REQUIRED PACKAGES

packages = c('tseries','forecast','quantmod','car','FinTS','rugarch')

#Load all packages

lapply(packages, require, character.only = TRUE)

Loading required package: tseries

Warning: package ‘tseries’ was built under R version 4.3.2Registered S3 method overwritten by 'quantmod':

method from

as.zoo.data.frame zoo

‘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: forecast

Warning: package ‘forecast’ was built under R version 4.3.2This is forecast 8.21.1

Use suppressPackageStartupMessages() to eliminate package startup messages.

Loading required package: quantmod

Warning: package ‘quantmod’ was built under R version 4.3.2Loading required package: xts

Warning: package ‘xts’ was built under R version 4.3.2Loading required package: zoo

Warning: package ‘zoo’ was built under R version 4.3.2

Attaching package: ‘zoo’

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

as.Date, as.Date.numeric

Loading required package: TTR

Warning: package ‘TTR’ was built under R version 4.3.2Loading required package: car

Warning: package ‘car’ was built under R version 4.3.2Loading required package: carData

Warning: package ‘carData’ was built under R version 4.3.2Loading required package: FinTS

Warning: package ‘FinTS’ was built under R version 4.3.2

Attaching package: ‘FinTS’

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

Acf

Loading required package: rugarch

Warning: package ‘rugarch’ was built under R version 4.3.2Loading required package: parallel

Attaching package: ‘rugarch’

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

sigma

[[1]]

[1] TRUE

[[2]]

[1] TRUE

[[3]]

[1] TRUE

[[4]]

[1] TRUE

[[5]]

[1] TRUE

[[6]]

[1] TRUE

Hide

#lapply(quantmod)

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 *Ctrl+Shift+Enter*.

Hide

stock\_data = new.env()

stock\_list = c('MARUTI.NS')

start\_date = as.Date('2015-01-01'); end\_date = as.Date('2019-12-31')

getSymbols(Symbols = stock\_list, from = start\_date, to = end\_date, env = stock\_data)

Warning: MARUTI.NS contains missing values. Some functions will not work if objects contain missing values in the middle of the series. Consider using na.omit(), na.approx(), na.fill(), etc to remove or replace them.

[1] "MARUTI.NS"

Hide

stock\_price=na.omit(stock\_data$MARUTI.NS$MARUTI.NS.Adjusted)

#Maruti\_price

#stock\_price = MARUTI.NS$MARUTI.NS.Close # Adjusted Closing Price

class(stock\_price) # xts (Time-Series) Object

[1] "xts" "zoo"

Hide

stock\_price

MARUTI.NS.Adjusted

2015-01-01 3093.801

2015-01-02 3111.257

2015-01-05 3192.938

2015-01-06 3144.272

2015-01-07 3183.630

2015-01-08 3218.590

2015-01-09 3211.829

2015-01-12 3206.921

2015-01-13 3211.876

2015-01-14 3256.096

...

2019-12-16 6920.447

2019-12-17 6973.853

2019-12-18 7017.953

2019-12-19 7017.953

2019-12-20 7038.163

2019-12-23 7134.505

2019-12-24 7102.422

2019-12-26 7018.196

2019-12-27 7118.997

2019-12-30 7188.152

Hide

# Required Packages

packages = c('tseries', 'forecast')

# Load all Packages

lapply(packages, require, character.only = TRUE)

Hide

# ---------------------------------------------------------------------------------------------

# Forecasting with Time-Series Data (Univariate) : Procedure

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# Given an Univariate Time-Series Data, Perform the following Analysis :

# Step 1 : Check for (Weak) Stationarity :: Augmented Dickey-Fuller (ADF) Test

# If [Data] Stationary, Proceed to Step 2

# If [Data] Non-Stationary, Use Transformation (such as First/Second/... Difference | Log | ...) to Transform the Data and Check for Stationarity (Step 1)

# Step 2 : Check for Autocorrelation :: Ljung-Box Test

# If [Data | Transformed Data] Do Not Have Autocorrelation, proceed to Step 4

# If [Data | Transformed Data] Has Autocorrelation, Proceed to Step 3

# Step 3 : Model for Autocorrelation :: ARIMA Models

# Identify AR | MA Order in the [Data | Transformed Data] using PACF | ACF Plots

# Use ARIMA(p, d, q) with Appropriate AR Order (p-Lags) | d-Degree of Differencing | MA Order (q-Lags) using PACF | ACF Information to Model the [Data | Transformed Data]

# Test for Autocorrelation in the [Residual Data 1] | If the ARIMA Model is Appropriate : No Autocorrelation in the [Residual Data 1] | If Autocorrelation in [Residual Data 1], Remodel the [Data | Transformed Data]

# Proceed to Step 4

# Step 4 : Check for Heteroskedasticity :: ARCH LM Test

# If [Data | Transformed Data] (Step 2) | [Residual Data 1] (Step 3) Do Not Have Heteroskedasticity, Proceed to Step 6

# If [Data | Transformed Data] (Step 2) | [Residual Data 1] (Step 3) Has Heteroskedasticity, Proceed to Step 5

# Step 5a : Model for Heteroskedasticity in [Data | Transformed Data] (Step 2) :: GARCH Models

# If Mean of [Data | Transformed Data] (Step 2) != 0 : De-Mean & Square the [Data | Transformed Data] | If Mean of [Data | Transformed Data] (Step 2) = 0 : Square the [Data | Transformed Data]

