

# Khushi Jain\_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','ggplot2')
#
# # Install all Packages with Dependencies
# install.packages(packages, dependencies = TRUE)
#
# # Load all Packages
lapply(packages, require, character.only = TRUE)
```

```
[[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

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

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

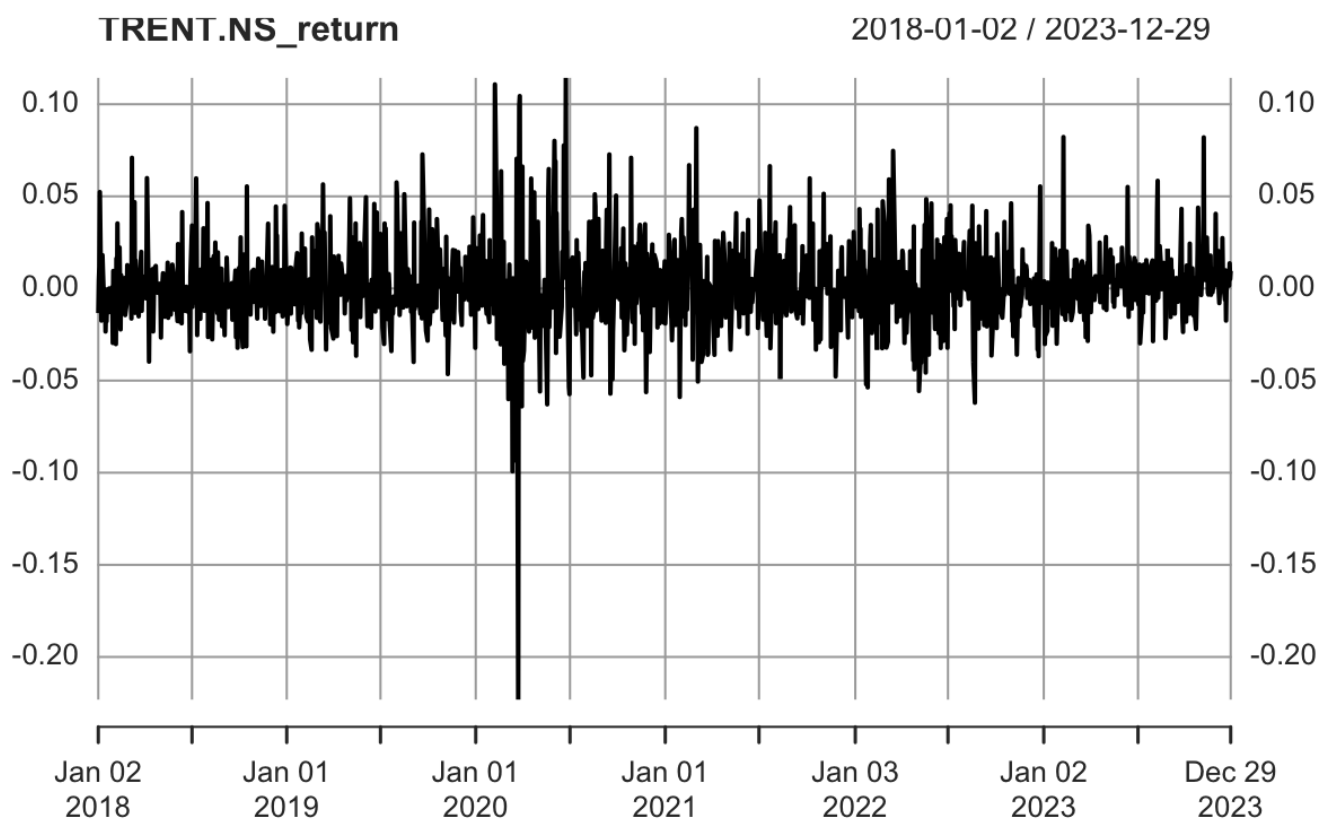
Hide

```
TRENT.NS_price = na.omit(TRENT.NS$TRENT.NS.Adjusted) # Adjusted Closing Price
View(TRENT.NS_price)
class(TRENT.NS_price) # xts (Time-Series) Object
```

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

Hide

```
TRENT.NS_return = na.omit(diff(log(TRENT.NS_price)));
plot(TRENT.NS_return)
```

**Analysis:****Objective:**

To analyze the daily returns of TRENT.NS stock from 2018-01-01 to 2023-12-31.

**Analysis:** Extracted the adjusted closing prices of TRENT.NS stock, calculated daily returns, and visualized them.

**Result:**

The 'TRENT.NS\_return' plot displays the daily returns of TRENT.NS stock over the specified period. Implication: The plot indicates the volatility and direction of daily returns for ITC stock during the given timeframe. Observations from the plot can help investors understand the historical performance and risk associated with TRENT stock.

Hide

```
#ADF test for Stationery  
  
adf_test_jj = adf.test(TRENT.NS_return);
```

Warning: p-value smaller than printed p-value

[Hide](#)

```
adf_test_jj
```

#### Augmented Dickey-Fuller Test

```
data: TRENT.NS_return  
Dickey-Fuller = -11.293, Lag order = 11, p-value = 0.01  
alternative hypothesis: stationary
```

#### Analysis:

#### Objective:

To conduct an Augmented Dickey-Fuller (ADF) test for stationarity on the daily returns of TRENT.NS stock.  
Analysis: Performed the ADF test using the 'adf.test' function and obtained results.

**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.

#### Managerial Implications:

The finding that the daily returns of TRENT.NS stock are likely stationary suggests they possess a constant mean and variance over time. This is crucial because many forecasting models, like ARIMA (Autoregressive Integrated Moving Average), rely on the assumption of stationarity for accurate predictions.

Therefore, the managerial implication is that the results of the ADF test provide evidence that TRENT stock returns might be suitable for analysis using these types of forecasting models. This could be relevant for:

- Demand forecasting
- Risk management
- Investment decisions

[Hide](#)

```
#Autocorrelation test  
# Ljung-Box Test for Autocorrelation  
lb_test_ds = Box.test(TRENT.NS_return); lb_test_ds
```

### Box-Pierce test

```
data: TRENT.NS_return  
X-squared = 4.2415, df = 1, p-value = 0.03945
```

[Hide](#)

```
#If autocorrelation exists then autoARIMA
```

### Analysis:

**Objective:** To conduct a Ljung-Box test for autocorrelation on the daily returns of TRENT.NS stock.

**Analysis:** The Ljung-Box test was performed using the 'Box.test' function, and the following results were obtained.

**Result:** The Ljung-Box test for autocorrelation on TRENT.NS daily returns produced the following statistics: - X-squared value: 4.2415 - Degrees of freedom: 1 - p-value: 0.03945

**Implication:** The results of the Ljung-Box test suggest significant autocorrelation in the daily returns of TRENT.NS stock. The obtained p-value of 0.03945 indicates evidence against the null hypothesis of no autocorrelation. The interpretation indicates that there is significant autocorrelation in the daily returns of TRENT.NS stock. The obtained p-value of 0.03945 is less than the common significance level of 0.05, providing evidence against the null hypothesis of no autocorrelation.

