# TIME SERIES MODELLING

**LOADING REQUIRED PACKAGES**

library(readr)

library(tseries)

library(forecast)

library(timeSeries)

library(ggplot2)

library(trend)

**IMPORTING DATASET**

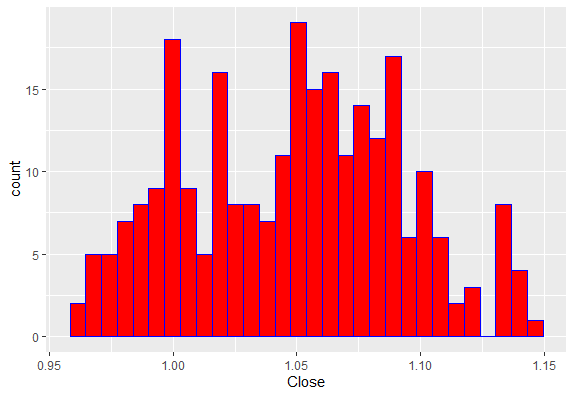
EURUSD\_X <- read\_csv("C:/Users/User/Downloads/EURUSD=X.csv")

View(EURUSD\_X)

EURUSD\_X <- EURUSD\_X[,-7]

**GRAPH OF THE ORIGINAL DATASET(TESTING FOR NORMALITY)**

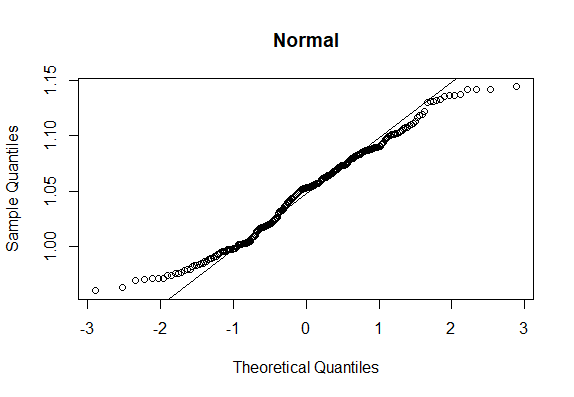
**Code:** ggplot(EURUSD\_X, aes(Close)) + geom\_histogram(color="Blue", fill="red")



**Fig 1:** Showing the original dataset

**Observations:** The datasets mostly fall in the middle of the distribution(U-Shaped), it is normal.

**Code:** qqline(EURUSD\_X$Close)



**Fig 2:** Showing normal probability plot

**Observations:** Most datasets fall within the line of best fit, it is normal.

**TESTING FOR STATIONARITY USING ADF & ACF**

**Code:** adf.test(EURUSD\_X$Close, alternative = c("stationary"))

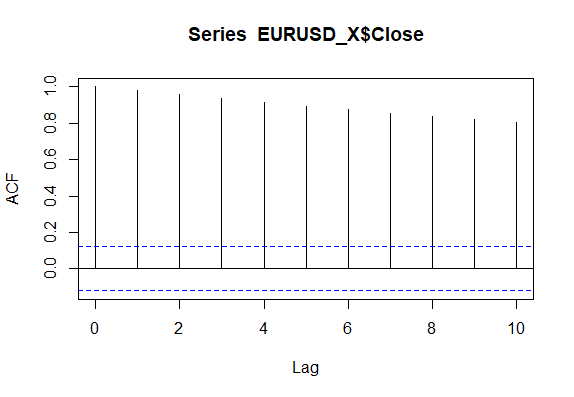
**Augmented Dickey-Fuller Test**

data: EURUSD\_X$Close

Dickey-Fuller = -1.1512, Lag order = 6, p-value = 0.9128

alternative hypothesis: stationary

**Interpretation:** From the result of Augmented Dickey-Fuller Test, p-value(0.9128) is greater than 0.05. The data is non-stationary