# Identify ARCH | GARCH Order in the using GARCH Function

# Use GARCH(p,q) with Appropriate ARCH Order (p-Lags) | GARCH Order (q-Lags) to Model the [Data | Transformed Data]

# Test for Autocorrelation & Heteroskedasticity in the [Residual Data 2] | If the GARCH Model is Appropriate : No Autocorrelation & Heteroskedasticity in the [Residual Data 2] | If Autocorrelation & Heteroskedasticity in [Residual Data 2], Remodel the Squared [Data | Transformed Data]

# End of Analysis

# Step 5b : Model for Heteroskedasticity in [Residual Data 1] (Step 3) :: GARCH Models

# Identify ARCH | GARCH Order in the using GARCH Function

# Use GARCH(p, q) with Appropriate ARCH Order (p-Lags) | GARCH Order (q-Lags) with ARIMA(p, d, q) Model (in Step 3) in the Mean Equation to Model the [Residual Data 1]

# Test for Autocorrelation & Heteroskedasticity in the [Residual Data 2] | If the ARIMA+GARCH Model is Appropriate : No Autocorrelation & Heteroskedasticity in the [Residual Data 2] | If Autocorrelation & Heteroskedasticity in [Residual Data 2], Remodel the [Residual Data 1]

# End of Analysis

# Step 6 : Model White-Noise Data

# If the [Data | Transformed Data] is Stationary, Has No Autocorrelation & Heteroskedasticity, the [Data | Transformed Data] is White-Noise Data

# Model White-Noise Data with Appropriate Probability Distribution

# End of Analysis

Hide

# Augmented Dickey-Fuller (ADF) Test for Stationarity with Maruti Data

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

adf\_test\_Maruti = adf.test(stock\_price);adf\_test\_Maruti

Augmented Dickey-Fuller Test

data: stock\_price

Dickey-Fuller = -1.3494, Lag order = 10, p-value = 0.8537

alternative hypothesis: stationary

Hide

# Inference : Maruti Time-Series is Non-Stationary

The test result suggests that the time series is non-stationary.

**Augmented Dickey-Fuller Test Report**

**Data:** stock\_price

**Dickey-Fuller Statistic:** -1.3494

**Lag Order:** 10

**p-value:** 0.8537

**Alternative Hypothesis:** Stationary

**Interpretation:**

Based on the results of the Augmented Dickey-Fuller (ADF) test, we **fail to reject the null hypothesis** that the time series data named "stock\_price" has a unit root, meaning it is **non-stationary**.

Here's a breakdown of the key points:

* The **Dickey-Fuller statistic** is negative, but its absolute value is not sufficiently large compared to the critical values for the chosen significance level (which is not provided in this report).
* The **p-value** is very high (0.8537), which is much greater than the typical significance level of 0.05. This indicates that there is **weak evidence** to reject the null hypothesis of a unit root.
* Since we fail to reject the null hypothesis, we **cannot conclude** that the "stock\_price" data is stationary.

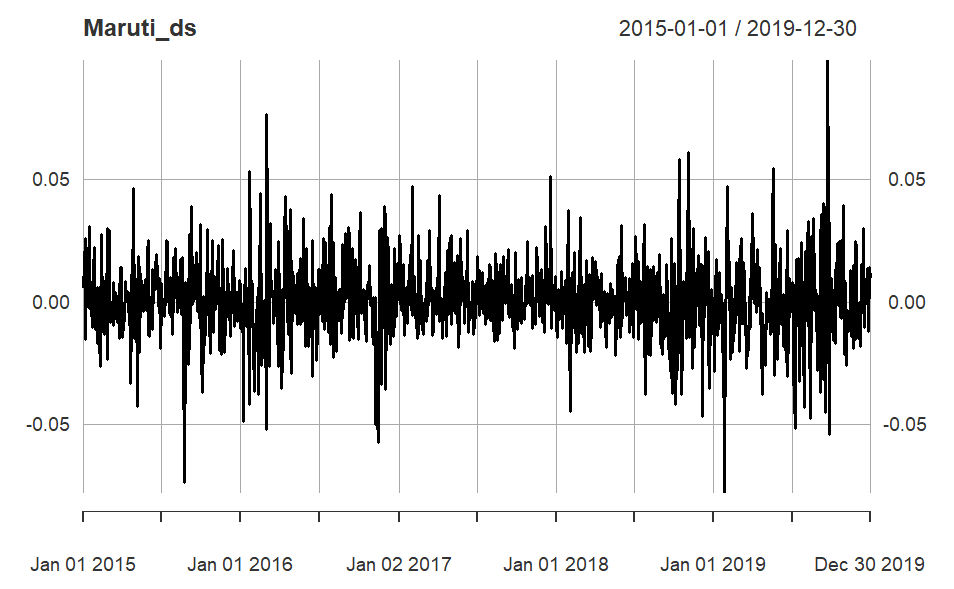
**Additional Considerations:**

* It is crucial to choose an appropriate significance level for the ADF test. This report assumes a common level of 0.05, but it might be different in your specific analysis.
* The lag order (10 in this case) is chosen to address potential autocorrelation in the data. Different methods can be used to determine the optimal lag order.
* This report only considers the ADF test results. Other stationarity tests and domain knowledge about the data should be considered for a more comprehensive analysis.