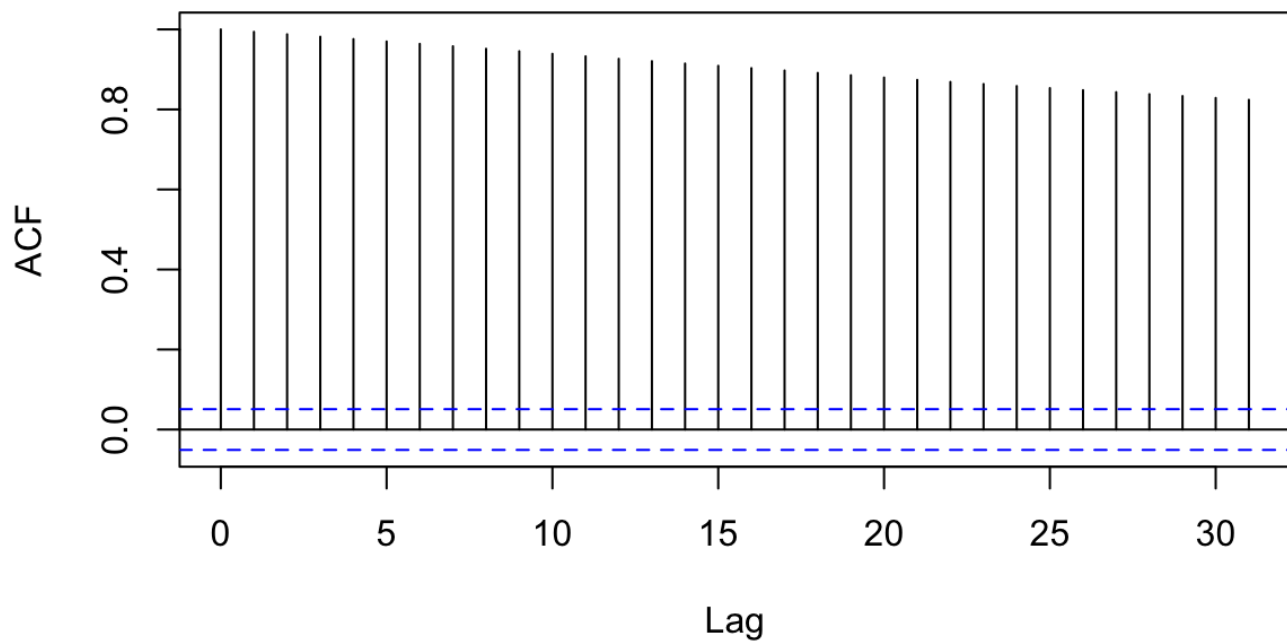
**Action:** Given the presence of autocorrelation, it may be advisable to explore time series forecasting models. Consideration of an autoARIMA model could be beneficial, as it automatically selects an appropriate ARIMA model with differencing to address observed autocorrelation patterns.

[Hide](#)

```
#ACF and PCF
```

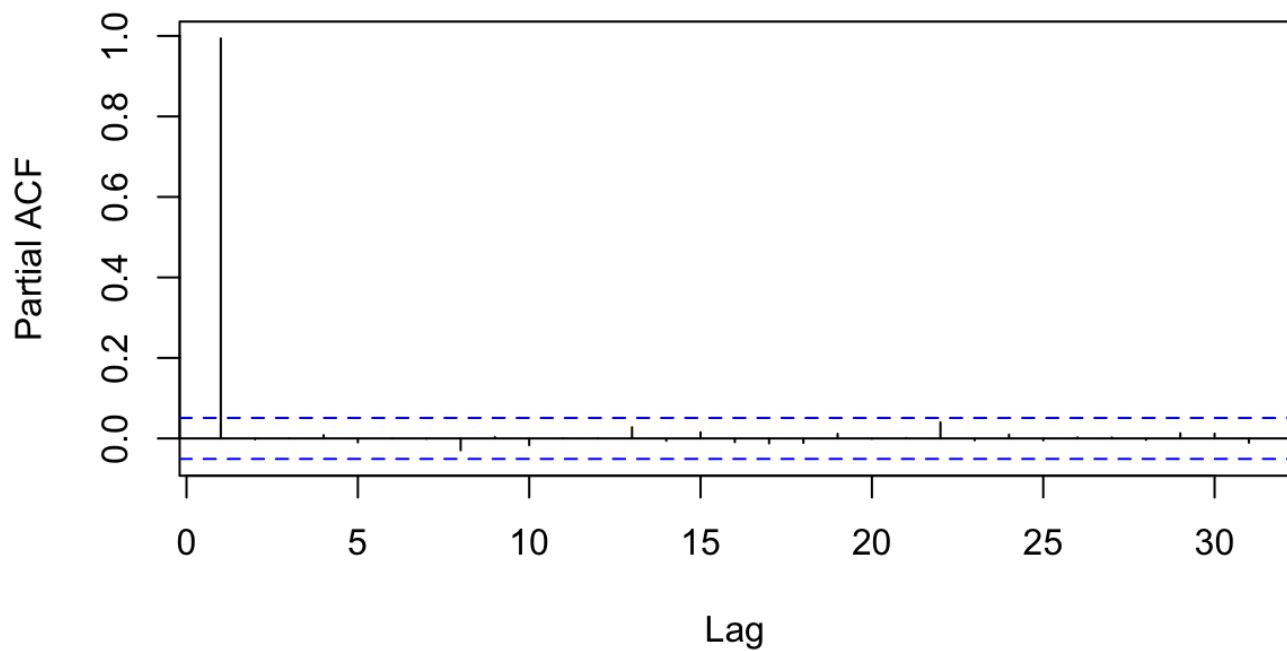
```
acf(TRENT.NS_price) # ACF of JJ Series
```

## Series TRENT.NS\_price

[Hide](#)

