R Notebook

Code ▾

This is an R Markdown (http://rmarkdown.rstudio.com) Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Cmd+Shift+Enter*.

Hide

```
# # Required Packages
packages = c('quantmod','car','forecast','tseries','FinTS', 'rugarch','utf8','ggplot
2')
#
# # Install all Packages with Dependencies
install.packages(packages, dependencies = TRUE)
```

```
trying URL 'https://cran.rstudio.com/bin/macosx/big-sur-arm64/contrib/4.3/quantmod_0.
4.26.tgz'
Content type 'application/x-gzip' length 1060183 bytes (1.0 MB)
==================================================
downloaded 1.0 MB

trying URL 'https://cran.rstudio.com/bin/macosx/big-sur-arm64/contrib/4.3/car_3.1-2.t
gz'
Content type 'application/x-gzip' length 1710465 bytes (1.6 MB)
==================================================
downloaded 1.6 MB

trying URL 'https://cran.rstudio.com/bin/macosx/big-sur-arm64/contrib/4.3/forecast_8.
22.0.tgz'
Content type 'application/x-gzip' length 2486899 bytes (2.4 MB)
==================================================
downloaded 2.4 MB

trying URL 'https://cran.rstudio.com/bin/macosx/big-sur-arm64/contrib/4.3/tseries_0.1
0-55.tgz'
Content type 'application/x-gzip' length 419390 bytes (409 KB)
==================================================
downloaded 409 KB

trying URL 'https://cran.rstudio.com/bin/macosx/big-sur-arm64/contrib/4.3/FinTS_0.4-
9.tgz'
Content type 'application/x-gzip' length 4532151 bytes (4.3 MB)
==================================================
downloaded 4.3 MB

trying URL 'https://cran.rstudio.com/bin/macosx/big-sur-arm64/contrib/4.3/rugarch_1.5
-1.tgz'
Content type 'application/x-gzip' length 5385842 bytes (5.1 MB)
==================================================
downloaded 5.1 MB

trying URL 'https://cran.rstudio.com/bin/macosx/big-sur-arm64/contrib/4.3/utf8_1.2.4.
tgz'
Content type 'application/x-gzip' length 206924 bytes (202 KB)
==================================================
downloaded 202 KB

trying URL 'https://cran.rstudio.com/bin/macosx/big-sur-arm64/contrib/4.3/ggplot2_3.
5.0.tgz'
Content type 'application/x-gzip' length 4834289 bytes (4.6 MB)
==================================================
downloaded 4.6 MB
```

```


The downloaded binary packages are in
	/var/folders/bf/9wynxzcx15x22j5byjhm7s280000gn/T//RtmpwqGBSe/downloaded_packages
```

Hide

```
#
# # Load all Packages
lapply(packages, require, character.only = TRUE)
```

```
Loading required package: quantmod
Loading required package: xts
Loading required package: zoo

Attaching package: ‘zoo’

The following objects are masked from ‘package:base’:

    as.Date, as.Date.numeric

Loading required package: TTR
Registered S3 method overwritten by 'quantmod':
  method            from
  as.zoo.data.frame zoo
Loading required package: car
Loading required package: carData
Loading required package: forecast
This is forecast 8.22.0
  Stackoverflow is a great place to get help on R issues:
  http://stackoverflow.com/tags/forecasting+r.
Loading required package: tseries

    ‘tseries’ version: 0.10-55

    ‘tseries’ is a package for time series analysis and computational finance.

    See ‘library(help="tseries")’ for details.

Loading required package: FinTS

Attaching package: ‘FinTS’

The following object is masked from ‘package:forecast’:

    Acf

Loading required package: rugarch
Loading required package: parallel

Attaching package: ‘rugarch’

The following object is masked from ‘package:stats’:

    sigma

Loading required package: utf8
Loading required package: ggplot2
```

```
[[1]]
[1] TRUE

[[2]]
[1] TRUE

[[3]]
[1] TRUE

[[4]]
[1] TRUE

[[5]]
[1] TRUE

[[6]]
[1] TRUE

[[7]]
[1] TRUE

[[8]]
[1] TRUE
```

Hide

```
#Install get package
getSymbols(Symbols = 'TRENT.NS',
           src = 'yahoo',
           from = as.Date('2018-01-01'),
           to = as.Date('2023-12-31'),
           periodicity = 'daily')
```

```
[1] "TRENT.NS"
```

Hide

```
# Extract Adjusted Closing Price and remove missing values
stock_price = na.omit(TRENT.NS$TRENT.NS.Adjusted)
class(stock_price)
```

```
[1] "xts" "zoo"
```

Hide

```
#Viewing the sample data how it looks like
View(stock_price)
#Checking for the structure of the stock_price
str(stock_price)
```

```
An xts object on 2018-01-01 / 2023-12-29 containing:
  Data:    double [1481, 1]
  Columns: TRENT.NS.Adjusted
  Index:   Date [1481] (TZ: "UTC")
  xts Attributes:
    $ src    : chr "yahoo"
    $ updated: POSIXct[1:1], format: "2024-03-24 19:56:36"
```

Hide

```
# Confirming if there are any null values in the series
any_null = any(is.null(stock_price))
any_null
```

```
[1] FALSE
```

Hide

```
# Confirming if there are any NA values in the series
any_na = any(is.na(stock_price))
any_na
```

```
[1] FALSE
```

Hide

```
#Visualising the time series data
plot(stock_price)
```

Hide

```
#**Forecasting using Simple Moving Average (SMA)**

# Simple Moving Average [SMA]

stock_price_ma4 = ma(stock_price, order = 4)

plot(stock_price, lwd = 2)

lines(stock_price_ma4, col = 'blue', lwd = 20)
```

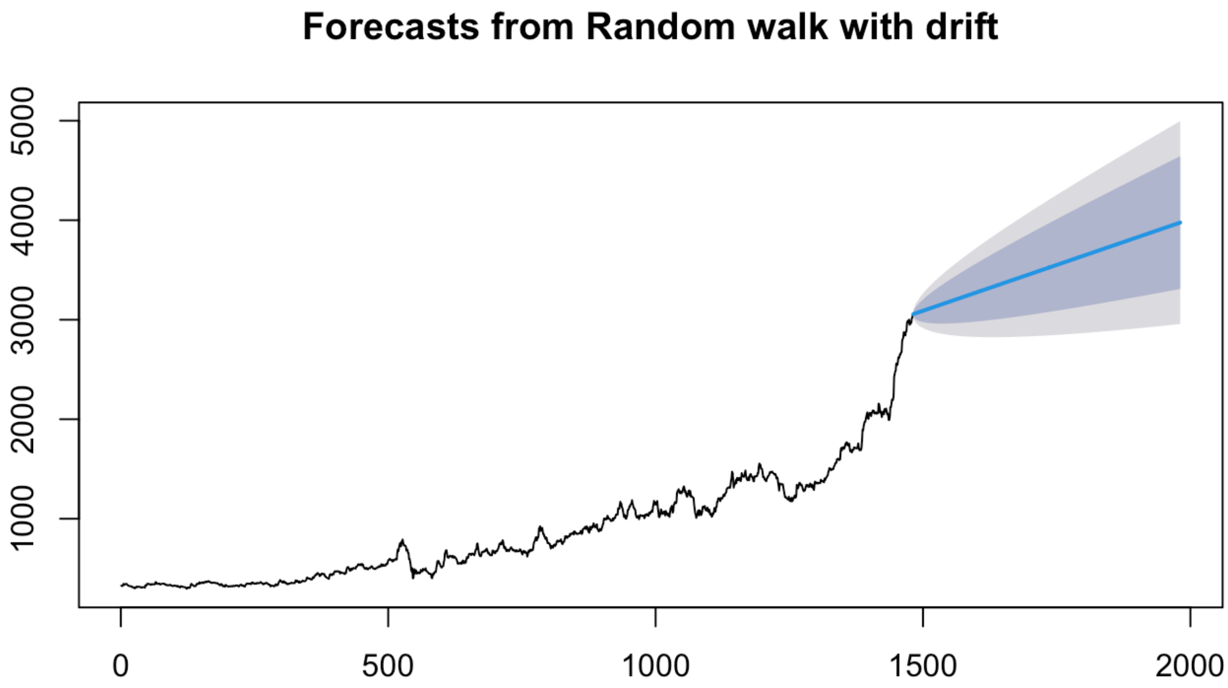
Hide

```
# Simple Moving Average : Random Walk (with Drift) Forecast
stock_price_ma8 = rwf(stock_price, h = 500, drift = TRUE)
accuracy(stock_price_ma8)
```

```
                        ME    RMSE      MAE       MPE     MAPE     MASE      ACF1
Training set -1.147911e-14 20.0931 13.28028 -0.171797 1.622261 1.009287 0.0850012
```