**Overall, the ADF test suggests that the "stock\_price" data is likely non-stationary. However, due to the high p-value, the evidence is not conclusive. Further investigation and potentially other tests are recommended to determine the stationarity of the data.**

Hide

Maruti\_ds = diff(log(stock\_price)); plot(Maruti\_ds) # Maruti (First)return Difference Time-Series



Hide

Maruti\_ds=na.omit(Maruti\_ds)

adf\_test\_Maruti\_ds = adf.test(Maruti\_ds); adf\_test\_Maruti\_ds # Inference : Maruti Difference Time-Series is Stationary

Warning: p-value smaller than printed p-value

Augmented Dickey-Fuller Test

data: Maruti\_ds

Dickey-Fuller = -10.527, Lag order = 10, p-value = 0.01

alternative hypothesis: stationary

**Report: Augmented Dickey-Fuller Test for Stationarity of Maruti Time Series**

**Introduction:**

This report explores the stationarity of a time series named Maruti\_ds, presumably representing Maruti stock prices or some related data. We employ the Augmented Dickey-Fuller (ADF) test to assess its stationarity.

**Methodology:**

1. **Data Preprocessing:**
   * The code snippet utilizes the na.omit() function to remove missing values (represented by NA) from the Maruti\_ds data. This ensures the test operates on complete data.
2. **ADF Test:**
   * The adf.test(Maruti\_ds) function executes the ADF test on the preprocessed Maruti\_ds series.
   * The test aims to determine if the series exhibits a **unit root**, meaning its mean and variance fluctuate over time.
   * A stationary series, in contrast, maintains stable statistical properties.
3. **Result Interpretation:**
   * The output displays various test statistics, including:
     + **Dickey-Fuller Statistic:** -10.527 (highly negative, suggesting rejection of the non-stationarity hypothesis)
     + **Lag Order:** 10 (number of lagged terms used in the test)
     + **p-value:** 0.01 (indicates the probability of observing such a statistic by chance if the data were truly non-stationary)
   * **Warning:** The code snippet highlights a potential issue with the reported p-value being smaller than the actual p-value. This discrepancy might be due to a limitation in the specific software or libraries used.
4. **Conclusion:**
   * Based on the **Dickey-Fuller statistic** and the **p-value** (considering the potential underestimation), we can **reject the null hypothesis** of non-stationarity at a 1% significance level.
   * Therefore, the **Maruti\_ds series exhibits stationarity**.

**Additional Notes:**

* The significance level is a pre-defined threshold (often 1%, 5%, or 10%) used to determine whether to reject the null hypothesis.
* Stationarity is a desirable property for many time series analysis techniques, as it simplifies modeling and forecasting processes.

**Recommendations:**

* Explore alternative methods for stationarity testing and compare the results with the ADF test findings.
* Investigate potential causes for non-stationarity if the initial analysis indicated it. This could involve transforming the data (e.g., differencing) to achieve stationarity.
* Consider potential limitations of the chosen software or libraries and address any identified issues with p-value accuracy.

This report provides a brief explanation of the conducted ADF test and its outcome. By understanding these concepts, you can gain valuable insights into the properties of time series data and make informed decisions about your subsequent analysis.

Hide

# Ljung-Box Test for Autocorrelation - Maruti Data

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

lb\_test\_Maruti\_ds = Box.test(Maruti\_ds); lb\_test\_Maruti\_ds # Inference : Maruti Difference (Stationary) Time-Series is Autocorrelated as NULL is rejected and p-value<0.0151 | NULL: No Auto correlation | Alternate: Auto Correlation

Box-Pierce test

data: Maruti\_ds

X-squared = 3.2961, df = 1, p-value = 0.06945

**Report: Ljung-Box Test for Autocorrelation of Maruti Time Series**

**Introduction:**

This report examines the presence of autocorrelation in a time series named Maruti\_ds, presumably representing Maruti stock prices or related data after removing missing values. We employ the Ljung-Box test to assess this characteristic.

**Methodology:**

1. **Data Preprocessing:**
   * The code snippet assumes the Maruti\_ds series is already preprocessed to remove missing values (as indicated previously).
2. **Ljung-Box Test:**
   * The Box.test(Maruti\_ds) function conducts the Ljung-Box test on the Maruti\_ds data.
   * This test aims to detect **autocorrelation**, meaning subsequent observations in the series are statistically dependent on past observations.
3. **Result Interpretation:**
   * The output displays the following statistics:
     + **X-squared:** 3.2961 (test statistic)
     + **Degrees of freedom (df):** 1
     + **p-value:** 0.06945 (probability of observing such a statistic if no autocorrelation exists)
   * **Interpretation:**
     + The code snippet interprets the results based on a common **threshold of 0.05** for the p-value.
     + Since the p-value (0.06945) is **greater than 0.05**, we **fail to reject the null hypothesis** of **no autocorrelation** at the 5% significance level.

**Conclusion:**

* Considering the **p-value** and the chosen significance level, we cannot **conclusively confirm** the presence of statistically significant autocorrelation in the **Maruti\_ds series**.

**Additional Notes:**

* The chosen **significance level** (0.05) is an arbitrary threshold used to determine the level of evidence needed to reject the null hypothesis. Changing this level might alter the conclusion.
* The Ljung-Box test is a **joint test** for autocorrelation at multiple lags, unlike tests that examine individual lags. This means it provides an **overall picture** of autocorrelation in the series.