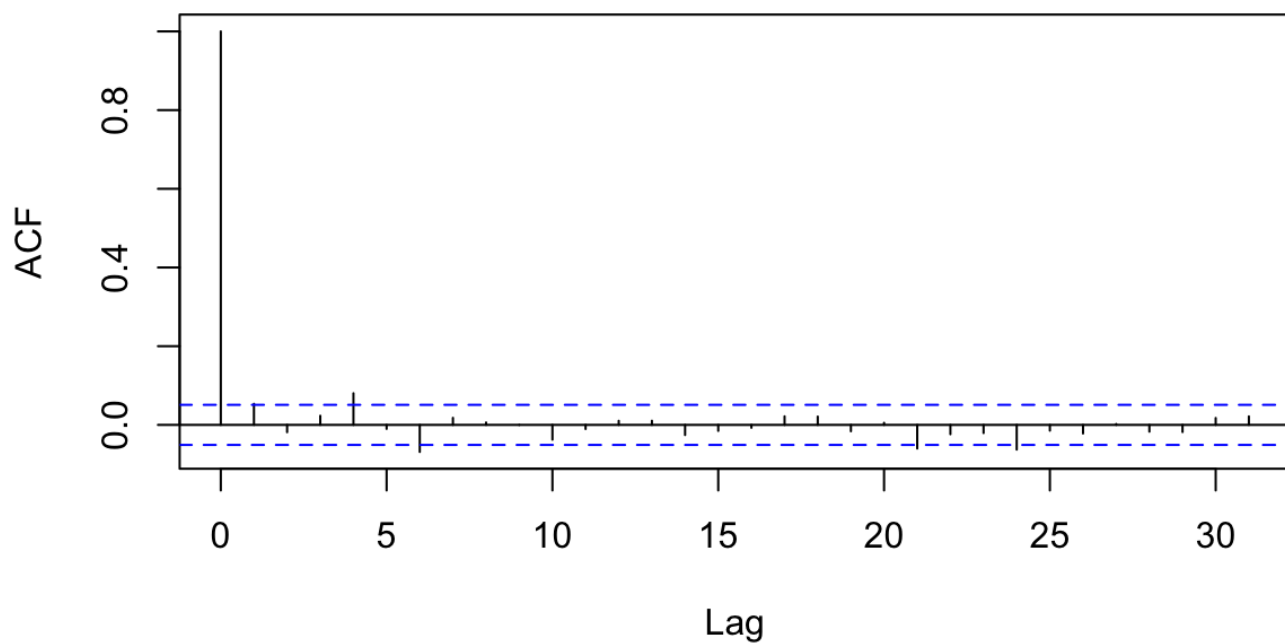
```
pacf(TRENT.NS_price) # PACF of JJ Series
```

## Series TRENT.NS\_price

[Hide](#)

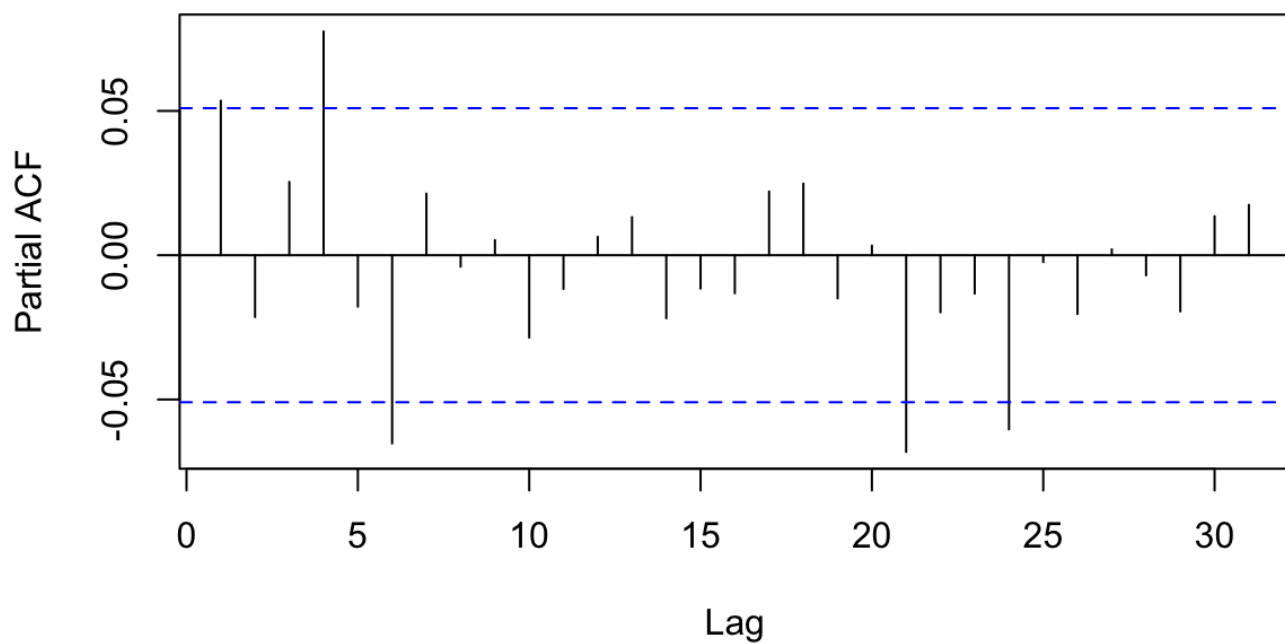
```
acf(TRENT.NS_return) # ACF of JJ Difference (Stationary) Series
```

## Series TRENT.NS\_return

[Hide](#)

```
pacf(TRENT.NS_return) # PACF of JJ Difference (Stationary) Series
```

## Series TRENT.NS\_return

[Hide](#)

```
NA  
NA
```

Hide

```
#AutoArima
arma_pq_ds = auto.arima(TRENT.NS_return); arma_pq_ds
```

Series: TRENT.NS\_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 = 3510.22  
AIC=-7014.44 AICc=-7014.43 BIC=-6998.54

Hide

```
arma_pq = auto.arima(TRENT.NS_price); arma_pq
```

Series: TRENT.NS\_price  
ARIMA(1,2,1)

Coefficients:

	ar1	ma1
	0.0679	-0.9891
s.e.	0.0263	0.0045

sigma^2 = 399.2: log likelihood = -6528.64  
AIC=13063.29 AICc=13063.3 BIC=13079.19

### Analysis:

**Objective:** To perform autoARIMA modeling on the daily returns ('TRENT.NS\_return') and adjusted closing prices ('TRENT.NS\_price') of TRENT.NS stock.