**Fig 3:** Showing the correlogram of the dataset

**Interpreting:** The lags are decaying slowly, it is non-stationary. I'll then difference the series.

**Differencing the series**

ndiffs(EURUSD\_X$Close) #it gives me 1, which I put in diff code below

new\_series <- diff(EURUSD\_X$Close, 1) #Stationary series

* **New test for stationarity using adf & acf after differencing**

**Code:** adf.test(new\_series, alternative = c("stationary"))

**Augmented Dickey-Fuller Test**

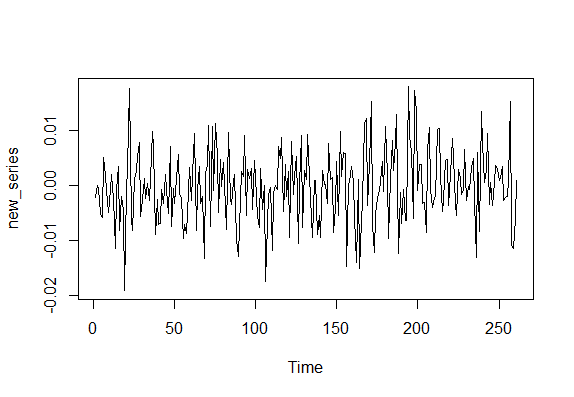
data: new\_series

Dickey-Fuller = -7.509, Lag

order = 6, p-value = 0.01

alternative hypothesis: stationary

**Interpretation:**From the result of Augmented Dickey-Fuller Test, p-value(0.01) is lessr than 0.05. The data is stationary



**Fig 4**: Showing the plot of the new data(that is, differenced)

**OBTAIN TRENDS AND PLOTS OF THE TRENDS**

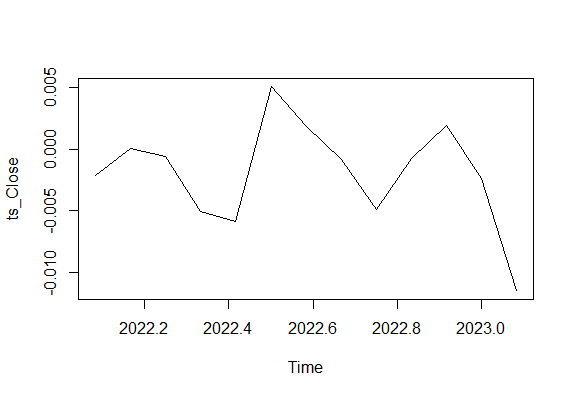
To obtain trends, I'll convert the data to a time series data

**Code:** ts\_Close <- ts(new\_series, start = c(2022,2), end = c(2023, 2), frequency = 12)

ts\_Close #Trend

|  |  |  |
| --- | --- | --- |
| **Year** | **Month** | **Trend values** |
| 2022 | February | -0.00217 |
| 2022 | March | 0.000026 |
| 2022 | April | -0.000599 |
| 2022 | May | -0.005047 |
| 2022 | June | -0.005873 |
| 2022 | July | 0.005047 |
| 2022 | August | 0.001847 |
| 2022 | September | -0.000879 |
| 2022 | October | -0.004927 |
| 2022 | November | -0.000755 |
| 2022 | December | 0.001947 |
| 2023 | January | -0.002344 |
| 2023 | February | -0.0115 |

**Table 1:** Showing the trends of the time series data



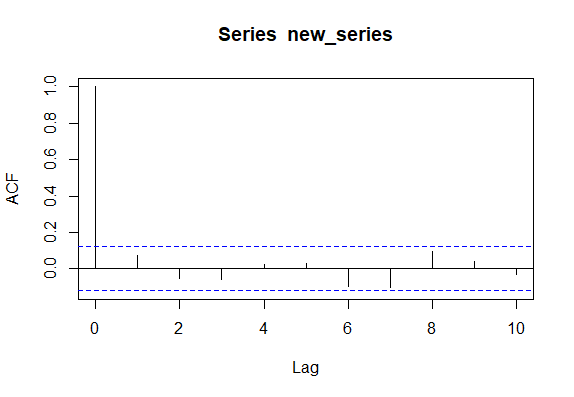
**Fig 5:** Showing the plot of the trend of the time series data

**ACF and PACF**

Autocorrelations of series ‘new\_series’, by lag

|  |  |
| --- | --- |
| lags | Values |
| 0 | 1.000 |
| 1 | 0.071 |
| 2 | -0.055 |
| 3 | -0.056 |
| 4 | 0.026 |
| 5 | 0.029 |
| 6 | -0.095 |
| 7 | -0.104 |
| 8 | 0.095 |
| 9 | 0.038 |
| 10 | -0.033 |