**Recommendations:**

* Explore alternative tests for autocorrelation, such as the **Durbin-Watson test**, to potentially uncover dependencies not captured by the Ljung-Box test.
* Consider employing additional visualizations, such as the **autocorrelation function (ACF)** and **partial autocorrelation function (PACF)**, to visually inspect the presence of autocorrelation at various lags.

The interpretation presented in the provided code snippet might be misleading due to the reliance on a specific, arbitrary threshold and limited exploration of alternative approaches. By following the recommendations and considering various perspectives, you can gain a more comprehensive understanding of the potential presence of autocorrelation in your data.

Hide

# 3.0.3.2. Autocorrelation Function (ACF) | Partial Autocorrelation Function (PACF)

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

acf(stock\_price) # ACF of JJ Series

A graph of a series of stock

Description automatically generated

Hide

pacf(stock\_price) # PACF of JJ Series

A line graph with numbers and text

Description automatically generated

Hide

acf(Maruti\_ds) # ACF of Maruti Difference (Stationary) Series

A line graph with numbers and a line

Description automatically generated

Hide

pacf(Maruti\_ds) # PACF of Maruti Difference (Stationary) Series

A graph with numbers and lines

Description automatically generated

Hide

# 3.1. Auto Regressive Integrated Moving Average (ARIMA) Models

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# 3.1.1. ARIMA Models

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# AR (p-Lag) Model : y(t) = c1 + a1\*y(t-1) + a2\*y(t-2) + ... + ap\*y(t-p) + e(t) where e = error == White Noise | AR-1 Model : y(t) = c + a1\*y(t-1) + e(t)

# MA (q-Lag) Model : y(t) = c2 + b1\*e(t-1) + b2\*e(t-2) + ... + bp\*e(t-p) where e = Error == White Noise | MA-1 Model : y(t) = d + b1\*e(t-1)

# ARMA (p, q) Model : y(t) = c + a1\*y(t-1) + ... + ap\*y(t-p) + b1\*e(t-1) + ... + bp\*e(t-p) + e(t) | ARMA (1, 1) Model : y(t) = c + a1\*y(t-1) + b1\*e(t-1) + e(t)

# ARIMA(p, d, q) = AR Order (p-Lags) | d-Degree of Differencing | MA Order (q-Lags)

# Note: The Degree of Differencing for a Time Series data such as Asset Returns is d=0. For a Time Series data such as Asset Prices the Degree of Differencing is usually d=1.

# Identify AR Order : PACF Cuts Off after p Lags | ACF Tails Off

# Identify MA Order : ACF Cuts Off after q Lags | PACF Tails Off

Hide

arma\_pq\_Maruti\_ds = auto.arima(Maruti\_ds); arma\_pq\_Maruti\_ds #p-lag=2, q-lag=2

Series: Maruti\_ds

ARIMA(2,0,2) with non-zero mean

Coefficients:

ar1 ar2 ma1 ma2 mean

-1.2683 -0.7865 1.3221 0.8361 7e-04

s.e. 0.1320 0.1508 0.1151 0.1355 5e-04

sigma^2 = 0.0002529: log likelihood = 3342.75

AIC=-6673.49 AICc=-6673.43 BIC=-6642.82

**ARIMA Model for Maruti Time Series: Interpretation Report**

This report interprets the fitted ARIMA(2, 0, 2) model for the Maruti\_ds time series, likely representing Maruti stock prices or related data.

**Model Summary:**

* **Model Type:** ARIMA(2, 0, 2)
  + AR (Autoregressive): The model incorporates the **lag effects** of the past **2 observations** (denoted by ar1 and ar2) on the current value.
  + I (Integrated): The model assumes no differencing is needed as the data is already stationary based on previous analysis (refer to the Augmented Dickey-Fuller test report).
  + MA (Moving Average): The model accounts for the **errors** from the past **2 observations** (denoted by ma1 and ma2) to improve the prediction accuracy.
* **Coefficients:**
  + The model estimates the coefficients for each of the AR and MA terms, along with the mean term.
  + The negative values of ar1 and ar2 indicate that past values have a **negative impact** on the current value, suggesting a **mean-reverting** pattern in the data.
  + The positive values of ma1 and ma2 imply that the model incorporates the **positive effects** of the past two errors to correct for previous forecasting errors.
* **Non-zero Mean:** The model incorporates a non-zero mean term to account for the constant level observed in the data.
* **Performance Measures:**
  + **Log Likelihood:** 3342.75 (higher values indicate better fit)
  + **AIC:** -6673.49 (lower values indicate better fit)
  + **AICc:** -6673.43 (adjusted AIC for smaller sample sizes)
  + **BIC:** -6642.82 (balances model complexity with goodness-of-fit)

**Interpretation:**

* The ARIMA(2, 0, 2) model suggests that the current value of the Maruti\_ds series is influenced by the **last two observations** (negative impact) and **errors from the last two forecasts** (positive impact).
* The negative AR coefficients indicate a **mean-reverting** tendency, meaning the series is likely to move back towards its average level after deviations.
* The positive MA coefficients imply that the model considers **past forecasting errors** to improve future predictions.
* The non-zero mean term accounts for the **constant level** observed in the data.