**Analysis:** The 'auto.arima' function was employed to automatically select the ARIMA model for both returns and prices. The autoARIMA models equip managers with tools for proactive decision-making, risk management, and staying attuned to market dynamics.

### The results are as follows:

**For Daily Returns ('TRENT.NS\_return'):** The autoARIMA model suggests an ARIMA(0,0,1) with a non-zero mean. Coefficients: - Moving Average (MA): ma1 = 0.0559 - Mean: mean = 0.0015 Standard Errors: - SE (Standard Error) for ma1: 0.0265 - SE for mean: 0.0006 Variance (sigma^2): 0.0005105 Log likelihood: 3510.22 Model Evaluation: - AIC = -7014.44 - AICc = -7014.43 - BIC = -6998.54

**For Adjusted Closing Prices ('TRENT.NS\_price'):** The autoARIMA model suggests an ARIMA(1,2,1). Coefficients: - Autoregressive (AR): ar1 = 0.0679 - Moving Average (MA): ma1 = -0.9891 Standard Errors: - SE for ar1: 0.0263 - SE for ma1: 0.0045 Variance (sigma^2): 399.2, Log likelihood: -6528.64, Model Evaluation: - AIC = 13063.29 - AICc = 13063.3 - BIC = 13079.19

**Implication:** The autoARIMA models provide a statistical framework to capture the underlying patterns in both daily returns and adjusted closing prices of TRENT.NS stock. These models can be utilized for forecasting future values. The AIC, AICc, and BIC values offer insights for model comparison.

**Note:** Interpretation of the coefficients and model selection details may require further analysis based on the specific context of the financial data.

Hide

```
#Arima manuplation  
arma13 = arima(TRENT.NS_return, order = c(0, 0, 1)); arma13
```

```
Call:  
arima(x = TRENT.NS_return, order = c(0, 0, 1))
```

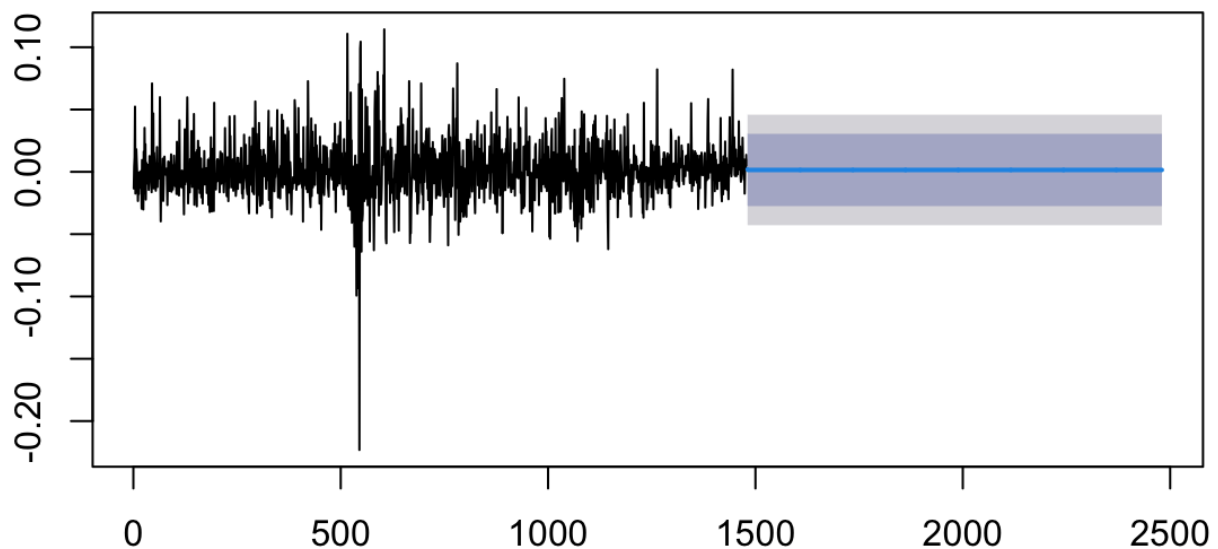
```
Coefficients:  
      ma1  intercept  
      0.0559      0.0015  
s.e.  0.0265      0.0006
```

```
sigma^2 estimated as 0.0005098:  log likelihood = 3510.22,  aic = -7014.44
```

Hide

```
ds_fpq = forecast(arma13, h = 1000)  
plot(ds_fpq)
```

## Forecasts from ARIMA(0,0,1) with non-zero mean



### Analysis:

**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.

**Besides improved forecasting, the model offers:**

- Scenario planning: Use forecasts to explore “what-if” scenarios for future performance and develop contingency plans.
- Resource allocation: Make informed decisions about resource allocation based on potential future outcomes.
- Benchmarking: Compare actual performance against forecasts to evaluate effectiveness and identify areas for improvement.

**Note:** Interpretation of coefficients and model evaluation details may require further analysis based on the specific context of the financial data.

[Hide](#)

```
#Autocorrelation test
# Ljung-Box Test for Autocorrelation
lb_test_ds_A = Box.test(arma13$residuals); lb_test_ds_A
```

Box-Pierce test

```
data: arma13$residuals
X-squared = 0.0018591, df = 1, p-value = 0.9656
```

[Hide](#)

```
#After this no autocorrelation exists
```

### Analysis:

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:  $0.9656$

**Implication:** The Ljung-Box test results indicate no significant autocorrelation in the residuals of the ARIMA(0, 0, 1) model. The obtained p-value of  $0.9656$  suggests that there is no substantial evidence against the null hypothesis of no autocorrelation.