**Table 2:** Showing the ACF of the first ten lags

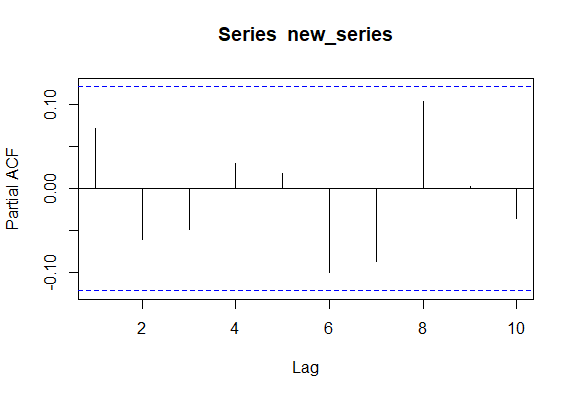
**Observation:** The correlogram tappers off quickly, which means it is stationary.

**Fig 6:** Showing the correlogram of the time series data of the first ten lags

Partial autocorrelations of series ‘new\_series’, by lag

|  |  |
| --- | --- |
| lags | Values |
| 1 | 0.071 |
| 2 | -0.061 |
| 3 | -0.048 |
| 4 | 0.031 |
| 5 | 0.019 |
| 6 | -0.099 |
| 7 | -0.086 |
| 8 | 0.103 |
| 9 | 0.003 |
| 10 | -0.036 |

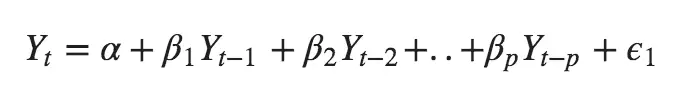
**Table 3:** Showing the PACF(Partial auto correlation function) of the first ten la



**Fig 7:** Showing the PACF of the time series data of the first ten lags

**MODELLING**

AR Model



AR(1)

**Code:** ar(ts\_Close, aic = F, order.max = 1)

Coefficients:

1

0.0633

Order selected 1 sigma^2 estimated as 1.921e-05

**Expression**: Yt = ϕ1 yt-1 + ϵ t

Yt = 0.0633Yt-1 + 4.3829e-03

AR(2)

**Code:** ar(ts\_Close, aic = F, order.max = 2)

Coefficients:

1 2

0.0936 -0.4782

Order selected 2 sigma^2 estimated as 1.63e-05

**Expression**:Yt = ϕ1 yt-1 + ϕ2 yt-2+ ϵ t

Yt = 0.0936Yt-1 + -0.4782Yt-2 + 4.0373e-03

AR(3)

**Code:** ar(ts\_Close, aic = F, order.max = 3)

ar(x = ts\_Close, aic = F, order.max = 3)

Coefficients:

1 2 3

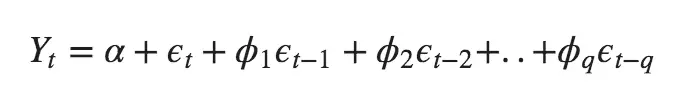
0.0344 -0.4666 -0.1238

Order selected 3 sigma^2 estimated as 1.783e-05

**Expression**: Yt = ϕ1 yt-1 + ϕ2 yt-2 + ϕ2 yt-3 + ϵ t

Yt = 0.0344Yt-1 + -0.4666Yt-2 + -0.1238Yt-3 + 4.2226e-03

MA Model



**Code:**

MA(1): ma(ts\_Close, order = 1, centre = T)

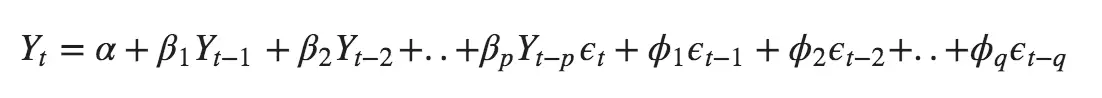
MA(2): ma(ts\_Close, order = 2, centre = T)

MA(3): ma(ts\_Close, order = 3, centre = T)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Year** | **Month** | **MA(1)** | **MA(2)** | **MA(3)** |
| 2022 | February | -0.00217 | - | - |
| 2022 | March | 0.000026 | -0.00067925 | -0.0009143333 |
| 2022 | April | -0.000599 | -0.00155475 | -0.0018733333 |
| 2022 | May | -0.005047 | -0.0041415 | -0.0039656667 |
| 2022 | June | -0.005873 | -0.0029365 | -0.0019576667 |
| 2022 | July | 0.005047 | 0.001517 | 0.0003403333 |
| 2022 | August | 0.001847 | 0.0019655 | 0.002005 |
| 2022 | September | -0.000879 | -0.0012095 | -0.0013196667 |
| 2022 | October | -0.004927 | -0.002872 | -0.00212187 |
| 2022 | November | -0.000755 | -0.0011225 | -0.001245 |
| 2022 | December | 0.001947 | 0.00019875 | -0.000384 |
| 2023 | January | -0.002344 | -0.00356025 | -0.0039656667 |
| 2023 | February | -0.0115 | - | - |

