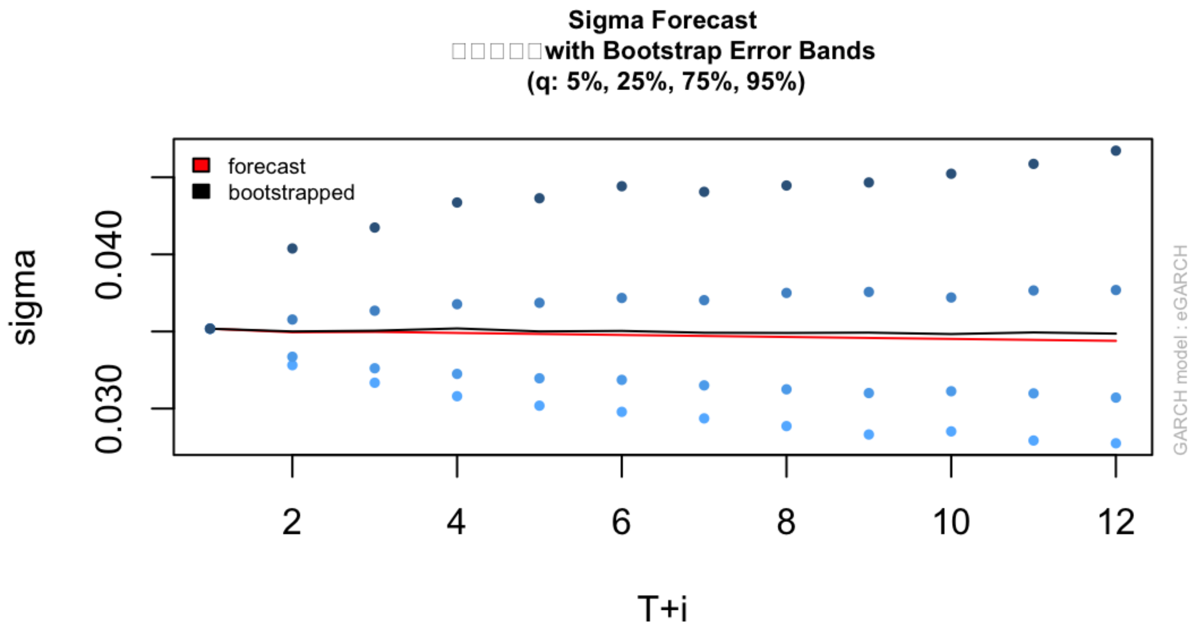
EGARCH Forecast

Our final EGARCH forecast allowed us to see the minimum mean maximum and 25th and 75th quantile variances from the GARCH forecast for both the return and the volatility of the log returns of NVDA stock For the next ten days in the time series the stock does not show much changes in both the volatility and the return which seem to be pretty close to the mean further volatilies and returns could be expected by a lomger term analysis