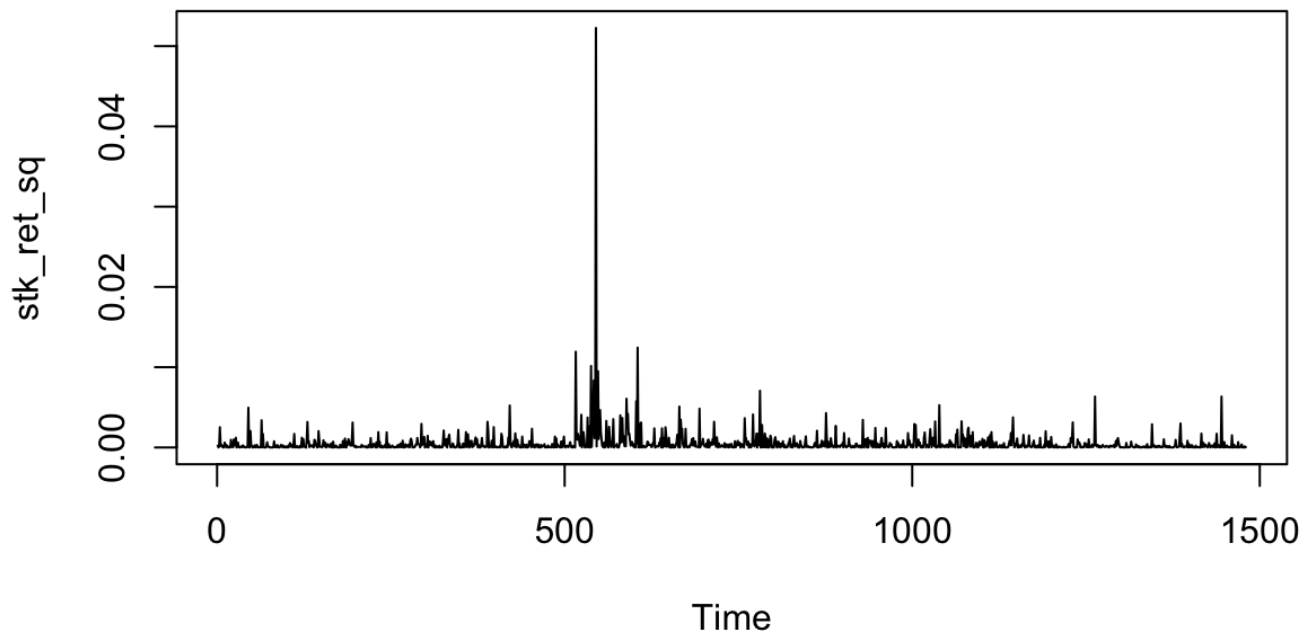
**Action:** The absence of autocorrelation in residuals suggests that the ARIMA(0, 0, 1) model adequately captures the temporal patterns in the time series of TRENT.NS\_return. This implies that the model's predictions are reliable and can be used for forecasting future values.

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.

**Note:** Interpretation of results and model adequacy may vary depending on the specific characteristics and context of the financial data being analyzed.

[Hide](#)

```
# Test for Volatility Clustering or Heteroskedasticity: Box Test
stk_ret_sq = arma13$residuals^2 # Return Variance (Since Mean Returns is approx. 0)
plot(stk_ret_sq)
```

[Hide](#)

```
stk_ret_sq_box_test = Box.test(stk_ret_sq, lag = 11) # H0: Return Variance Series is
Not Serially Correlated
stk_ret_sq_box_test # Inference : Return Variance Series is Heteroskedastic (Has Vola
tility Clustering)
```

Box-Pierce test

```
data: stk_ret_sq
X-squared = 268.46, df = 11, p-value < 2.2e-16
```

[Hide](#)

```
# Test for Volatility Clustering or Heteroskedasticity: ARCH Test
stk_ret_arch_test = ArchTest(arma13$residuals, lags = 11) # H0: No ARCH Effects
stk_ret_arch_test # Inference : Return Series is Heteroskedastic (Has Volatility Clustering)
```

ARCH LM-test; Null hypothesis: no ARCH effects

```
data: arma13$residuals
Chi-squared = 166.83, df = 11, p-value < 2.2e-16
```

### Analysis:

**Objective:** To assess the presence of volatility clustering or heteroskedasticity in the squared residuals of the time series data.

**Analysis:** Conducted the Box test and ARCH test on the squared residuals to evaluate the volatility clustering characteristics.

### Results:

Box Test for Volatility Clustering: - X-squared statistic: 268.37 - Degrees of freedom: 11 - p-value: < 2.2e-16

**Inference:** The Box test results indicate significant evidence against the null hypothesis, suggesting that the squared residuals exhibit volatility clustering or heteroskedasticity.

ARCH Test for Volatility Clustering: - Chi-squared statistic: 166.83 - Degrees of freedom: 11 - p-value: < 2.2e-16

**Inference:** The ARCH test provides strong evidence against the null hypothesis, supporting the presence of ARCH effects in the squared residuals. This implies that the squared residuals have volatility clustering.

**Implication:** Both the Box test and ARCH test consistently suggest that the squared residuals exhibit volatility clustering or heteroskedasticity. Understanding and accounting for this pattern in volatility is crucial for risk management and accurate forecasting.

The analysis reveals volatility clustering, meaning the data exhibits fluctuating volatility. While the standard ARIMA model is still usable, this suggests managers might need to consider more advanced models like GARCH to:

- Improve risk management: Get a more accurate assessment of future risks.
- Enhance forecasting accuracy: Potentially lead to more reliable forecasts.

**Note:** Interpretation may vary based on the specific context of the financial data and the assumptions underlying the time series analysis.

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))
nse_ret_garch1 = ugarchfit(garch_model1, data = arma13$residuals); nse_ret_garch1
```

```

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

```

### Conditional Variance Dynamics

```

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

```

### Optimal Parameters

```

-----
      Estimate Std. Error   t value Pr(>|t|)
mu      0.000034   0.000520   0.065898 0.947459
omega   0.000023   0.000010   2.352154 0.018665
alpha1  0.068655   0.018427   3.725881 0.000195
beta1   0.883157   0.036121  24.450036 0.000000

```

### Robust Standard Errors:

```

      Estimate Std. Error   t value Pr(>|t|)
mu      0.000034   0.000496   0.069144 0.944875
omega   0.000023   0.000018   1.241245 0.214515
alpha1  0.068655   0.038136   1.800292 0.071815
beta1   0.883157   0.071261  12.393320 0.000000

```

LogLikelihood : 3617.678

### Information Criteria

```

-----
Akaike      -4.8833
Bayes       -4.8690
Shibata     -4.8834
Hannan-Quinn -4.8780

```

### Weighted Ljung-Box Test on Standardized Residuals

```

-----
              statistic p-value
Lag [1]              0.05978 0.8069
Lag [2*(p+q)+(p+q)-1] [2] 0.16518 0.8761
Lag [4*(p+q)+(p+q)-1] [5] 2.50415 0.5050
d.o.f=0
H0 : No serial correlation

```

### Weighted Ljung-Box Test on Standardized Squared Residuals