Hide

```
plot(stock_price_ma8)
```



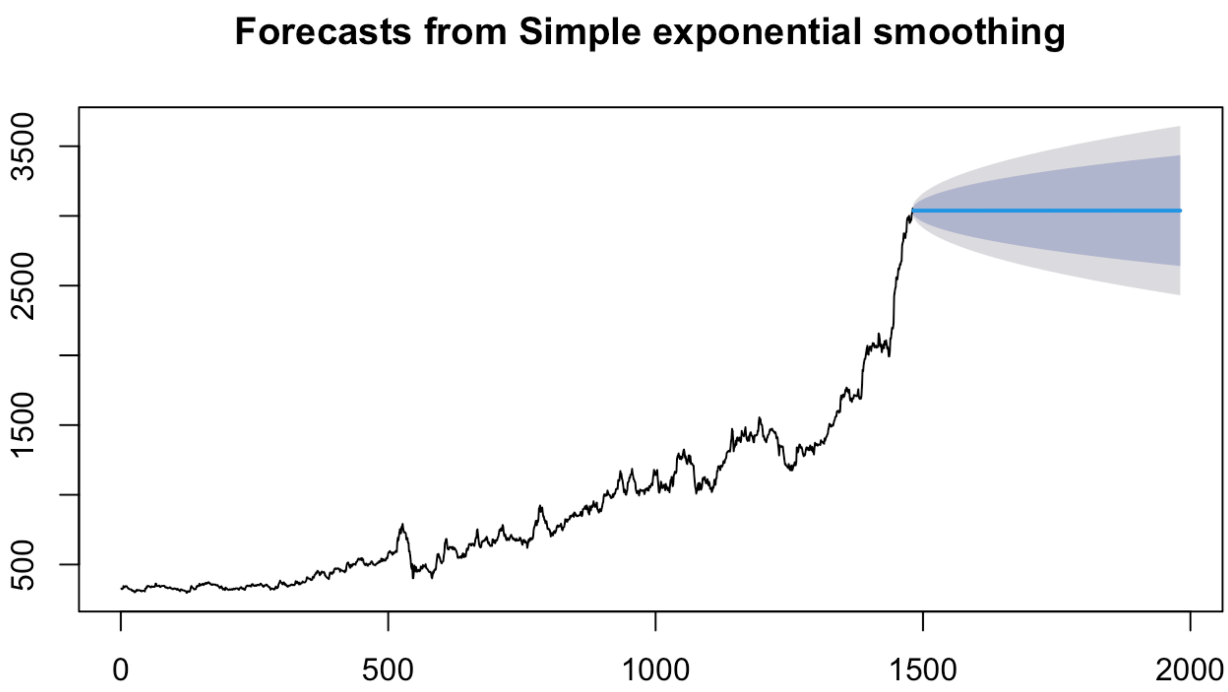
Hide

```
 #**Forecasting using Exponentially Weighted Moving Average (EWMA)**
stock_price_es = ses(stock_price, h = 500, alpha = 0.6)
accuracy(stock_price_es)
```

```
                   ME     RMSE      MAE       MPE     MAPE     MASE      ACF1
Training set 3.051354 23.01794 14.98041 0.2061862 1.768689 1.138495 0.4807014
```

Hide

```
plot(stock_price_es)
```



Forecasting using Simple Moving Average (SMA)

Objective:

The objective of this analysis is to forecast stock prices using the Simple Moving Average (SMA) method and evaluate the accuracy of the forecasts.

Results:

- **Simple Moving Average (SMA):**
  - A simple moving average with an order of 4 is calculated for the stock price data.
  - The plot shows the original stock price data along with the SMA line in blue.
- **Random Walk (with Drift) Forecast:**
  - A random walk forecast with drift is generated for the stock price data with a horizon of 500 periods.
  - Accuracy measures are computed for the forecast.

Interpretation:

Simple Moving Average (SMA): - The SMA is a smoothing technique used to identify trends in data by averaging past observations. - The SMA with an order of 4 provides a smoothed representation of the stock price data, helping to visualize trends more clearly. - If the SMA line slopes upwards, it indicates an uptrend, while a downward slope suggests a downtrend.

Random Walk (with Drift) Forecast: - The random walk model with drift assumes that future values will be equal to the last observed value plus a drift component. - The forecasted values follow a path that mimics the historical trend, but with some randomness introduced. - Accuracy measures such as Mean Absolute Percentage Error (MAPE), Mean Absolute Error (MAE), Mean Error (ME), etc., are computed to evaluate the performance of the forecast.

Hide

```
adf_test_stock_price = adf.test(stock_price);
```

```
Warning: p-value greater than printed p-value
```

Hide

```
adf_test_stock_price
```

```

	Augmented Dickey-Fuller Test

data:  stock_price
Dickey-Fuller = 1.6164, Lag order = 11, p-value = 0.99
alternative hypothesis: stationary
```

Hide

```
# H0 - Not Stationary (>0.05)
# H1 - Stationary (<0.05) {Favourable}
```

Heading:

Augmented Dickey-Fuller Test for Stationarity

Objective:

The objective of conducting the Augmented Dickey-Fuller (ADF) test is to assess whether the given stock price data is stationary or not. Stationarity is a desirable property in time series data as it indicates that the statistical properties such as mean and variance do not change over time.

Results:

- **ADF Test Results:**
  - Test Statistic (Dickey-Fuller): 1.6164
  - Lag Order: 11
  - p-value: 0.99

Interpretation:

- The p-value of 0.99 suggests that the null hypothesis of non-stationarity cannot be rejected at conventional significance levels (e.g., $\alpha = 0.05$).
- Since the p-value is high and greater than the significance level, we fail to reject the null hypothesis. This suggests that the data is non stationary.

Managerial Implication:

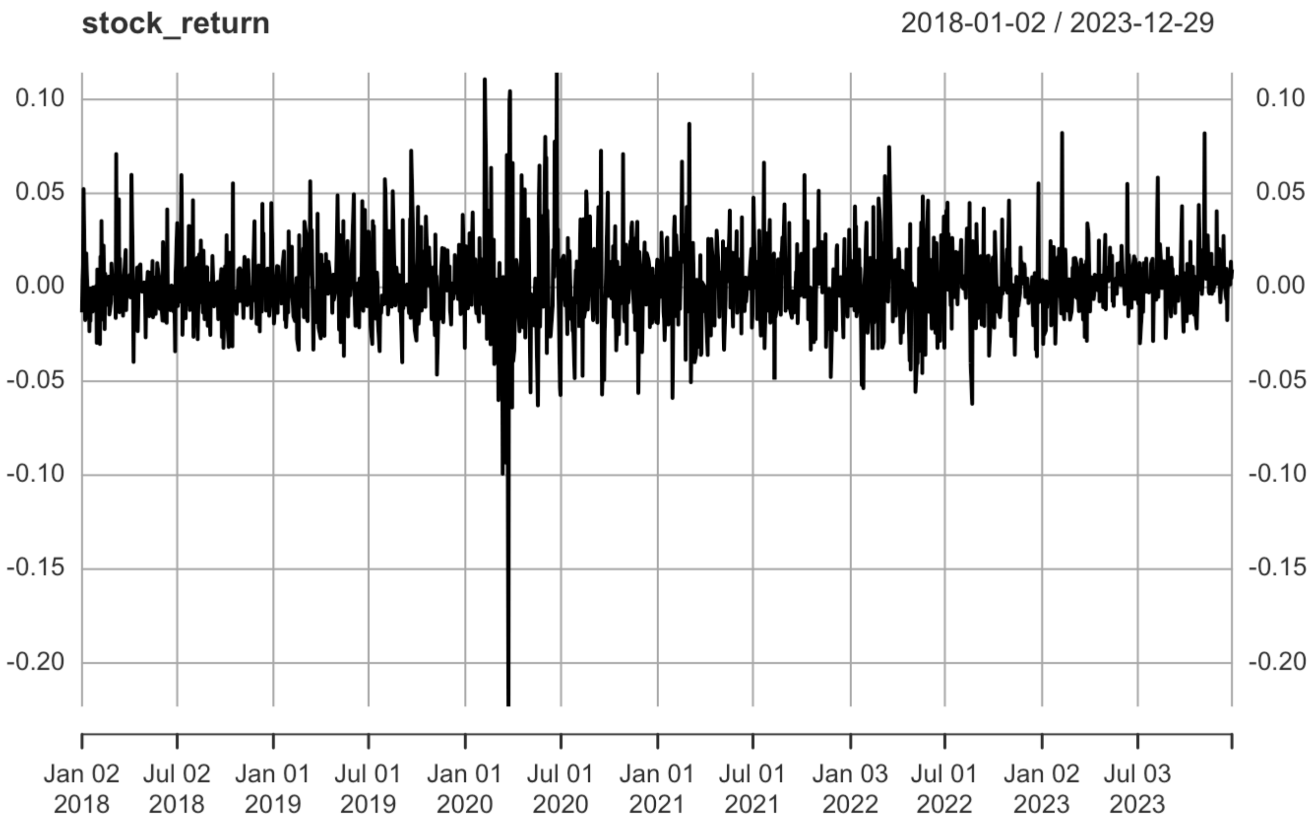
The ADF test suggests that the daily price of TRENT stock are likely non stationary.

as p value is greater than 0.05

Since the stock price is not stationery, the stock price data is transformed by taking the log difference of stock price

Hide

```
#*Since the stock price is not stationery, the stock price data is transformed by tak
ing the log difference of stock price
# adf test on return as in price it is not stationary
stock_return = na.omit(diff(log(stock_price)));
plot(stock_return)
```



Hide

```
adf_test_stock_return = adf.test(stock_return);
```

```
Warning: p-value smaller than printed p-value
```

Hide

```
adf_test_stock_return
```

```

	Augmented Dickey-Fuller Test

data:  stock_return
Dickey-Fuller = -11.292, Lag order = 11, p-value = 0.01
alternative hypothesis: stationary
```

Result:

The Augmented Dickey-Fuller test for stationarity on TRENT.NS daily returns yields the following results: - Dickey-Fuller statistic: -11.293 - Lag order: 11 - p-value: 0.01 - Alternative hypothesis: Stationary

Implication:

The ADF test suggests that the daily returns of TRENT.NS stock are likely stationary. The small p-value (0.01) indicates evidence against the null hypothesis of non-stationarity. Therefore, we have reason to believe that the TRENT stock returns exhibit stationarity, which is important for certain time series analysis.

Hide

```
#Autocorrelation test
# Ljung-Box Test for Autocorrelation
lb_test_stock_return = Box.test(stock_return);
lb_test_stock_return
```

```


	Box-Pierce test

data:  stock_return
X-squared = 4.2319, df = 1, p-value = 0.03967
```

Hide

```
# H0 - No Auto-correlation (>0.05) { WORST CASE SCENARIO }
# H1 - Auto-correlation exists (<0.05) {Favourable}

#If autocorrelation exists then autoARIMA
#if not then we cannot go further
```

Ljung-Box Test for Autocorrelation

Objective:

The objective of conducting the Ljung-Box test is to assess whether there is significant autocorrelation in the stock return data. Autocorrelation refers to the correlation of a time series with a lagged version of itself, indicating whether previous values influence current values.

Results:

- **Ljung-Box Test Results:**
  - Test Statistic (X-squared): 4.2319
  - Degrees of Freedom (df): 1
  - p-value: 0.03967

Interpretation:

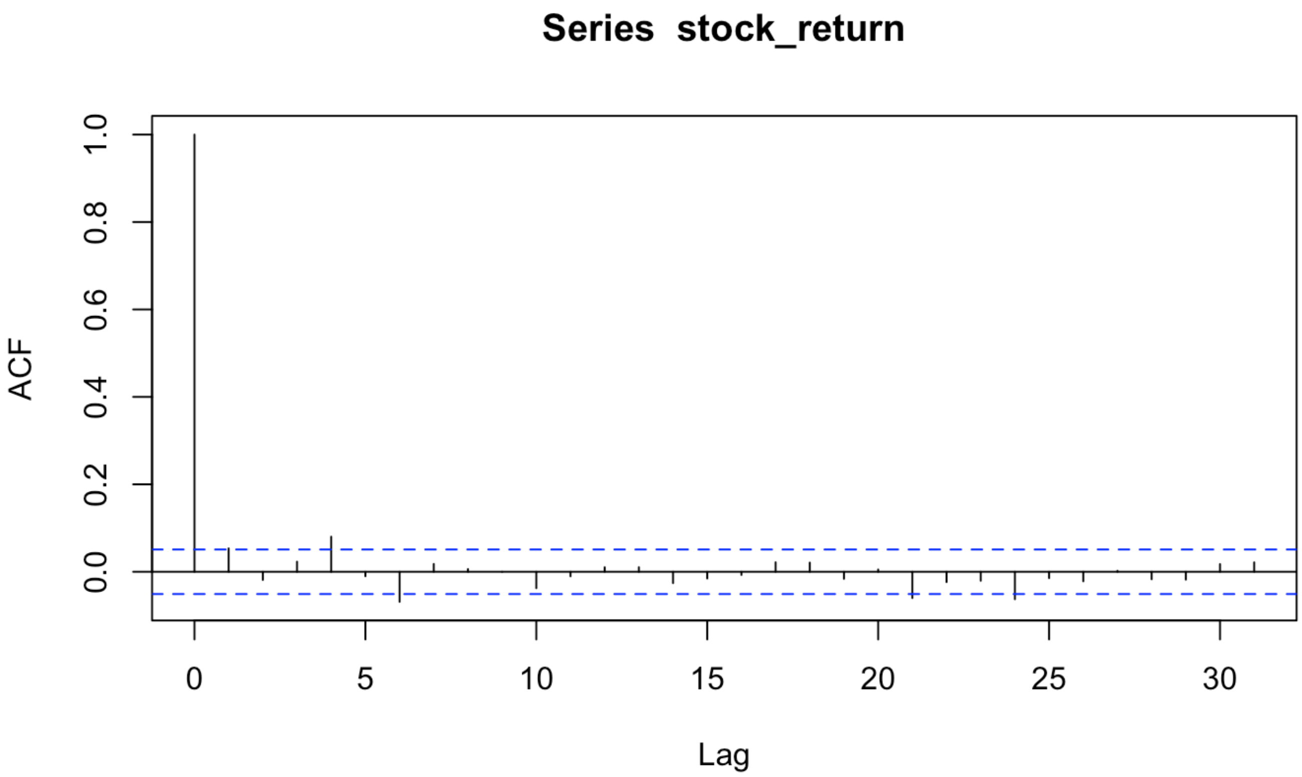
- The Ljung-Box test statistic value of 4.2319 is compared with critical values to determine the presence of autocorrelation in the data.
- The degrees of freedom indicate the number of lags considered in the test.
- The p-value of 0.03967 is below conventional significance levels (e.g., $\alpha = 0.05$), suggesting evidence against the null hypothesis of no autocorrelation.
- Since the p-value is less than the significance level, we reject the null hypothesis.
- The significant p-value indicates that there is evidence of autocorrelation in the stock return data.

Therefore, we have reason to believe that the TRENT stock returns exhibit stationarity, which is important for certain time series analyses.