**Table 4:** Showing the moving averages at order 1, order 2 and order 3 respectively

ARMA Model

ARMA(1,1)

**Code:** arma(ts\_Close, order = c(1,1))

Residuals:

Min 1Q Median 3Q Max

-0.0088440 -0.0026922 0.0006271 0.0022288 0.0079491

Coefficient(s):

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

ar1 -0.135911 0.505259 -0.269 0.7879

ma1 0.534976 0.321528 1.664 0.0961 .

intercept -0.002338 0.001771 -1.320 0.1869

---

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

Fit:

sigma^2 estimated as 1.753e-05, Conditional Sum-of-Squares = 0, AIC = -99.48

**Expression**:Yt = c + ϕ1 yt-1+ θ1 ϵ t-1 + ϵ t

Yt = -0.002338 - 0.135911Yt-1 + 0.534976εt-1 + 4.1869e-03

ARMA(1,2)

**Code:** arma(ts\_Close, order = c(1,2))

Residuals:

Min 1Q Median 3Q Max

-0.0031928 -0.0018972 0.0002489 0.0013273 0.0037206

Coefficient(s):

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

ar1 1.5909865 NA NA NA

ma1 -2.4475635 NA NA NA

ma2 -1.7444564 NA NA NA

intercept -0.0008892 NA NA NA

Fit:

sigma^2 estimated as 4.711e-06, Conditional Sum-of-Squares = 0, AIC = -114.56

**Expression**: Yt = c + ϕ1 yt-1 + θ1 ϵ t-1 + + θ2 ϵ t-2 + ϵ t

Yt = -0.0008892 + 1.5909865Yt-1 - 2.4475635εt-1 - 1.7444564εt-2 + 2.1705e-03

ARMA(2,2)

**Code:** arma(ts\_Close, order = c(2,2))

Residuals:

Min 1Q Median 3Q Max

-0.0021103 -0.0011494 -0.0007849 0.0001478 0.0033085

Coefficient(s):

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

ar1 -0.605884 0.038323 -15.810 <2e-16 \*\*\*

ar2 -1.443356 0.069810 -20.675 <2e-16 \*\*\*

ma1 1.437691 0.112623 12.766 <2e-16 \*\*\*

ma2 3.020410 0.091364 33.059 <2e-16 \*\*\*

intercept -0.002574 0.000289 -8.908 <2e-16 \*\*\*

---

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

Fit:

sigma^2 estimated as 2.505e-06, Conditional Sum-of-Squares = 0, AIC = -120.77

**Expression**: Yt = c + ϕ1 yt-1 + ϕ2 yt-2 + θ1 ϵ t-1 + + θ2 ϵ t-2 + ϵ t

Yt = -0.002574 - 0.605884Yt-1 -1.443356Yt-2 + 1.437691εt-1 + 3.02041εt-2 + 1.5827e-03

ARMA(2,3)

**Code:** arma(ts\_Close, order = c(2,3))

Residuals:

Min 1Q Median 3Q Max

-0.0012139 -0.0002374 0.0006031 0.0009727 0.0028767

Coefficient(s):

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

ar1 0.1002860 0.0675309 1.485 0.138

ar2 -1.3225818 0.2505848 -5.278 1.31e-07 \*\*\*

ma1 1.0075685 0.1023619 9.843 < 2e-16 \*\*\*

ma2 3.3093106 0.5657793 5.849 4.94e-09 \*\*\*

ma3 -0.1817324 0.7903336 -0.230 0.818

intercept -0.0059311 0.0003585 -16.546 < 2e-16 \*\*\*

---

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

Fit:

sigma^2 estimated as 1.298e-06, Conditional Sum-of-Squares = 0, AIC = -127.32

**Expression**: Yt = c + ϕ1 yt-1 + ϕ2 yt-2 + θ1 ϵ t-1 + + θ2 ϵ t-2 + θ3 ϵ t-3 + ϵ t

Yt = -0.0059311 + 0.100286Yt-1 -1.3225818Yt-2 + 1.0075685εt-1 + 3.3093106εt-2 -0.1817324εt-3 + 1.1393e-03