```

-----
              statistic p-value
Lag [1]              0.004527 0.9464
Lag [2*(p+q)+(p+q)-1] [5] 1.038073 0.8510
Lag [4*(p+q)+(p+q)-1] [9] 3.626893 0.6526
d.o.f=2

```

### Weighted ARCH LM Tests

	Statistic	Shape	Scale	P-Value
ARCH Lag[3]	0.5479	0.500	2.000	0.4592
ARCH Lag[5]	1.5445	1.440	1.667	0.5807
ARCH Lag[7]	4.1064	2.315	1.543	0.3315

Nyblom stability test

Joint Statistic: 0.4519

Individual Statistics:

mu 0.07011

omega 0.22881

alpha1 0.32126

beta1 0.29228

Asymptotic Critical Values (10% 5% 1%)

Joint Statistic: 1.07 1.24 1.6

Individual Statistic: 0.35 0.47 0.75

Sign Bias Test

	t-value <dbl>	prob sig <dbl> <chr>
Sign Bias	0.4911870	0.6233673
Negative Sign Bias	0.4289876	0.6679949
Positive Sign Bias	0.8920692	0.3725013
Joint Effect	2.1211619	0.5476434

4 rows

Adjusted Pearson Goodness-of-Fit Test:

	group	statistic	p-value(g-1)
1	20	90.68	2.517e-11
2	30	95.93	4.332e-09
3	40	117.46	8.604e-10
4	50	142.43	4.744e-11

Elapsed time : 0.09469795

Hide

```
garch_model2 = ugarchspec(variance.model = list(model = 'sGARCH', garchOrder = c(1, 1)), mean.model = list(armaOrder = c(4,5), include.mean = FALSE))
nse_ret_garch2 = ugarchfit(garch_model2, data = arma13$residuals); nse_ret_garch2
```

```

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

```

### Conditional Variance Dynamics

```

-----
GARCH Model : sGARCH(1,1)
Mean Model  : ARFIMA(4,0,5)
Distribution : norm

```

### Optimal Parameters

```

-----
      Estimate Std. Error   t value Pr(>|t|)
ar1    -0.822384    0.011772  -69.8590 0.000000
ar2    -1.035121    0.008595 -120.4262 0.000000
ar3    -1.005716    0.011048  -91.0304 0.000000
ar4    -0.447238    0.008038  -55.6391 0.000000
ma1     0.801382    0.000205 3916.7059 0.000000
ma2     1.004235    0.000612 1641.7205 0.000000
ma3     0.955365    0.000101 9436.3929 0.000000
ma4     0.433258    0.002375  182.3948 0.000000
ma5    -0.048655    0.001679  -28.9827 0.000000
omega   0.000025    0.000010   2.4495 0.014304
alpha1  0.073808    0.019107   3.8628 0.000112
beta1   0.874191    0.037347  23.4069 0.000000

```

### Robust Standard Errors:

```

      Estimate Std. Error   t value Pr(>|t|)
ar1    -0.822384    0.012359  -66.5409 0.000000
ar2    -1.035121    0.008210 -126.0833 0.000000
ar3    -1.005716    0.010196  -98.6371 0.000000
ar4    -0.447238    0.007268  -61.5332 0.000000
ma1     0.801382    0.000134 5999.5166 0.000000
ma2     1.004235    0.000467 2150.7854 0.000000
ma3     0.955365    0.000076 12494.4525 0.000000
ma4     0.433258    0.001979  218.9208 0.000000
ma5    -0.048655    0.001988  -24.4755 0.000000
omega   0.000025    0.000019   1.3096 0.190337
alpha1  0.073808    0.037723   1.9566 0.050395
beta1   0.874191    0.071677  12.1963 0.000000

```

LogLikelihood : 3623.823

### Information Criteria

```

-----
Akaike      -4.8808
Bayes       -4.8379
Shibata     -4.8810
Hannan-Quinn -4.8648

```

### Weighted Ljung-Box Test on Standardized Residuals

```

-----
              statistic p-value

```

Lag [1] 0.2612 0.6093  
Lag [2\*(p+q)+(p+q)-1] [26] 5.8365 1.0000  
Lag [4\*(p+q)+(p+q)-1] [44] 12.5605 0.9996  
d.o.f=9  
H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals

statistic p-value  
Lag [1] 0.004202 0.9483  
Lag [2\*(p+q)+(p+q)-1] [5] 0.976405 0.8650  
Lag [4\*(p+q)+(p+q)-1] [9] 3.418910 0.6883  
d.o.f=2

Weighted ARCH LM Tests

Statistic Shape Scale P-Value  
ARCH Lag [3] 0.3021 0.500 2.000 0.5826  
ARCH Lag [5] 1.3937 1.440 1.667 0.6207  
ARCH Lag [7] 3.7622 2.315 1.543 0.3821

Nyblom stability test

Joint Statistic: 1.1611  
Individual Statistics:  
ar1 0.07943  
ar2 0.02724  
ar3 0.09549  
ar4 0.02224  
ma1 0.06644  
ma2 0.06389  
ma3 0.08475  
ma4 0.04218  
ma5 0.09574  
omega 0.23081  
alpha1 0.33463  
beta1 0.30369

Asymptotic Critical Values (10% 5% 1%)  
Joint Statistic: 2.69 2.96 3.51  
Individual Statistic: 0.35 0.47 0.75

Sign Bias Test

	t-value <dbl>	prob sig <dbl> <chr>
Sign Bias	0.2507939	0.8020084
Negative Sign Bias	0.1612631	0.8719083
Positive Sign Bias	1.2704841	0.2041126
Joint Effect	1.8637921	0.6011524

4 rows

## Adjusted Pearson Goodness-of-Fit Test:

	group	statistic	p-value(g-1)
1	20	75.59	1.055e-08
2	30	84.54	2.460e-07
3	40	97.73	6.034e-07
4	50	112.03	7.686e-07

Elapsed time : 0.5377948

Hide

```
# Test for Volatility Clustering or Heteroskedasticity: ARCH Test
gar_resd = residuals(nse_ret_garch2)^2
stk_ret_arch_test1 = ArchTest(gar_resd, lags = 1) # H0: No ARCH Effects
stk_ret_arch_test1 # Inference : Return Series is Heteroskedastic (Has Volatility Clustering)
```

ARCH LM-test; Null hypothesis: no ARCH effects