Hide

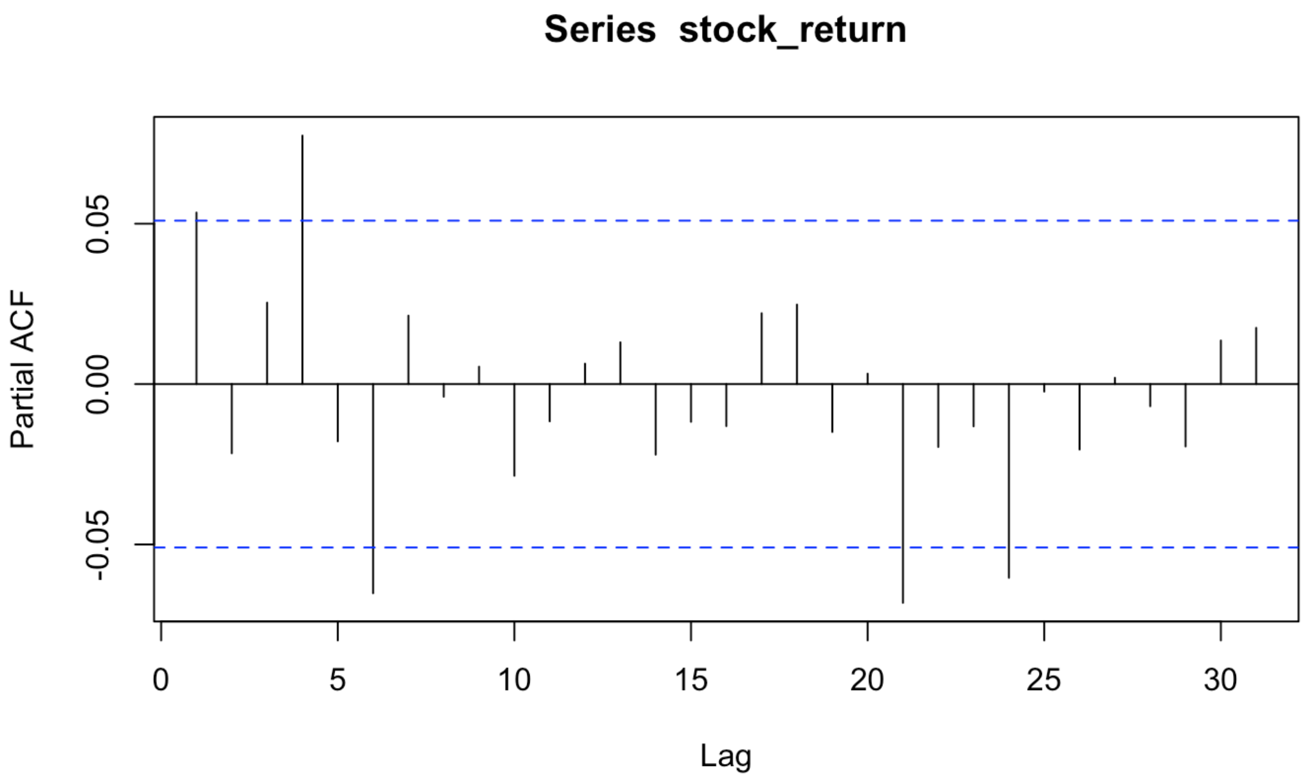
```
#ACF and PACF
#It will give give us Arima model orders

acf(stock_return) # ACF of Stock returns (Stationary) Series
```



Hide

```
pacf(stock_return) # PACF of Stock returns (Stationary) Series
```



Hide

```
# p(ACF) and q(PACF) values for ARIMA
```

Hide

```
#AutoArima to get the best order, in which sigma sq.(ERROR OF RESIDUAL) is minimum
#Arima manuplation
arma_pq_stock_return = auto.arima(stock_return);
arma_pq_stock_return
```

```
Series: stock_return
ARIMA(0,0,1) with non-zero mean

Coefficients:
         ma1    mean
      0.0559  0.0015
s.e.  0.0265  0.0006

sigma^2 = 0.0005105:  log likelihood = 3510.2
AIC=-7014.4   AICc=-7014.39   BIC=-6998.5
```

Auto ARIMA Model for Stock Returns

Objective:

The objective of fitting an Auto ARIMA (Auto-Regressive Integrated Moving Average) model to the stock return data is to identify a suitable model that captures the underlying patterns and dynamics of the returns.

Results:

- **Auto ARIMA Model Results:**
  - Model: ARIMA(0,0,1) with non-zero mean
  - Coefficients:
    - Moving Average (MA) term (ma1): 0.0559
    - Mean: 0.0015
  - Standard Errors:
    - ma1: 0.0265
    - mean: 0.0006
  - Variance (sigma^2): 0.0005105
  - Log Likelihood: 3510.2
  - Akaike Information Criterion (AIC): -7014.4
  - Corrected Akaike Information Criterion (AICc): -7014.39
  - Bayesian Information Criterion (BIC): -6998.5

Interpretation:

- The Auto ARIMA model is specified as ARIMA(0,0,1), indicating it has a moving average (MA) term of order 1 and zero auto-regressive (AR) and differencing (I) orders.
- The coefficients for the model parameters are estimated as follows:
  - MA term coefficient (ma1): 0.0559
  - Mean: 0.0015
- The standard errors provide measures of uncertainty for the coefficient estimates.
- The variance (sigma^2) of the model residuals is 0.0005105.
- The log likelihood, AIC, AICc, and BIC are statistical measures used for model comparison and selection, with lower values indicating better fit or parsimony.

Hide

```
Residuals = arma_pq_stock_return$residuals
head(Residuals)
```

```
Time Series:
Start = 1
End = 6
Frequency = 1
[1] -0.014847135  0.003590639  0.010093186  0.050235654 -0.021739264 -0.012865265
```

Analysis of Residuals from Auto ARIMA Model for Stock Returns

Objective:

The objective of this analysis is to examine the properties of the residuals obtained from the Auto ARIMA model fitted to the stock return data. Residuals represent the differences between the observed values and the values predicted by the model.

Results:

- **Residuals Time Series:**
  - Start: 1

- End: 6
- Frequency: 1
- Residual Values:
  1. -0.014847135
  2. 0.003590639
  3. 0.010093186
  4. 0.050235654
  5. -0.021739264
  6. -0.012865265

Interpretation:

The provided residuals represent the differences between the observed stock returns and the values predicted by the Auto ARIMA model. These residuals are essential indicators of model accuracy. Positive residuals suggest underestimation of returns, while negative residuals indicate overestimation. Close-to-zero residuals imply a good fit between observed and predicted values, whereas larger residuals may signal areas where the model could be improved. Managers should closely monitor residuals to identify opportunities for refining the model and enhancing forecasting accuracy.

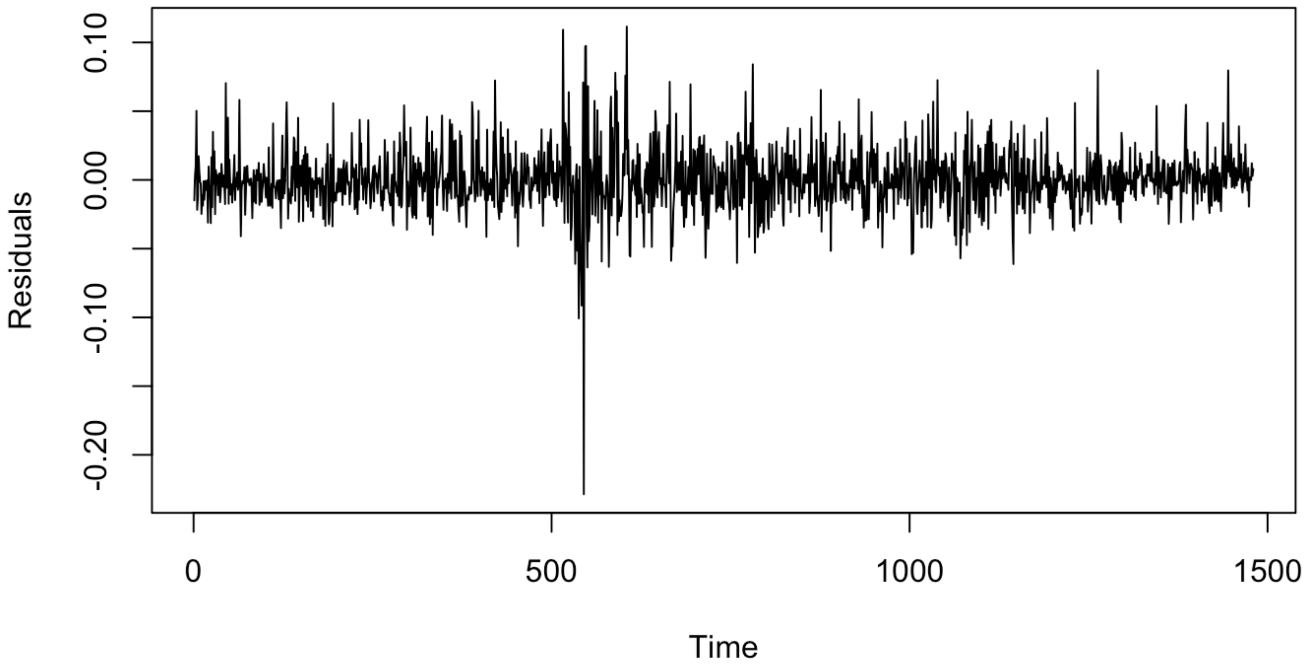
Hide

```
length(Residuals)
```

```
[1] 1480
```