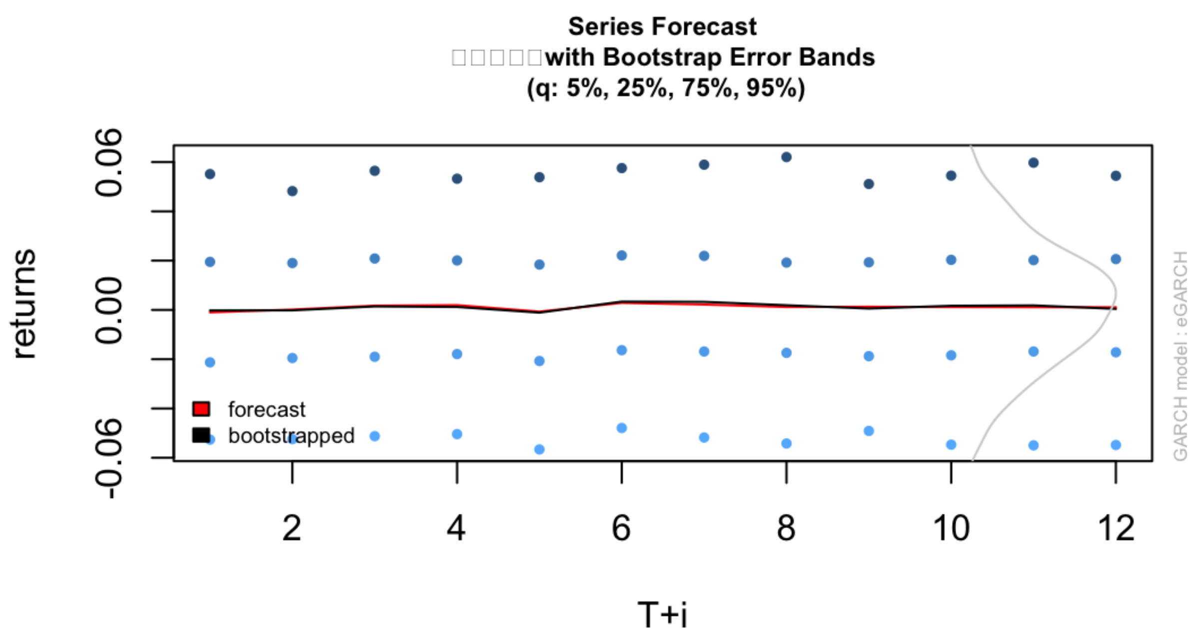
```
*------------------------------------*
*     GARCH Bootstrap Forecast       *
*------------------------------------*
Model : eGARCH
n.ahead : 12
Bootstrap method:  partial
Date (T[0]): 2023-03-13

Series (summary):
         min     q.25      mean     q.75     max forecast[analytic]
t+1  -0.12531 -0.021286 -0.000216 0.019495 0.24248          -0.001002
t+2  -0.17347 -0.019520 -0.000169 0.019030 0.27299           0.000143
t+3  -0.19404 -0.019011  0.001419 0.020859 0.30007           0.001719
t+4  -0.13343 -0.017917  0.001201 0.020085 0.23219           0.001950
t+5  -0.15055 -0.020730 -0.001102 0.018394 0.32764          -0.000684
t+6  -0.16183 -0.016338  0.003373 0.022106 0.24181           0.002860
t+7  -0.19189 -0.016885  0.003274 0.021923 0.31539           0.002179
t+8  -0.14605 -0.017428  0.001899 0.019209 0.27646           0.001182
t+9  -0.12829 -0.018767  0.000555 0.019317 0.29885           0.001288
t+10 -0.14009 -0.018425  0.001681 0.020315 0.28345           0.001171
.....................

Sigma (summary):
         min    q0.25     mean    q0.75      max forecast[analytic]
t+1  0.035182 0.035182 0.035182 0.035182 0.035182           0.035182
t+2  0.032681 0.033361 0.035003 0.035776 0.053732           0.034951
```

t+3	0.031186	0.032615	0.035050	0.036347	0.064870	0.034975
t+4	0.029991	0.032249	0.035197	0.036772	0.076530	0.034908
t+5	0.029135	0.031964	0.035005	0.036860	0.069592	0.034841
t+6	0.027817	0.031862	0.035039	0.037174	0.065759	0.034776
t+7	0.027435	0.031505	0.034919	0.037031	0.068376	0.034710
t+8	0.026576	0.031250	0.034905	0.037501	0.072615	0.034646
t+9	0.025847	0.031006	0.034926	0.037565	0.063122	0.034581
t+10	0.025654	0.031124	0.034833	0.037204	0.059994	0.034518

Forecasts:



Sigma Forecast
□□□□□□with Bootstrap Error Bands
(q: 5%, 25%, 75%, 95%)
forecast
bootstrapped
sigma
0.030
0.040
2
4
6
8
10
12
T+i
GARCH model : eGARCH



Series Forecast
□□□□□□with Bootstrap Error Bands
(q: 5%, 25%, 75%, 95%)
forecast
bootstrapped
returns
-0.06
0.00
0.06
2
4
6
8
10
12
T+i
GARCH model : eGARCH

VaR

Value at risk (VaR) is a way to quantify the risk of potential losses for a firm or an investment.
This metric can be computed in three ways: the historical, variance-covariance, and Monte Carlo methods. Investment banks commonly apply VaR modeling to firm-wide risk due to the potential for independent trading desks to unintentionally expose the firm to highly correlated assets.