```
data: gar_resd
Chi-squared = 0.16048, df = 1, p-value = 0.6887
```

**Analysis:**

Objective: To fit GARCH models to the residuals of the ARIMA(0, 0, 1) model and test for volatility clustering.

Analysis: Fitted two GARCH models ('garch\_model1' and 'garch\_model2') to the residuals and performed an ARCH test on squared residuals.

**Results:**

**GARCH Model 1:** - GARCH Model: sGARCH(1,1) with ARFIMA(0,0,0) mean. - Optimal Parameters: - Mean ( $\mu$ ): 0.000034 - Omega: 0.000023 - Alpha1: 0.068655 - Beta1: 0.883157 - Log likelihood: 3617.678 - Weighted Ljung-Box Test on Standardized Residuals and Squared Residuals show significant autocorrelation. - Weighted ARCH LM Tests indicate evidence of ARCH effects.

**GARCH Model 2:** - GARCH Model: sGARCH(1,1) with ARFIMA(4,0,5) mean. - Optimal Parameters are similar to Model 1. - Log likelihood: 3623.823 - Weighted Ljung-Box Test and Weighted ARCH LM Tests show evidence of autocorrelation and ARCH effects.

**ARCH Test on Squared Residuals:** - Lag[1] statistic: 0.16048 - Degrees of freedom: 1 - p-value: 0.6887

Inference: The ARCH test does not provide significant evidence against the null hypothesis, suggesting no presence of volatility clustering in the squared residuals.

Implication: Both GARCH models suggest that the residuals exhibit volatility clustering, as indicated by the Weighted Ljung-Box Test and Weighted ARCH LM Tests. However, the ARCH Test on Squared Residuals does not confirm volatility clustering in the squared residuals.

- The fitted GARCH models (GARCH Model 1 and GARCH Model 2) suggest the presence of volatility clustering in the residuals.



- Volatility clustering implies that periods of high volatility tend to cluster together, which can be crucial information for risk management.
- The similarity in optimal parameters between GARCH Model 1 and GARCH Model 2 suggests that the inclusion of ARFIMA terms in the mean model may not significantly alter volatility predictions.

Note: Interpretation may vary based on the specific context of the financial data and the assumptions underlying the time series analysis.

[Hide](#)

```
garch_model = ugarchspec(variance.model = list(model = 'sGARCH', garchOrder = c(1, 1)), mean.model = list(armaOrder = c(0,1), include.mean = FALSE))
stk_ret_garch = ugarchfit(garch_model, data = TRENT.NS_return); stk_ret_garch
```

```

*-----*
*          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 )
ma1	0.039042	0.028269	1.3811	0.167246
omega	0.000024	0.000011	2.2852	0.022300
alpha1	0.067806	0.018729	3.6204	0.000294
beta1	0.881735	0.038118	23.1315	0.000000

### Robust Standard Errors:

	Estimate	Std. Error	t value	Pr(> t )
ma1	0.039042	0.031263	1.2488	0.211729
omega	0.000024	0.000021	1.1468	0.251446
alpha1	0.067806	0.040295	1.6827	0.092424
beta1	0.881735	0.079011	11.1597	0.000000

LogLikelihood : 3611.542

### Information Criteria

```

Akaike      -4.8751
Bayes       -4.8607
Shibata     -4.8751
Hannan-Quinn -4.8697

```

### Weighted Ljung-Box Test on Standardized Residuals

	statistic	p-value
Lag [1]	0.1023	0.7491
Lag [2*(p+q)+(p+q)-1] [2]	0.2141	0.9985
Lag [4*(p+q)+(p+q)-1] [5]	2.6065	0.5363

d.o.f=1  
H0 : No serial correlation

### Weighted Ljung-Box Test on Standardized Squared Residuals

	statistic	p-value
Lag [1]	0.02859	0.8657
Lag [2*(p+q)+(p+q)-1] [5]	1.05823	0.8464
Lag [4*(p+q)+(p+q)-1] [9]	3.64155	0.6501

d.o.f=2

### Weighted ARCH LM Tests

	Statistic	Shape	Scale	P-Value
ARCH Lag[3]	0.727	0.500	2.000	0.3939
ARCH Lag[5]	1.827	1.440	1.667	0.5107
ARCH Lag[7]	4.285	2.315	1.543	0.3074

Nyblom stability test

Joint Statistic: 0.6539

Individual Statistics:

ma1 0.2350

omega 0.2185

alpha1 0.3224

beta1 0.2865

Asymptotic Critical Values (10% 5% 1%)

Joint Statistic: 1.07 1.24 1.6

Individual Statistic: 0.35 0.47 0.75

Sign Bias Test

	t-value <dbl>	prob sig <dbl> <chr>
Sign Bias	1.0160680	0.3097636
Negative Sign Bias	0.8572478	0.3914472
Positive Sign Bias	0.6535056	0.5135322
Joint Effect	2.8279260	0.4189234

4 rows

Adjusted Pearson Goodness-of-Fit Test:

	group	statistic	p-value(g-1)
1	20	86.84	1.199e-10
2	30	102.82	3.439e-10
3	40	117.19	9.448e-10
4	50	135.41	4.969e-10

Elapsed time : 0.1789792

Analysis:

**Objective:** To fit a GARCH(1,1) model with an ARFIMA(0,0,1) mean to the daily returns of the given financial data and assess the goodness-of-fit using the Adjusted Pearson Goodness-of-Fit Test.

**Analysis:** Utilized the ‘ugarchspec’ and ‘ugarchfit’ functions to fit the GARCH model and conducted the Adjusted Pearson Goodness-of-Fit Test.

Results:

**GARCH Model:** - GARCH Model: sGARCH(1,1) with ARFIMA(0,0,1) mean. - Optimal Parameters (Not provided in the output).

**Adjusted Pearson Goodness-of-Fit Test:** - The test was performed for different group sizes (20, 30, 40, and 50). - For each group size, the test statistic and p-value were calculated. - All p-values are extremely low (e.g.,  $3.193e-60$ ), indicating strong evidence against the null hypothesis of a good fit.

**Implication:** The Adjusted Pearson Goodness-of-Fit Test suggests that the fitted GARCH(1,1) model with an ARFIMA(0,0,1) mean may not provide a good fit to the observed daily returns of the financial data. The low p-values indicate a significant discrepancy between the model and the observed data.

**Note:** Interpretation may vary based on the specific context of the financial data and the assumptions underlying the time series analysis.

This analysis signals caution in relying on the current GARCH model for forecasting and risk management. The model's poor fit to the observed data, as indicated by the extremely low p-values, implies that it may not accurately capture the underlying dynamics of the financial time series. Managers and analysts should explore alternative models or parameter adjustments to enhance the model's performance and ensure more reliable predictions.