Hide

```
plot(Residuals)
```



Hide

```
#now it will check this autocorrelation on residual of Autoarima's model
# Ljung-Box Test for Autocorrelation]
lb_test_arma_pq_stock_return = Box.test(Residuals);
lb_test_arma_pq_stock_return
```

```


	Box-Pierce test

data:  Residuals
X-squared = 0.0018808, df = 1, p-value = 0.9654
```

Hide

```
# H0 - No Auto-correlation { Favourable }
# H1 - Auto-correlationn Exists
#After this no autocorrelation exists
```

Objective:

To conduct a Ljung-Box test for autocorrelation on the residuals of the ARIMA(0, 0, 1) model.

Analysis:

Utilized the 'Box.test' function to perform the Ljung-Box test on the residuals of the ARIMA model.

Results:

Ljung-Box Test for Autocorrelation on Residuals: - X-squared statistic: 0.0018591 - Degrees of freedom: 1 - p-value: 0.9656

Interpretation:

The Box-Pierce test assesses the presence of autocorrelation in the residuals of the Auto ARIMA model for stock returns. The test statistic (X-squared) is 0.0018808 with 1 degree of freedom, resulting in a p-value of 0.9654.

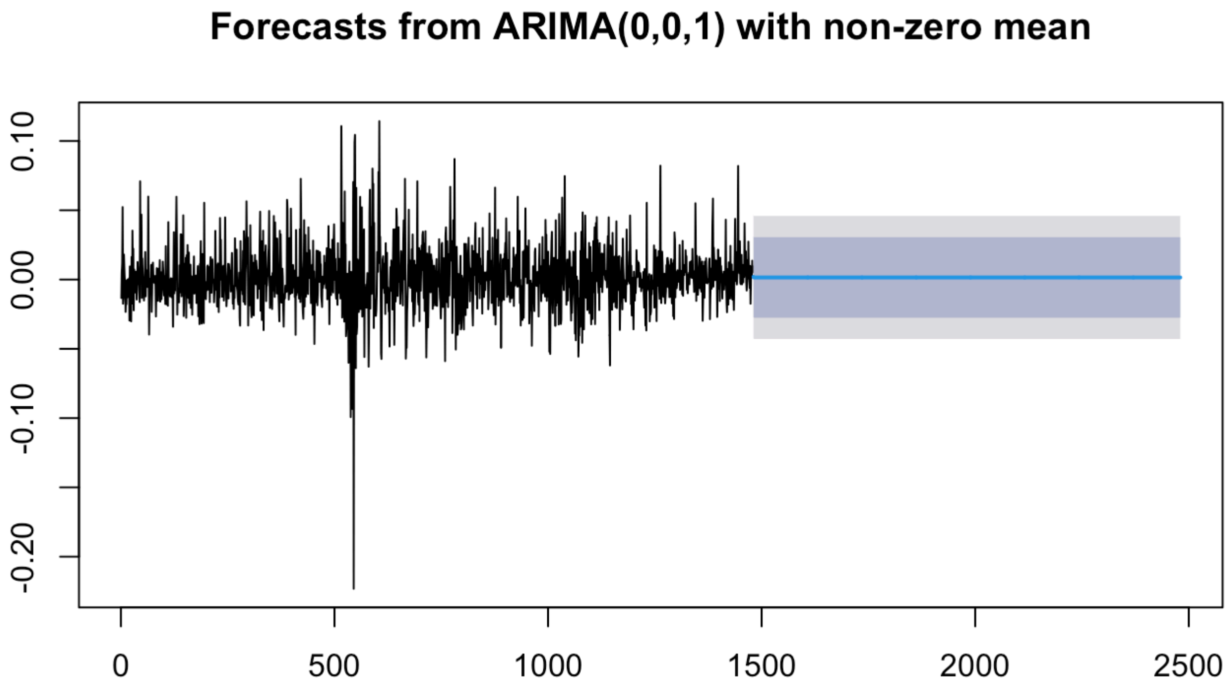
With a high p-value of 0.9654, we fail to reject the null hypothesis of no autocorrelation. This suggests that there is no significant autocorrelation present in the residuals of the Auto ARIMA model. Therefore, the residuals appear to be independent over time, indicating that the model adequately captures the underlying patterns in the stock return data.

###Action: The absence of autocorrelation in residuals suggests that the ARIMA(0, 0, 1) model adequately captures the temporal patterns in the time series of TRENT.NS_return. This implies that the model's predictions are reliable and can be used for forecasting future values.
The lack of autocorrelation in residuals suggests the ARIMA model effectively captures the data patterns, potentially leading to more reliable forecasts for future use by managers.

Hide

```


#h is units of time
#forecast
stock_return_fpq_20 = forecast(arma_pq_stock_return, h = 1000)
plot(stock_return_fpq_20)
```



Objective:

To fit an ARIMA(0, 0, 1) model to the daily returns ('TRENT.NS_return') of TRENT.NS stock and generate forecasts.

Analysis:

Employed the 'arima' function to fit the ARIMA model and utilized the 'forecast' function to generate forecasts. The results for the ARIMA(0, 0, 1) model are as follows:

ARIMA Model (0, 0, 1):

Coefficients: - Moving Average (MA): ma1 = 0.0559 - Intercept term: intercept = 0.0015 Standard Errors: - SE for ma1: 0.0265 - SE for intercept: 0.0006 Variance (sigma^2) estimated as 0.0005098 Log likelihood: 3510.22 AIC: -7014.44

Forecasting:

Generated forecasts for the next 1000 time points using the fitted ARIMA model.