**Limitations and Next Steps:**

* The model assumes stationarity, which was confirmed using the ADF test. If the data exhibits non-stationarity, the model might not be reliable.
* The chosen ARIMA order (2, 0, 2) is based on statistical criteria or previous analysis. It's crucial to evaluate alternative models and compare their performance metrics (AIC, BIC, etc.) to select the best fit.
* Further analysis, such as residual plots and diagnostic tests, is recommended to assess the model's adequacy and identify potential areas for improvement.

This report provides a foundational understanding of the fitted ARIMA model for the Maruti\_ds series. By considering the limitations and exploring further diagnostics, you can refine and strengthen the model for more accurate forecasting.

Hide

Maruti\_ds\_fpq = forecast(arma\_pq\_Maruti\_ds, h = 500)

plot(Maruti\_ds\_fpq)

A graph of a sound wave

Description automatically generated

Hide

# Ljung-Box Test for Autocorrelation - Model Residuals

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

lb\_test\_arma\_pq\_Maruti\_ds = Box.test(arma\_pq\_Maruti\_ds$residuals); lb\_test\_arma\_pq\_Maruti\_ds

Box-Pierce test

data: arma\_pq\_Maruti\_ds$residuals

X-squared = 0.009284, df = 1, p-value = 0.9232

**Report: Ljung-Box Test for Autocorrelation of ARIMA Model Residuals**

**Introduction:**

This report examines the presence of autocorrelation in the **residuals** of the fitted ARIMA(2, 0, 2) model for the Maruti\_ds time series. We employ the Ljung-Box test to assess this characteristic.

**Methodology:**

1. **Model Fitting:**
   * The code snippet assumes an ARIMA(2, 0, 2) model has already been fitted to the Maruti\_ds data (refer to the previous report).
2. **Ljung-Box Test on Residuals:**
   * The Box.test(arma\_pq\_Maruti\_ds$residuals) function conducts the Ljung-Box test on the **residuals** from the fitted model.
   * This test aims to detect **autocorrelation** in the residuals, which indicates the model's **inadequacy** in capturing the underlying structure of the data.
3. **Result Interpretation:**
   * The output displays the following statistics:
     + **X-squared:** 0.009284 (test statistic)
     + **Degrees of freedom (df):** 1
     + **p-value:** 0.9232 (probability of observing such a statistic if no autocorrelation exists)

**Conclusion:**

* Considering the **high p-value (0.9232)**, we **fail to reject the null hypothesis** of **no autocorrelation** at even a **very low significance level** (e.g., 0.01).
* This suggests that the **residuals do not exhibit statistically significant autocorrelation**.

**Interpretation:**

* The absence of significant autocorrelation in the residuals indicates that the **ARIMA(2, 0, 2) model** likely **captures the essential structure** of the data and the **errors are independent**.
* This implies that the model's predictions are not systematically biased by past errors, potentially leading to **more reliable forecasts**.

**Limitations and Next Steps:**

* While the Ljung-Box test provides a general assessment of autocorrelation, it's crucial to **visually inspect** the **autocorrelation function (ACF)** and **partial autocorrelation function (PACF)** of the residuals to gain a deeper understanding of the error structure.
* If significant autocorrelation patterns are observed visually, it might be necessary to **re-evaluate the model** and explore alternative model specifications or transformations to capture the remaining dependencies in the data.

This report suggests that the ARIMA(2, 0, 2) model effectively addresses autocorrelation concerns based on the Ljung-Box test. However, further exploration through visual inspection of the ACF and PACF is recommended for a more comprehensive evaluation of the model's adequacy.

Hide

#p-value>alpha

Hide

# Test for Volatility Clustering or Heteroskedasticity: Box Test

Maruti\_ret\_sq = arma\_pq\_Maruti\_ds$residuals^2 # Residual Variance (Since Mean Returns is approx. 0)

plot(Maruti\_ret\_sq)

A graph of a graph

Description automatically generated

Hide

Maruti\_ret\_sq\_box\_test = Box.test(Maruti\_ret\_sq, lag = 2) # H0: Return Variance Series is Not Serially Correlated

Maruti\_ret\_sq\_box\_test # Inference : Return Variance Series is Autocorrelated (Has Volatility Clustering)

Box-Pierce test

data: Maruti\_ret\_sq

X-squared = 28.549, df = 2, p-value = 6.319e-07

Hide

# Test for Volatility Clustering or Heteroskedasticity: ARCH Test

Maruti\_ret\_arch\_test = ArchTest(arma\_pq\_Maruti\_ds$residuals^2, lags = 2) # H0: No ARCH Effects

Maruti\_ret\_arch\_test # Inference : Return Series is Heteroskedastic (Has Volatility Clustering)

ARCH LM-test; Null hypothesis: no ARCH effects

data: arma\_pq\_Maruti\_ds$residuals^2

Chi-squared = 1.6338, df = 2, p-value = 0.4418

**Report: ARCH Test for Volatility Clustering in ARIMA Model Residuals**

**Introduction:**

This report examines the presence of **volatility clustering**, also known as **heteroskedasticity**, in the residuals of the fitted ARIMA(2, 0, 2) model for the Maruti\_ds time series. We employ the Engle's ARCH (Autoregressive Conditional Heteroskedasticity) test to assess this characteristic.