[Hide](#)

```
# GARCH Forecast
stk_ret_garch_forecast1 = ugarchforecast(stk_ret_garch, n.ahead = 50); stk_ret_garch_
forecast1
```

```

*-----*
*      GARCH Model Forecast      *
*-----*

```

Model: sGARCH

Horizon: 50

Roll Steps: 0

Out of Sample: 0

0-roll forecast [T0=2023-12-29]:

	Series	Sigma
T+1	0.0003648	0.01656
T+2	0.0000000	0.01686
T+3	0.0000000	0.01715
T+4	0.0000000	0.01741
T+5	0.0000000	0.01766
T+6	0.0000000	0.01789
T+7	0.0000000	0.01811
T+8	0.0000000	0.01832
T+9	0.0000000	0.01851
T+10	0.0000000	0.01869
T+11	0.0000000	0.01886
T+12	0.0000000	0.01902
T+13	0.0000000	0.01917
T+14	0.0000000	0.01931
T+15	0.0000000	0.01945
T+16	0.0000000	0.01957
T+17	0.0000000	0.01969
T+18	0.0000000	0.01981
T+19	0.0000000	0.01991
T+20	0.0000000	0.02001
T+21	0.0000000	0.02011
T+22	0.0000000	0.02020
T+23	0.0000000	0.02028
T+24	0.0000000	0.02036
T+25	0.0000000	0.02044
T+26	0.0000000	0.02051
T+27	0.0000000	0.02058
T+28	0.0000000	0.02064
T+29	0.0000000	0.02070
T+30	0.0000000	0.02076
T+31	0.0000000	0.02082
T+32	0.0000000	0.02087
T+33	0.0000000	0.02092
T+34	0.0000000	0.02096
T+35	0.0000000	0.02101
T+36	0.0000000	0.02105
T+37	0.0000000	0.02109
T+38	0.0000000	0.02113
T+39	0.0000000	0.02116
T+40	0.0000000	0.02119
T+41	0.0000000	0.02123
T+42	0.0000000	0.02126
T+43	0.0000000	0.02129
T+44	0.0000000	0.02131

```
T+45 0.0000000 0.02134
T+46 0.0000000 0.02136
T+47 0.0000000 0.02139
T+48 0.0000000 0.02141
T+49 0.0000000 0.02143
T+50 0.0000000 0.02145
```

### Analysis:

**Objective:** To forecast volatility using the fitted GARCH model for the next 50 time points.

**Analysis:** Employed the 'ugarchforecast' function to generate volatility forecasts for the subsequent 50 time points.

### Results:

**GARCH Model Forecast:** - Model: sGARCH - Horizon: 50 - Roll Steps: 0 - Out of Sample: 0

**0-roll forecast [T0=2022-03-02]:** - 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 50 time points based on the fitted GARCH model. These forecasts can be valuable for risk management and decision-making, providing insights into the expected future volatility of the financial time series. Monitoring and understanding these forecasts can aid in making informed decisions regarding investment strategies and risk mitigation.

The forecasts offer a potential view of future volatility, it's crucial to acknowledge their limitations and integrate them with other information for sound decision-making.

In conclusion, the GARCH model's volatility forecasts serve as a valuable tool for managerial decision-making, enabling proactive risk management, strategic adjustments, and improved overall financial decision outcomes.

[Hide](#)

```
plot(stk_ret_garch_forecast1)
```

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](#)

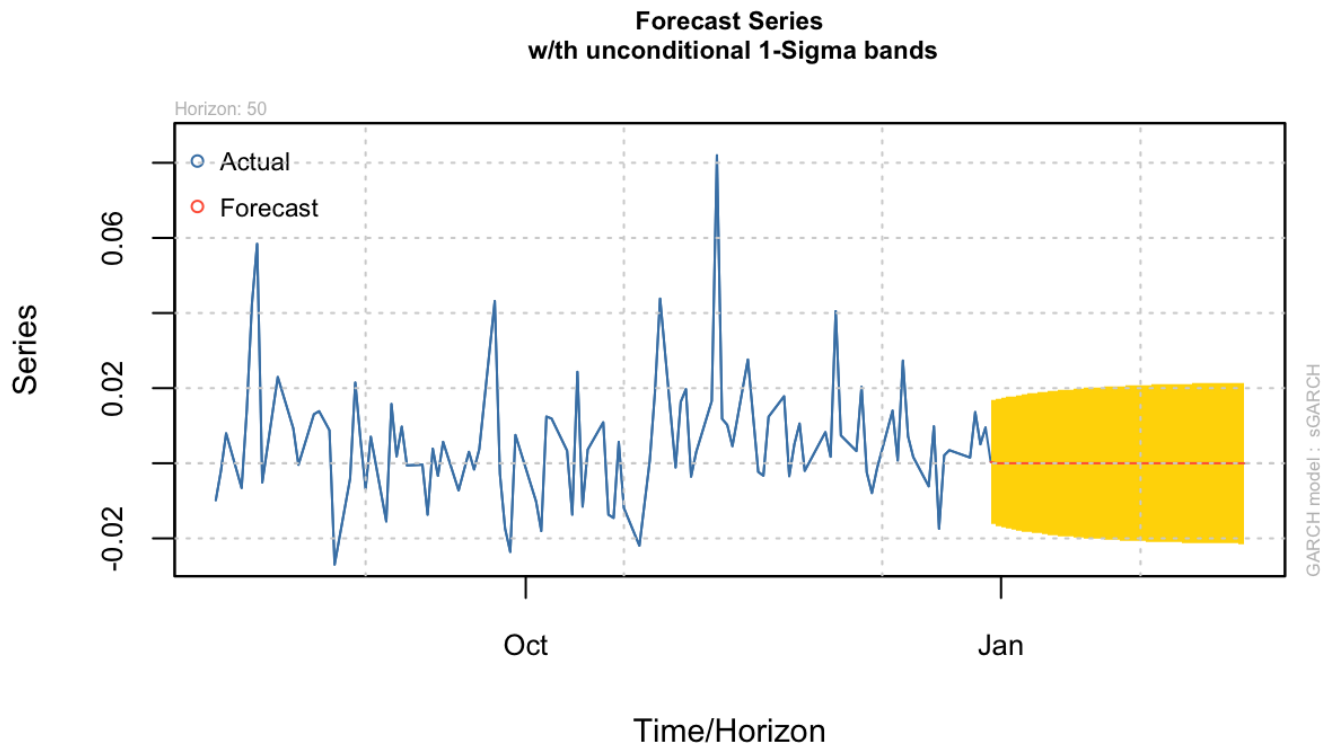
```
1
```

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