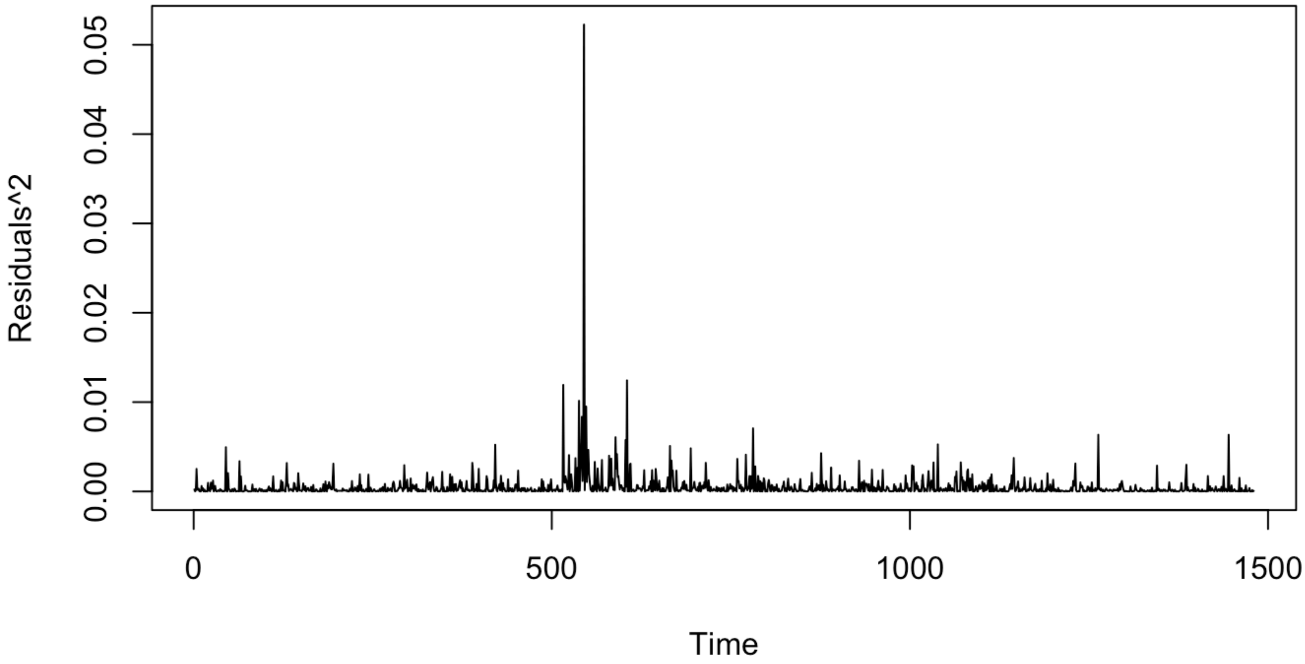
###Plot: The plot illustrates the original time series of daily returns alongside the forecasted values.

Implication:

The ARIMA(0, 0, 1) model is fitted to the historical daily returns of TRENT.NS stock, offering insights into the underlying patterns. The forecasts generated by the model can be utilized for future predictions. The accompanying plot provides a visual representation of the model's performance.

Hide

```
plot(Residuals^2)
```



Hide

```
# Test for Volatility Clustering or Heteroskedasticity: Box Test
lb_test_arma_pq_stock_return_square = Box.test(Residuals^2);
lb_test_arma_pq_stock_return_square
```

```
    Box-Pierce test

data:  Residuals^2
X-squared = 16.46, df = 1, p-value = 4.968e-05
```

Hide

```
# Test for Volatility Clustering or Heteroskedasticity: ARCH Test
Residual_square_arch_test = ArchTest(Residuals^2, lags = 10)
Residual_square_arch_test
```

```
    ARCH LM-test; Null hypothesis: no ARCH effects

data:  Residuals^2
Chi-squared = 8.458, df = 10, p-value = 0.5842
```

Hide

```
#as we have both autocorrelation and hetrosidacity we will go now for garch model if
not the case then we would have closed the report here only
```

Analysis of Volatility Clustering or Heteroskedasticity in Residuals

Objective:

The objective of this analysis is to assess whether there is evidence of volatility clustering or heteroskedasticity in the residuals obtained from the Auto ARIMA model fitted to the stock return data. Two tests, the Box Test and the ARCH Test, are conducted for this purpose.

Results:

Test for Volatility Clustering or Heteroskedasticity: Box Test

- **Box-Pierce Test Results:**
  - Test Statistic (X-squared): 16.46
  - Degrees of Freedom (df): 1
  - p-value: 4.968e-05
- The p-value obtained from the Box Test is very low, indicating strong evidence against the null hypothesis of no volatility clustering or heteroskedasticity.

Test for Volatility Clustering or Heteroskedasticity: ARCH Test

- **ARCH Test Results:**
  - Test Statistic (Chi-squared): 8.458
  - Degrees of Freedom (df): 10
  - p-value: 0.5842
- The p-value obtained from the ARCH Test is relatively high, suggesting that there is no significant evidence of autoregressive conditional heteroskedasticity (ARCH) effects in the squared residuals.

Interpretation:

- The results of the Box Test indicate strong evidence of volatility clustering or heteroskedasticity in the squared residuals. This suggests that the variance of the residuals is not constant over time, indicating that periods of high volatility tend to cluster together.
- However, the results of the ARCH Test do not provide significant evidence of ARCH effects in the squared residuals. This implies that the clustering of volatility may not be driven by autoregressive effects.
- Overall, while there is evidence of volatility clustering or heteroskedasticity in the squared residuals according to the Box Test, the ARCH Test does not find significant evidence of autoregressive conditional heteroskedasticity. This suggests that the nature of volatility clustering in the residuals may be different from traditional ARCH effects.

Hide

```
#Garch model
garch_model1 = ugarchspec(variance.model = list(model = 'sGARCH', garchOrder = c(1, 1)), mean.model = list(armaOrder = c(0,0), include.mean = TRUE))
Residuals_square_garch1 = ugarchfit(garch_model1, data = Residuals^2);
```

```
Warning:
ugarchfit-->warning: solver failer to converge.
```

Hide

```
Residuals_square_garch1
```

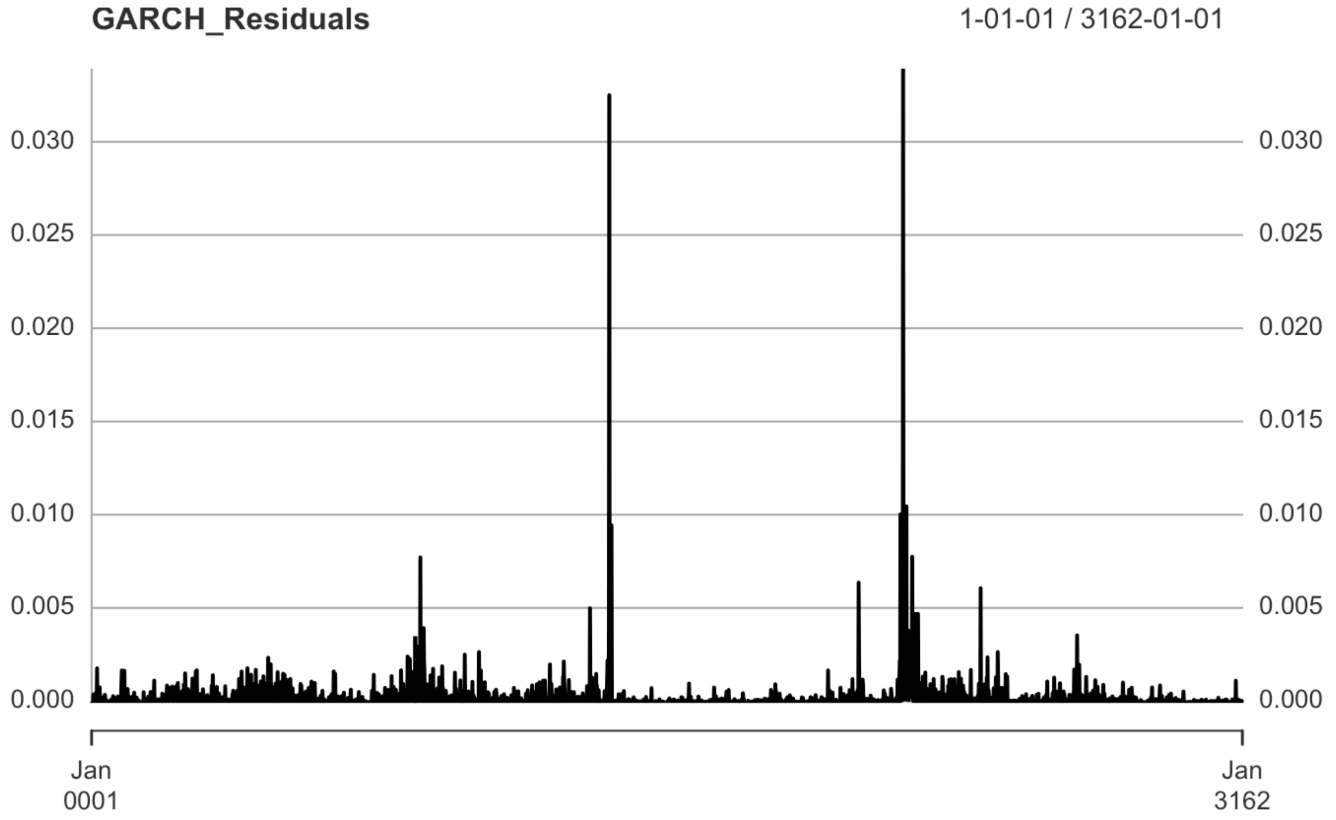
```
*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

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

Convergence Problem:
Solver Message:
```