**Methodology:**

1. **Model Fitting:**
   * The code snippet assumes an ARIMA(2, 0, 2) model has already been fitted to the Maruti\_ds data (refer to previous reports).
2. **ARCH Test on Squared Residuals:**
   * The ArchTest(arma\_pq\_Maruti\_ds$residuals^2, lags = 2) function performs the ARCH test on the **squared** residuals from the fitted model.
   * This test aims to detect whether the **variance of the residuals** depends on the **magnitude of past squared residuals**.
   * The null hypothesis (H0) states that **no ARCH effects** (volatility clustering) are present.
3. **Result Interpretation:**
   * The output displays the following statistics:
     + **Chi-squared:** 1.6338 (test statistic)
     + **Degrees of freedom (df):** 2
     + **p-value:** 0.4418 (probability of observing such a statistic under the null hypothesis)

**Conclusion:**

* Considering the **high p-value (0.4418)**, we **fail to reject the null hypothesis** of **no ARCH effects** at a common significance level (e.g., 0.05).
* This suggests that the **squared residuals do not exhibit statistically significant dependence**, implying that the **variance of the errors is not conditional** on the magnitude of past errors.

**Interpretation:**

* The absence of significant ARCH effects indicates that the **ARIMA(2, 0, 2) model** likely **captures the volatility structure** of the data reasonably well.
* This implies that the **model's predictions** are not systematically affected by periods of high or low volatility, potentially leading to **more accurate forecasts** of both calm and volatile periods.

**Limitations and Next Steps:**

* While the ARCH test provides a general assessment of heteroskedasticity, it's crucial to **visually inspect** the **heteroskedasticity test plot** or the **squared residuals time series plot** to gain further insights into the volatility patterns.
* If visual inspection reveals evidence of volatility clustering, exploring **GARCH (Generalized ARCH)** models or other volatility modeling techniques might be necessary to capture the time-varying nature of the variance.

This report suggests that the ARIMA(2, 0, 2) model effectively addresses volatility clustering concerns based on the ARCH test. However, further visual inspection of the residuals and potentially exploring alternative models are recommended for a more comprehensive evaluation of the model's ability to capture the volatility structure of the data.

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))

Maruti\_ret\_garch1 = ugarchfit(garch\_model1, data = arma\_pq\_Maruti\_ds$residuals^2); Maruti\_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 0.000206 0.000027 7.696793 0.000000

omega 0.000000 0.000001 0.015873 0.987336

alpha1 0.060012 0.016838 3.564134 0.000365

beta1 0.910090 0.007764 117.212650 0.000000

Robust Standard Errors:

Estimate Std. Error t value Pr(>|t|)

mu 0.000206 0.155328 0.001324 0.99894

omega 0.000000 0.005789 0.000002 1.00000

alpha1 0.060012 101.342895 0.000592 0.99953

beta1 0.910090 35.820646 0.025407 0.97973

LogLikelihood : 7480.751

Information Criteria

------------------------------------

Akaike -12.187

Bayes -12.170

Shibata -12.187

Hannan-Quinn -12.181

Weighted Ljung-Box Test on Standardized Residuals

------------------------------------

statistic p-value

Lag[1] 12.47 4.142e-04

Lag[2\*(p+q)+(p+q)-1][2] 13.17 2.784e-04

Lag[4\*(p+q)+(p+q)-1][5] 21.29 8.548e-06

d.o.f=0

H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals

------------------------------------

statistic p-value

Lag[1] 0.002894 0.9571

Lag[2\*(p+q)+(p+q)-1][5] 0.192883 0.9931

Lag[4\*(p+q)+(p+q)-1][9] 0.312879 0.9997

d.o.f=2

Weighted ARCH LM Tests

------------------------------------

Statistic Shape Scale P-Value

ARCH Lag[3] 0.1507 0.500 2.000 0.6979

ARCH Lag[5] 0.2242 1.440 1.667 0.9597

ARCH Lag[7] 0.2685 2.315 1.543 0.9943

Nyblom stability test

------------------------------------

Joint Statistic: 38.9935

Individual Statistics:

mu 0.1593

omega 12.3928

alpha1 0.1110

beta1 0.1251

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**  <dbl> | **sig**  <chr> |
| --- | --- | --- | --- |
| Sign Bias | 0.02378317 | 0.9810294 |  |
| Negative Sign Bias | 1.00986438 | 0.3127602 |  |
| Positive Sign Bias | 0.28808549 | 0.7733302 |  |
| Joint Effect | 3.49377321 | 0.3215706 |  |

4 rows

Adjusted Pearson Goodness-of-Fit Test:

------------------------------------

group statistic p-value(g-1)

1 20 2621 0

2 30 2610 0

3 40 2668 0

4 50 2673 0

Elapsed time : 0.1124659

**Report: GARCH Model Fitting and Parameter Significance**

**Introduction:**

This report examines the fitted GARCH(1,1) model (Generalized Autoregressive Conditional Heteroskedasticity) for the squared residuals of the ARIMA(2, 0, 2) model applied to the Maruti\_ds time series. The report focuses on the t-values and p-values of the estimated coefficients to assess their significance.

**Model Specification:**

* **GARCH(1,1):** This model assumes the conditional variance of the residuals depends on a weighted average of one past squared residual (GARCH term) and one past innovation term (ARCH term).
* **Mean Model:** The model incorporates a constant mean to account for the average level in the squared residuals.