ARIMA Model

ARIMA(1,1,1)

**Code:** Arima(ts\_Close,order = c(1,1,1))

Coefficients:

ar1 ma1

-0.3011 0.4362

s.e. 0.7373 0.6399

sigma^2 = 3e-05: log likelihood = 46.54

AIC=-87.08 AICc=-84.08 BIC=-85.63

Training set error measures:

ME RMSE MAE MPE MAPE

Training set -0.0007035502 0.004803578 0.003722683 658.8333 794.3384

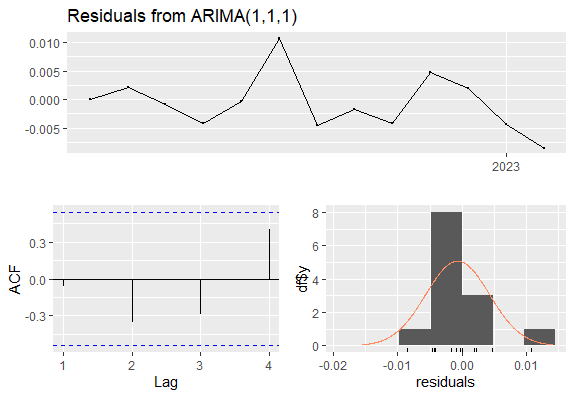
MASE ACF1

Training set 0.3990014 -0.06283331

**Expression**: Yt = ϕ1 yt-1 + θ1 ϵ t-1 + ϵ t

Yt = -0.3011Yt-1 + 0.4362εt-1 + 5.4772e-03

checkresiduals(arima\_1\_1\_1)



**Figure 8:** Showing the residuals of ARIMA(1,1,1)

ARIMA(2,1,2)

**Code:** Arima(ts\_Close,order = c(2,1,2))

Coefficients:

ar1 ar2 ma1 ma2

0.3139 -0.9492 -0.8434 1.0000

s.e. 0.1250 0.0693 0.2945 0.3776

sigma^2 = 1.013e-05: log likelihood = 51.48

AIC=-92.96 AICc=-82.96 BIC=-90.54

Training set error measures:

ME RMSE MAE MPE MAPE MASE

Training set -0.00047646 0.002497153 0.001939881 342.1489 406.7378 0.2079187

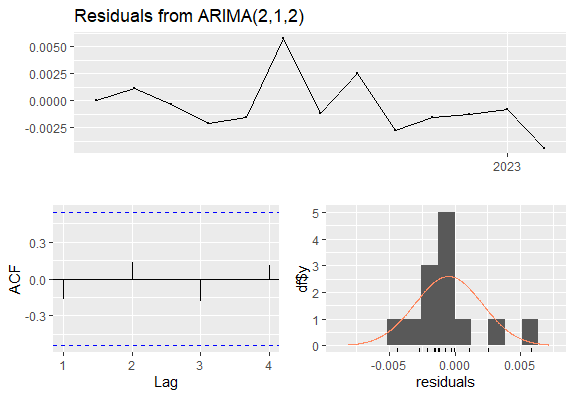
ACF1

Training set -0.1646571

**Expression**: Yt = ϕ1 yt-1 + ϕ2 yt-2 + θ1 ϵ t-1 + + θ2 ϵ t-2 + ϵ t

Yt = 0.3139Yt-1 -0.9492Yt-2 + -0.8434εt-1 + 1.0εt-2 + 3.1828e-03

checkresiduals(arima\_2\_1\_2)



**Figure 9:** Showing the residuals of ARIMA(2,1,2)

**MODEL SELECTION USING AIC and BIC**

**Code:** AIC(arima\_1\_1\_1)

Result:-87.08195

**Code:** AIC(arima\_2\_1\_2)

Result**:** -92.96109

**Code:** BIC(arima\_1\_1\_1)

Result: -85.62723

**Code:** BIC(arima\_1\_1\_1)

Result: -85.62723

**Interpretation:** The two models have the same BIC(-85.62723), but ARIMA(2,1,2) has AIC=-92.96109 which is lesser than ARIMA(1,1,1) AIC=-87.08195. I will therefore conclude that ARIMA(1,1,1) is not a good fit. ARIMA(2,1,2) is the best model for the data because it has the least AIC.