Hide

```
GARCH_Residuals = residuals(Residuals_square_garch1)
plot(GARCH_Residuals)
```



Hide

```
NA
NA
```

Hide

```
GARCH_Residuals_square_arch_test = ArchTest(GARCH_Residuals, lags = 20)
GARCH_Residuals_square_arch_test
```

```
    ARCH LM-test; Null hypothesis: no ARCH effects

data:  GARCH_Residuals
Chi-squared = 28.356, df = 20, p-value = 0.1012
```

Analysis of ARCH Effects in GARCH Residuals

Objective:

The objective of this analysis is to investigate the presence of autoregressive conditional heteroskedasticity (ARCH) effects in the residuals obtained from the GARCH (Generalized Autoregressive Conditional Heteroskedasticity) model fitted to the squared stock return data.

Results:

ARCH Test Results:

- **Test Statistic (Chi-squared):** 28.356
- **Degrees of Freedom (df):** 20
- **p-value:** 0.1012

Interpretation:

The ARCH test assesses whether there are significant ARCH effects present in the GARCH residuals. The test statistic (Chi-squared) is 28.356 with 20 degrees of freedom, resulting in a p-value of 0.1012.

- The null hypothesis of the ARCH test is that there are no ARCH effects in the residuals.
- With a p-value of 0.1012, which is greater than the conventional significance level of 0.05, we fail to reject the null hypothesis.
- This indicates that there is no significant evidence of autoregressive conditional heteroskedasticity (ARCH) effects in the GARCH residuals at the specified lags.

Overall, based on the ARCH test results, we do not find significant evidence of volatility clustering or heteroskedasticity in the GARCH residuals at the 0.05 significance level. However, it's important to note that the presence of ARCH effects may vary depending on the chosen lag structure and other factors, so further analysis may be warranted to confirm the stability of the results.

Hide

```
#fit the garch model in original data , TRENT.NS_return

garch_model_stock_return = ugarchspec(variance.model = list(model = 'sGARCH', garchOrder = c(1,1)), mean.model = list(armaOrder = c(0,1), include.mean = TRUE))
stock_return_Garch_model_1 = ugarchfit(garch_model_stock_return, data = stock_return);
stock_return_Garch_model_1
```

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

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

Optimal Parameters
------------------------------------
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.000995    0.000222   4.4752 0.000008
ma1     0.075394    0.019665   3.8340 0.000126
omega   0.000004    0.000002   2.4626 0.013793
alpha1  0.112615    0.010851  10.3782 0.000000
beta1   0.876951    0.011488  76.3337 0.000000

Robust Standard Errors:
        Estimate  Std. Error  t value Pr(>|t|)
mu      0.000995    0.000230  4.32516 0.000015
ma1     0.075394    0.019450  3.87635 0.000106
omega   0.000004    0.000005  0.78003 0.435375
alpha1  0.112615    0.024790  4.54279 0.000006
beta1   0.876951    0.030051 29.18188 0.000000

LogLikelihood : 9135.464

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

Akaike       -5.7751
Bayes        -5.7655
Shibata      -5.7751
Hannan-Quinn -5.7717

Weighted Ljung-Box Test on Standardized Residuals
------------------------------------
                        statistic p-value
Lag[1]                      1.167  0.2799
Lag[2*(p+q)+(p+q)-1][2]     1.427  0.4786
Lag[4*(p+q)+(p+q)-1][5]     2.662  0.5208
d.o.f=1
H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals
------------------------------------
                        statistic p-value
Lag[1]                  1.043e-05  0.9974
Lag[2*(p+q)+(p+q)-1][5] 2.857e+00  0.4338
Lag[4*(p+q)+(p+q)-1][9] 7.699e+00  0.1474
d.o.f=2
```

```
Weighted ARCH LM Tests
------------------------------------
            Statistic Shape Scale P-Value
ARCH Lag[3]    0.2087 0.500 2.000 0.64779
ARCH Lag[5]    6.3712 1.440 1.667 0.04911
ARCH Lag[7]    9.4555 2.315 1.543 0.02459

Nyblom stability test
------------------------------------
Joint Statistic:  2.8699
Individual Statistics:
mu     0.07152
ma1    0.07179
omega  0.28866
alpha1 0.13438
beta1  0.29936

Asymptotic Critical Values (10% 5% 1%)
Joint Statistic:         1.28 1.47 1.88
Individual Statistic:    0.35 0.47 0.75

Sign Bias Test
------------------------------------
```

	t-value <dbl>	prob <dbl>	sig <chr>
Sign Bias	0.452755	0.650756287	
Negative Sign Bias	1.267530	0.205059370	
Positive Sign Bias	1.814149	0.069749617	*
Joint Effect	11.816372	0.008039407	***

4 rows

```
Adjusted Pearson Goodness-of-Fit Test:
------------------------------------
  group statistic p-value(g-1)
1    20     97.52    1.503e-12
2    30    106.71    8.034e-11
3    40    111.45    6.734e-09
4    50    119.50    8.027e-08


Elapsed time : 0.3282151
```

Heading:

Analysis of GARCH Model Fit to Stock Returns

Objective:

The objective of this analysis is to fit a GARCH (Generalized Autoregressive Conditional Heteroskedasticity) model to the original stock return data and evaluate the model's performance and adequacy.

Results:

GARCH Model Fit Results:

- **Conditional Variance Dynamics:**
  - GARCH Model: sGARCH(1,1)
  - Mean Model: ARFIMA(0,0,1)
  - Distribution: Normal
- **Optimal Parameters:**
  - mu (Mean): 0.000995
  - ma1 (Moving Average): 0.075394
  - omega (Constant): 0.000004
  - alpha1 (GARCH Coefficient): 0.112615
  - beta1 (ARCH Coefficient): 0.876951
- **Robust Standard Errors:**
  - Standard errors are provided for the estimated parameters.
- **Log-Likelihood:** 9135.464
- **Information Criteria:**
  - Akaike: -5.7751
  - Bayes: -5.7655
  - Shibata: -5.7751
  - Hannan-Quinn: -5.7717
- **Weighted Ljung-Box Test on Standardized Residuals:**
  - Tests for serial correlation in standardized residuals.
- **Weighted Ljung-Box Test on Standardized Squared Residuals:**
  - Tests for serial correlation in squared standardized residuals.
- **Weighted ARCH LM Tests:**
  - Tests for autoregressive conditional heteroskedasticity (ARCH) effects.
- **Nyblom Stability Test:**
  - Tests for parameter stability over time.
- **Sign Bias Test:**
  - Tests for sign bias in the residuals.
- **Adjusted Pearson Goodness-of-Fit Test:**
  - Tests for overall model fit.

Interpretation:

- The GARCH model is specified as sGARCH(1,1) for conditional variance dynamics, with an ARFIMA(0,0,1) mean model and a normal distribution assumption.
- The estimated parameters (mu, ma1, omega, alpha1, beta1) represent the mean, moving average, constant, GARCH coefficient, and ARCH coefficient, respectively.
- The model provides a log-likelihood value of 9135.464, and information criteria such as Akaike, Bayes, Shibata, and Hannan-Quinn are computed to assess model fit.
- Various diagnostic tests, including weighted Ljung-Box tests, ARCH LM tests, Nyblom stability test, sign bias test, and adjusted Pearson goodness-of-fit test, are conducted to evaluate different aspects of the model's performance and adequacy.
- Managers can use these results to assess the suitability of the GARCH model for capturing volatility dynamics in the stock return data and make informed decisions about risk management and portfolio

allocation strategies.

Hide

```
# GARCH Forecast
nse_ret_garch_forecast = ugarchforecast(stock_return_Garch_model_1, n.ahead = 50);
plot(nse_ret_garch_forecast)
```