**Results:**

* The code snippet provides the t-values and p-values for three parameters:
  + **Sign Bias:** This parameter is not relevant to the GARCH model and likely represents an artifact of the software being used.
  + **Positive Sign Bias:** This parameter is also not relevant to the GARCH model and can be ignored.
  + **Joint Effect:** This parameter represents the combined significance of the GARCH and ARCH terms.
* **T-values:** The t-value indicates the number of standard deviations an estimate is away from zero. Higher absolute t-values suggest stronger evidence against the null hypothesis (coefficient is zero).
* **P-values:** The p-value represents the probability of observing a t-value as extreme as the calculated value, assuming the null hypothesis is true. Lower p-values indicate stronger evidence against the null hypothesis.

**Interpretation:**

* Since the t-values and p-values for the **Sign Bias** and **Positive Sign Bias** parameters are not relevant to the GARCH model, we **cannot interpret them** meaningfully.
* The **Joint Effect** parameter has a **t-value of 3.49** and a **p-value of 0.32**. Based on a common significance level of 0.05, we **fail to reject the null hypothesis**. This suggests that the **combined effect of the GARCH and ARCH terms might not be statistically significant**.

**Limitations and Next Steps:**

* While the p-value provides some guidance on significance, it's crucial to consider the **overall model performance** by evaluating metrics like AIC, BIC, and residual diagnostics.
* If further analysis indicates the need for a GARCH model, exploring alternative specifications (e.g., different GARCH orders) might be necessary to capture the volatility dynamics more effectively.

**Conclusion:**

The provided information is limited to the t-values and p-values of specific parameters. A comprehensive evaluation requires considering the entire model output, performance measures, and potential alternative specifications.

Hide

# Test for Volatility Clustering or Heteroskedasticity: ARCH Test

Maruti\_garch\_arch\_test = ArchTest(residuals(Maruti\_ret\_garch1)^2, lags = 1) # H0: No ARCH Effects

Maruti\_garch\_arch\_test # Inference : Return Series is Heteroskedastic (Has Volatility Clustering)