Expected shortfall, also known as conditional value at risk or cVaR, is a popular measure of tail risk. One shortcoming of value at risk (VaR) is that it does not tell us anything about losses beyond the VaR level. Expected shortfall is what we expect the loss to be, on average, when a fund exceeds its VaR level. If we are measuring VaR at the 95% confidence level, then the expected shortfall would be the mean loss in the 5% of scenarios where the fund exceeds its VaR.

Few methods were explored for calculating VaR and ES for nvidia stock:

Risk Metrics:

RiskMetrics is a method for calculating the potential downside risk of a single investment or an investment portfolio. The method assumes that an investment's returns follow a normal distribution over time. It provides an estimate of the probability of a loss in an investment's value during a given period of time.

```
Coefficient(s):
      Estimate  Std. Error  t value   Pr(>|t|)
beta 0.97903040  0.00202132  484.352 < 2.22e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

 Volatility prediction:
    Orig      Vpred
[1,] 4075 0.03698022

 Risk measure based on RiskMetrics:
      prob        VaR         ES
[1,] 0.950 0.06082706 0.07627958
[2,] 0.990 0.08602887 0.09856022
[3,] 0.999 0.11427748 0.12451575
```

To give an example, if we are making an investment of $10,000 the VaR for 95%, 99% and 99.9% would be:

95% VAR : $10000 * 0.06082706 = **$ 608.2706**
99% VAR : $10000 * 0.08602887 = **$ 860.2887**
99.9% VAR: $10000 * 0.1142774 = **$ 1142.774**

Default Estimate:

Default beta = 0.96 is used.

```
 Volatility prediction:
     Orig     Vpred
[1,] 4075 0.0360755

 Risk measure based on RiskMetrics:
      prob        VaR         ES
[1,] 0.950 0.05933892 0.07441339
[2,] 0.990 0.08392416 0.09614893
[3,] 0.999 0.11148167 0.12146946
```

Economic Modelling

Econometric models are constructed from economic data with the aid of the techniques of statistical inference. These models are usually based on economic theories that assume optimizing behavior on the part of economic agents.

```
Conditional Distribution:
 std

Coefficient(s):
        mu       omega      alpha1      alpha2      alpha3       beta1       shape
1.2106e-03  3.2392e-06  5.0356e-02  1.0000e-08  1.0000e-08  9.4951e-01  4.4035e+00

Std. Errors:
 based on Hessian

Error Analysis:
        Estimate  Std. Error  t value Pr(>|t|)
mu     1.211e-03   3.285e-04    3.686 0.000228 ***
omega  3.239e-06   1.449e-06    2.235 0.025394 *
alpha1 5.036e-02   1.784e-02    2.823 0.004755 **
alpha2 1.000e-08   2.877e-02    0.000 1.000000
alpha3 1.000e-08   2.506e-02    0.000 1.000000
beta1  9.495e-01   8.669e-03  109.533  < 2e-16 ***
shape  4.403e+00   3.010e-01   14.628  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Log Likelihood:
 9075.523    normalized:  2.227122

Description:
 Tue May  9 01:49:52 2023 by user:


Standardised Residuals Tests:
```

```
                         Statistic p-Value
 Jarque-Bera Test   R    Chi^2  49398.2   0
 Shapiro-Wilk Test  R    W      0.9196567 0
 Ljung-Box Test     R    Q(10)  12.48872  0.2536783
 Ljung-Box Test     R    Q(15)  17.73646  0.276774
 Ljung-Box Test     R    Q(20)  23.17403  0.2803302
 Ljung-Box Test     R^2  Q(10)  1.815427  0.997569
 Ljung-Box Test     R^2  Q(15)  2.201587  0.9999442
 Ljung-Box Test     R^2  Q(20)  3.173187  0.9999934
 LM Arch Test       R    TR^2   1.85159   0.9996018

Information Criterion Statistics:
      AIC       BIC       SIC      HQIC
-4.450809 -4.439965 -4.450815 -4.446969

> pm1=predict(m1,10)
> pm1
   meanForecast  meanError standardDeviation
1   0.001210642 0.03628920        0.03628920
2   0.001210642 0.03633140        0.03633140
3   0.001210642 0.03637354        0.03637354
4   0.001210642 0.03641562        0.03641562
5   0.001210642 0.03645765        0.03645765
6   0.001210642 0.03649963        0.03649963
7   0.001210642 0.03654155        0.03654155
8   0.001210642 0.03658342        0.03658342
9   0.001210642 0.03662524        0.03662524
10  0.001210642 0.03666700        0.03666700


Risk Measures for selected probabilities:
       prob        VaR         ES
[1,] 0.9500 0.06089100 0.07605218
[2,] 0.9900 0.08561765 0.09791273
[3,] 0.9990 0.11333366 0.12337888
[4,] 0.9999 0.13614783 0.14483627
```