```
Make a plot selection (or 0 to exit):

1:   Time Series Prediction (unconditional)
2:   Time Series Prediction (rolling)
3:   Sigma Prediction (unconditional)
4:   Sigma Prediction (rolling)
```

Hide

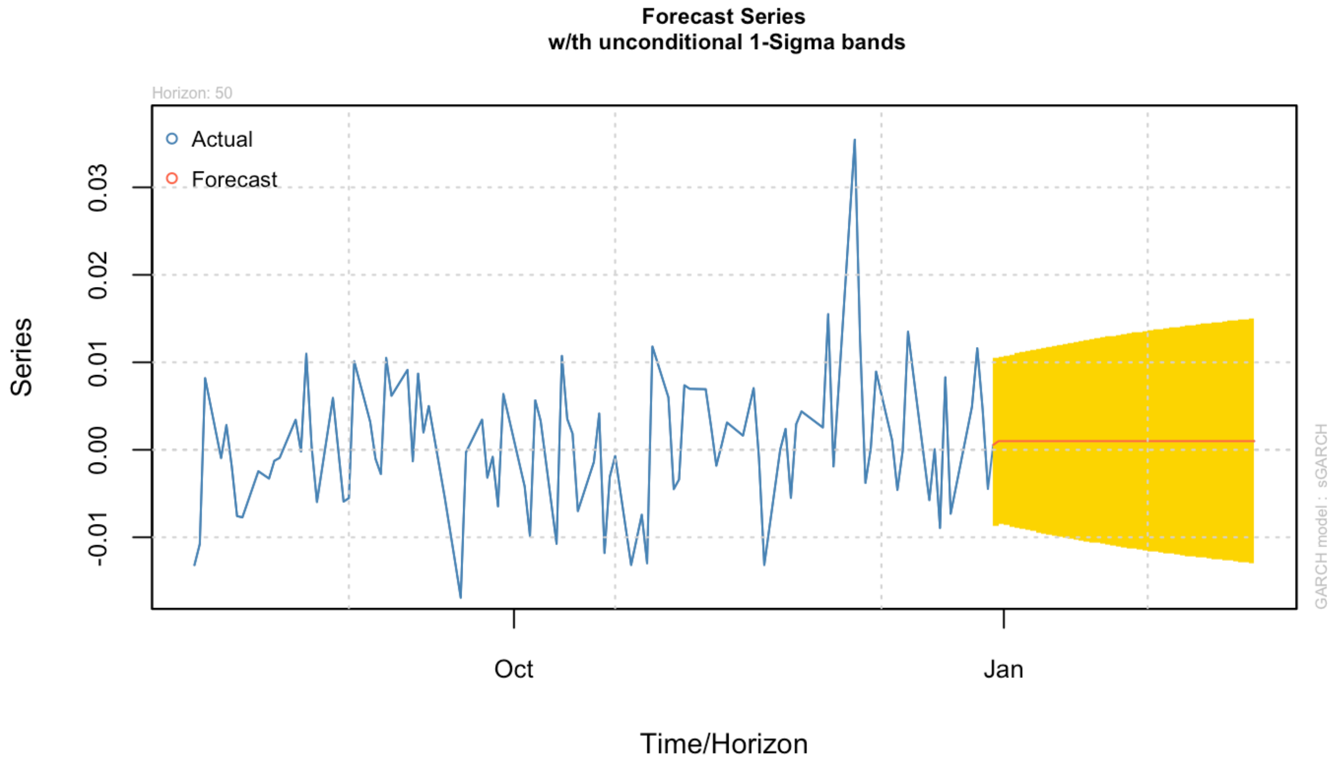
```
1
```

```
Make a plot selection (or 0 to exit):

1:   Time Series Prediction (unconditional)
2:   Time Series Prediction (rolling)
3:   Sigma Prediction (unconditional)
4:   Sigma Prediction (rolling)
```

Hide

```
3
```

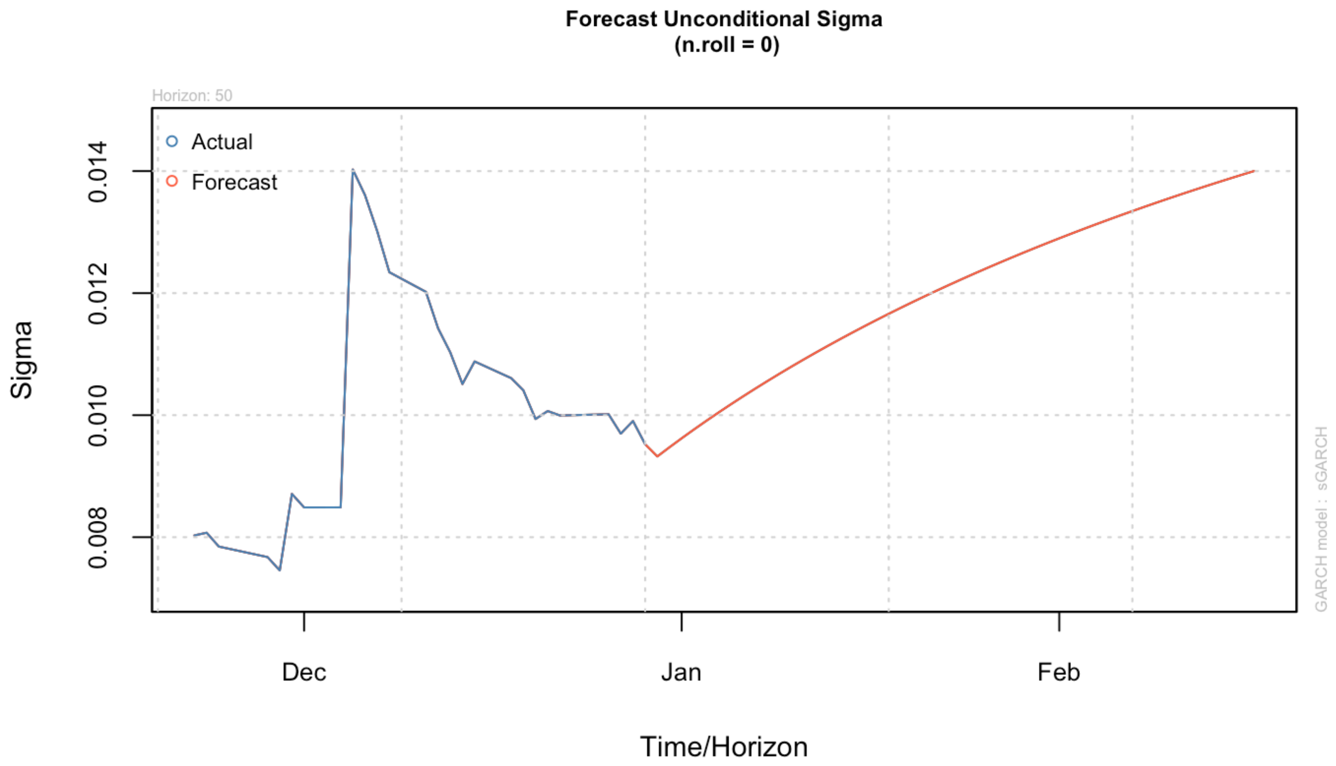


```
Make a plot selection (or 0 to exit):

1:   Time Series Prediction (unconditional)
2:   Time Series Prediction (rolling)
3:   Sigma Prediction (unconditional)
4:   Sigma Prediction (rolling)
```

Hide

```
0
```



Objective:

To forecast volatility using the fitted GARCH model for the next 50 timepoints.

Analysis:

Used the 'ugarchforecast' function to generatevolatility forecasts for the next 50 time points.

Results:GARCH Model Forecast** -

Model: sGARCH-

Horizon: 50-

Forecasted Series: - T+1 to T+50:

Contains forecasted values of volatility (Sigma) for each time point.

Implication:

The forecasted values represent the predicted volatility for the next 50time points based on the fitted GARCH model. These forecasts can beuseful for risk management and decision-making, providing insights intothe expected future volatility of the financial time series.

Thanks

Submitted by- Khushi Jain