**ADEQUACY OF MODEL**

I will test the adequacy of the model using LjungBox test

**Hypothesis:**

H0: Data values are independent i.e time series is not auto correlated(auto correlations are all zero)

H1: Data values are dependent i.e time series are auto correlated(auto correlations are all different zero)

**Code:** Box.test(ts\_Close,lag = 10, type = "Ljung")

Box-Ljung test

data: ts\_Close

X-squared = 18.808, df = 10, p-value = 0.04278

**Interpretation:** Since the p-value=0.04278<0.05, there is significant different from H0. There are auto correlations in the data.

**TESTING THE SIGNIFICANCE OF PARAMTER OF ARIMA MODEL**

Test of significance pf paramter of ARIMA model is also used to know the best model to work with. It is an alternative to AIC and BIC.

**Hypothesis:**

H0: Estimates of the parameters are not significantly different from zero(0)

H1: Estimates of the parameters are significantly different from zero(0)

**Code:** coeftest(arima\_1\_1\_1)

**Result:** z test of coefficients:

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

ar1 -0.30112 0.73725 -0.4084 0.6830

ma1 0.43619 0.63987 0.6817 0.4954

**Interpretation:** It affirms that ARIMA(1,1,1) is not the best model, the parameters are all greater than 0.05 and are not significantly different from zero(0)

**Code:** coeftest(arima\_2\_1\_2)

**Result:** z test of coefficients:

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

ar1 0.313925 0.125007 2.5113 0.012030 \*

ar2 -0.949210 0.069258 -13.7055 < 2.2e-16 \*\*\*

ma1 -0.843363 0.294473 -2.8640 0.004184 \*\*

ma2 1.000000 0.377616 2.6482 0.008092 \*\*

---

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

**Interpretation:** It affirms that ARIMA(2,1,2) is the best model, the parameters are all lesser than 0.05 and are significantly different from zero(0)

**FORECASTING**

Aim is to forecast using the best model from the dataset and predict for the next four months.

**Code:** forecast(arima\_2\_1\_2, h = 4, level = 95)

**Result:**  Point Forecast Lo 95 Hi 95

Mar 2023 -0.007270241 -0.014012503 -0.0005279785

Apr 202 -0.001325902 -0.008947166 0.0062953618

May 2023 -0.003474756 -0.011263773 0.0043142620

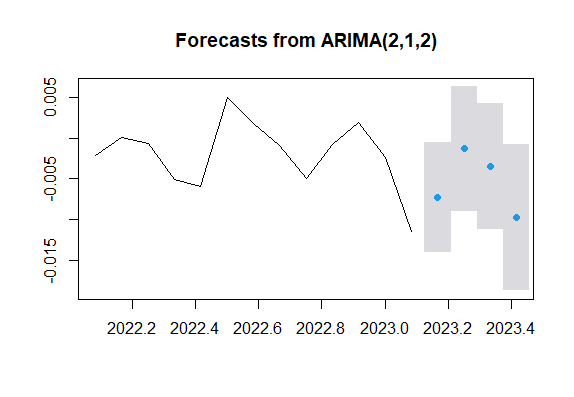
Jun 2023 -0.009791759 -0.018749178 -0.0008343391

**Interpretation:** Recall from Table 1, the chocolate color indicate the forecast values

|  |  |  |
| --- | --- | --- |
| **Year** | **Month** | **Trend values** |
| 2022 | February | -0.00217 |
| 2022 | March | 0.000026 |
| 2022 | April | -0.000599 |
| 2022 | May | -0.005047 |
| 2022 | June | -0.005873 |
| 2022 | July | 0.005047 |
| 2022 | August | 0.001847 |
| 2022 | September | -0.000879 |
| 2022 | October | -0.004927 |
| 2022 | November | -0.000755 |
| 2022 | December | 0.001947 |
| 2023 | January | -0.002344 |
| 2023 | February | -0.0115 |
| 2023 | March | -0.007270241 |
| 2023 | April | -0.001325902 |
| 2023 | May | -0.003474756 |
| 2023 | June | -0.009791759 |

**Table 5:** Showing the trend forecast of the best model.

**Code:** plot(forecast\_Close)



**Interpretation:** The blue dots are the forecast values. It shows a downward trend towards the end.