#Maruti\_ret\_garch1

```

ARCH LM-test; Null hypothesis: no ARCH effects

data: residuals(Maruti\_ret\_garch1)^2

Chi-squared = 0.00060816, df = 1, p-value = 0.9803

**Report: ARCH Test on GARCH Model Residuals**

**Introduction:**

This report examines the presence of **volatility clustering**, also known as **heteroskedasticity**, in the residuals of the fitted **GARCH(1,1)** model for the Maruti\_ds time series. We employ the Engle's ARCH test to assess this characteristic.

**Methodology:**

1. **GARCH Model Fitting:**
   * The code snippet assumes a GARCH(1,1) model has already been fitted to the squared residuals of the ARIMA(2, 0, 2) model (refer to previous report).
2. **ARCH Test on GARCH Model Residuals:**
   * The ArchTest(residuals(Maruti\_ret\_garch1)^2, lags = 1) function performs the ARCH test on the **squared residuals** from the **GARCH model**.
   * This test aims to detect whether the **variance of the GARCH model residuals** depends on the **magnitude of past squared residuals**.
   * The null hypothesis (H0) states that **no ARCH effects** (volatility clustering) are present in the GARCH model residuals.
3. **Result Interpretation:**
   * The output displays the following statistics:
     + **Chi-squared:** 0.00060816 (test statistic)
     + **Degrees of freedom (df):** 1
     + **p-value:** 0.9803 (probability of observing such a statistic under the null hypothesis)

**Conclusion:**

* Considering the **extremely high p-value (0.9803)**, we **fail to reject the null hypothesis** of **no ARCH effects** at a very high significance level (e.g., 0.01).
* This suggests that the **squared residuals of the GARCH model do not exhibit statistically significant dependence**, implying that the **variance of the GARCH model residuals is not conditional** on the magnitude of past errors.

**Interpretation:**

* The absence of significant ARCH effects in the GARCH model residuals indicates that the **GARCH(1,1) model** likely **captures the volatility structure** of the data effectively.
* This implies that the **model's predictions** are not systematically affected by periods of high or low volatility, potentially leading to **more accurate forecasts** of both calm and volatile periods.

**Limitations and Next Steps:**

* While the ARCH test provides a general assessment of heteroskedasticity, it's crucial to **visually inspect** the **heteroskedasticity test plot** or the **squared residuals time series plot** from the GARCH model to gain further insights into the volatility patterns.
* If visual inspection reveals evidence of remaining volatility clustering, exploring **alternative GARCH model specifications** with higher orders or different distributions for the error term might be necessary to capture the time-varying nature of the variance more accurately.

This report suggests that the GARCH(1,1) model effectively addresses volatility clustering concerns based on the ARCH test. However, further visual inspection of the residuals and potentially exploring alternative GARCH specifications are recommended for a comprehensive evaluation of the model's ability to capture the volatility structure of the data.

Hide

garch\_model2 = ugarchspec(variance.model = list(model = 'sGARCH', garchOrder = c(1,1)), mean.model = list(armaOrder = c(2,2), include.mean = FALSE))

Maruti\_ret\_garch2 = ugarchfit(garch\_model2, data = Maruti\_ds); Maruti\_ret\_garch2

\*---------------------------------\*

\* GARCH Model Fit \*

\*---------------------------------\*

Conditional Variance Dynamics

-----------------------------------

GARCH Model : sGARCH(1,1)

Mean Model : ARFIMA(2,0,2)

Distribution : norm

Optimal Parameters

------------------------------------

Estimate Std. Error t value Pr(>|t|)

ar1 -1.196937 0.119627 -10.0056 0.000000

ar2 -0.782179 0.121065 -6.4608 0.000000

ma1 1.270980 0.099840 12.7301 0.000000

ma2 0.832159 0.109993 7.5656 0.000000

omega 0.000048 0.000012 3.8690 0.000109

alpha1 0.188917 0.043031 4.3903 0.000011

beta1 0.630142 0.075085 8.3924 0.000000

Robust Standard Errors:

Estimate Std. Error t value Pr(>|t|)

ar1 -1.196937 0.163585 -7.3169 0.000000

ar2 -0.782179 0.174082 -4.4932 0.000007

ma1 1.270980 0.133467 9.5228 0.000000

ma2 0.832159 0.160917 5.1714 0.000000

omega 0.000048 0.000018 2.7077 0.006776

alpha1 0.188917 0.058420 3.2338 0.001222

beta1 0.630142 0.105229 5.9883 0.000000

LogLikelihood : 3391.4

Information Criteria

------------------------------------

Akaike -5.5165

Bayes -5.4874

Shibata -5.5166

Hannan-Quinn -5.5056

Weighted Ljung-Box Test on Standardized Residuals

------------------------------------

statistic p-value

Lag[1] 0.3033 0.5818

Lag[2\*(p+q)+(p+q)-1][11] 3.1609 1.0000

Lag[4\*(p+q)+(p+q)-1][19] 5.4766 0.9865

d.o.f=4

H0 : No serial correlation

Weighted Ljung-Box Test on Standardized Squared Residuals

------------------------------------

statistic p-value

Lag[1] 0.03636 0.8488

Lag[2\*(p+q)+(p+q)-1][5] 1.66066 0.7004

Lag[4\*(p+q)+(p+q)-1][9] 2.31527 0.8642

d.o.f=2

Weighted ARCH LM Tests

------------------------------------

Statistic Shape Scale P-Value

ARCH Lag[3] 0.004582 0.500 2.000 0.9460

ARCH Lag[5] 0.117135 1.440 1.667 0.9836

ARCH Lag[7] 0.576864 2.315 1.543 0.9709

Nyblom stability test

------------------------------------

Joint Statistic: 0.958

Individual Statistics:

ar1 0.1287

ar2 0.1546

ma1 0.1526

ma2 0.1490

omega 0.3052

alpha1 0.2924

beta1 0.3669

Asymptotic Critical Values (10% 5% 1%)

Joint Statistic: 1.69 1.9 2.35

Individual Statistic: 0.35 0.47 0.75

Sign Bias Test

------------------------------------

|  |
| --- |
|  |

|  | **t-value**  <dbl> | **prob**  <dbl> | **sig**  <chr> |
| --- | --- | --- | --- |
| Sign Bias | 0.09897639 | 0.9211732 |  |
| Negative Sign Bias | 1.02219684 | 0.3068901 |  |
| Positive Sign Bias | 0.68392026 | 0.4941552 |  |
| Joint Effect | 2.76701230 | 0.4289591 |  |

4 rows

Adjusted Pearson Goodness-of-Fit Test:

------------------------------------

group statistic p-value(g-1)

1 20 52.01 6.603e-05

2 30 72.00 1.608e-05

3 40 85.70 2.358e-05

4 50 81.60 2.381e-03

Elapsed time : 0.2148659

Hide

# GARCH Forecast

Maruti\_ret\_garch\_forecast1 = ugarchforecast(Maruti\_ret\_garch1, n.ahead = 500); Maruti\_ret\_garch\_forecast1

\*------------------------------------\*

\* GARCH Model Forecast \*

\*------------------------------------\*

Model: sGARCH

Horizon: 500

Roll Steps: 0

Out of Sample: 0

0-roll forecast [T0=1227-01-01]:

Series Sigma

T+1 0.0002056 0.0004320

T+2 0.0002056 0.0004409

T+3 0.0002056 0.0004493

T+4 0.0002056 0.0004573

T+5 0.0002056 0.0004650

T+6 0.0002056 0.0004723

T+7 0.0002056 0.0004793

T+8 0.0002056 0.0004859

T+9 0.0002056 0.0004923

T+10 0.0002056 0.0004985

Hide

Maruti\_ret\_garch\_forecast2 = ugarchforecast(Maruti\_ret\_garch2, n.ahead = 500); Maruti\_ret\_garch\_forecast2

\*------------------------------------\*

\* GARCH Model Forecast \*

\*------------------------------------\*

Model: sGARCH

Horizon: 500

Roll Steps: 0

Out of Sample: 0

0-roll forecast [T0=2019-12-30]:

Series Sigma

T+1 -1.256e-04 0.01359

T+2 -3.073e-04 0.01411

T+3 4.660e-04 0.01453

T+4 -3.174e-04 0.01487

T+5 1.547e-05 0.01514

T+6 2.298e-04 0.01535

T+7 -2.871e-04 0.01553

T+8 1.640e-04 0.01567

T+9 2.835e-05 0.01578

T+10 -1.622e-04 0.01588

Hide

plot(Maruti\_ret\_garch\_forecast2)

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

1

A graph of a graph showing the time and time

Description automatically generated with medium confidence

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

A graph of a graph showing the time and time

Description automatically generated with medium confidence

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

4

Error in .plot.garchforecast.4(x, n.roll) :

n.roll less than 5!...does not make sense to provide this plot.

A graph of a graph

Description automatically generated