To give an example, if we are making an investment of $10,000 the VaR for 95%, 99% and 99.9%, After 1 day, the returns would be:

95% VAR : $10000 * 0.06089100 = $ 608.91
99% VAR : $10000 * 0.08561765 = $ 856.1765
99.9% VAR: $10000 * 0.1133336 = $ 1133.336
99.99% VAR: $10000 * 0.136147 = $ 1361.47

```
Risk Measures for selected probabilities:
       prob       VaR        ES
```

[1,] 0.9500 0.2018430 0.2500453
[2,] 0.9900 0.2804571 0.3195472
[3,] 0.9990 0.3685754 0.4005125
[4,] 0.9999 0.4411091 0.4687325

To give an example, if we are making an investment of $10,000 the VaR for 95%, 99% and 99.9%, After 10 days, the returns will be:

95% VAR : $10000 * 0.2018430 = **$** 2018.43
99% VAR : $10000 * 0.2804571 = **$** 2804.571
99.9% VAR: $10000 * 0.368575 = **$** 3685.75
99.99% VAR: $10000 * 0.44110 = **$** 4411

Quantile regression:

Quantile regression models the relationship between a set of predictor (independent) variables and specific percentiles (or "quantiles") of a target (dependent) variable, most often the median.

Call: rq(formula = nnvda ~ vol + GSPC, tau = 0.95, data = df)

tau: [1] 0.95
Coefficients:

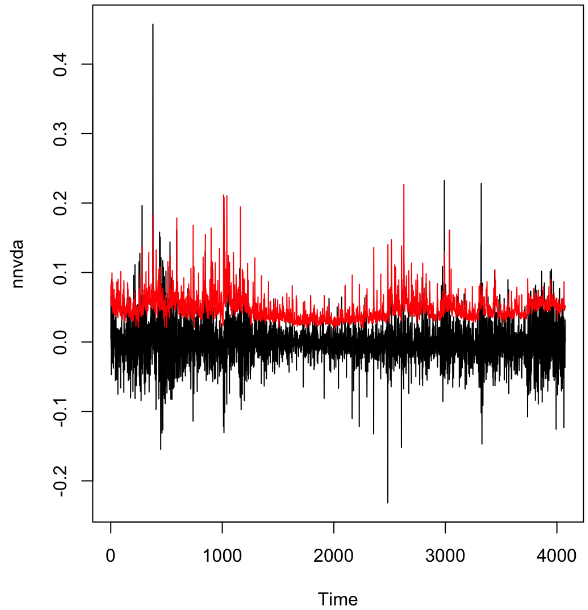
```
            Value   Std. Error t value Pr(>|t|)
(Intercept) 0.00961 0.00525    1.83009 0.06731
vol         0.00000 0.00000    7.20539 0.00000
```

VaR_quant <- 0.00961

VaR : $10000 * 0.00961 = $ 96.1

Comparison table for 95% VaR and 99& VaR for case 1,2 and 3:

Model	VaR 95%	VaR 99%
1	608.2706	860.2887
2	2018.43	2804.571
3	608.91	856.1765



Conclusion

At the end, for our Nvidia analysis, we ended up using a ARMA(8,0) for our mean model and a GARCH(3,1) for out volatility model. We faced challenges pertaining stationarity and order determination because of volatile nature of the Nvidia time series but, if extended for a long term investment horizon these models can help model the fluctuating mean and volatility of series.