**Code for ARIMA:**

# Reinstall the required packages

install.packages("fpp2")

install.packages("forecast")

install.packages("fma")

install.packages("expsmooth")

# Reinstall the ggplot2 package

install.packages("ggplot2")

# Load the library again

library(ggplot2)

# Load the libraries again

library(ggplot2)

library(fpp2)

library(tseries)

library(forecast)

library(zoo)

library(lmtest)

# ARIMA lecture 8

# Set working directory (ensure this is the directory containing the CSV file)

setwd("C:/Users/ASUS/Downloads")

# Read the data from the CSV file

data <- read.csv("Inflation.csv")

# Create a time series object

ts\_data <- ts(data$Inflation.Rate, start = 1991, frequency = 1)

# Load required libraries

library(ggplot2)

library(fpp2)

library(tseries)

library(forecast)

library(zoo)

library(lmtest)

# Display the time series data

ggtsdisplay(ts\_data)

autoplot(ts\_data)

ggAcf(ts\_data)

ggPacf(ts\_data)

adf.test(ts\_data)

# Differencing the time series data

diff1 <- diff(ts\_data)

adf.test(diff1)

ggtsdisplay(diff1)

diff2 <- diff(diff1)

adf.test(diff2)

ggtsdisplay(diff2)

# Fit the ARIMA model

fitdata <- Arima(diff2, order = c(2, 2, 0))

coeftest(fitdata)

summary(fitdata)

checkresiduals(fitdata)

# Forecasting

fc <- forecast(fitdata, h = 10)

print(fc)

plot(fc)

**Code for ARMA:**

# Load required libraries

library(ggplot2)

library(fpp2)

library(tseries)

library(forecast)

library(zoo)

library(lmtest)

# ARMA lecture 8

# Set working directory (ensure this is the directory containing the CSV file)

setwd("C:/Users/ASUS/Downloads")

# Read the data from the CSV file

data <- read.csv("Inflation.csv")

# Create a time series object

ts\_data <- ts(data$Inflation.Rate, start = 1991, frequency = 1)

# Display the time series data

ggtsdisplay(ts\_data)

autoplot(ts\_data)

ggAcf(ts\_data)

ggPacf(ts\_data)

# Check if the series is stationary

adf\_test\_result <- adf.test(ts\_data)

print(adf\_test\_result)

# Differencing the time series data if it's not stationary

if(adf\_test\_result$p.value > 0.05) {

diff\_ts\_data <- diff(ts\_data)

adf\_test\_result\_diff <- adf.test(diff\_ts\_data)

print(adf\_test\_result\_diff)

# Fit an ARMA model to the differenced time series data

arma\_fit <- Arima(diff\_ts\_data, order = c(2, 0, 2))

ggtsdisplay(diff\_ts\_data)

} else {

# Fit an ARMA model to the original time series data

arma\_fit <- Arima(ts\_data, order = c(2, 0, 2))

}

# Summary of the fitted ARMA model

summary(arma\_fit)

# Check residuals of the fitted model

checkresiduals(arma\_fit)

# Forecasting with the fitted ARMA model

fc\_arma <- forecast(arma\_fit, h = 10)

print(fc\_arma)

plot(fc\_arma)