Forecast the CocaCola Sales data set. Prepare a document for each model explaining how many dummy variables you have created and RMSE value for each model. Finally which model you will use for Forecasting.

**Ans :**

**R Code :**

## Forecasting Data Driven

########## CocaCola Sales Data Set #########

install.packages(c("forecast","fpp","smooth","tseries"))

library(forecast)

library(fpp)

library(smooth)

library(tseries)

library(readxl)

cocacolasales<-read\_xlsx("D:\\Data Science\\Excelr\\Assignments\\Assignment\\Forecasting\\CocaCola\_Sales\_Rawdata.xlsx")

View(cocacolasales)

# Converting data into time series object

sales<-ts(cocacolasales$Sales,frequency = 4,start=c(86))

View(sales)

plot(sales)

# dividing entire data into training and testing data

train<-sales[1:38]

test<-sales[39:42] # Considering only 4 Quarters of data for testing because data itself is Quarterly

# seasonal data

# converting time series object

train<-ts(train,frequency = 4)

test<-ts(test,frequency = 4)

# Plotting time series data

plot(train) # Visualization shows that it has level, trend, seasonality => Additive seasonality

#### USING HoltWinters function ################

# Optimum values

# with alpha = 0.2 which is default value

# Assuming time series data has only level parameter

hw\_a<-HoltWinters(train,alpha = 0.2,beta = F,gamma = F)

hwa\_pred<-data.frame(predict(hw\_a,n.ahead=4))

# By looking at the plot the forecasted values are not showing any characters of train data

plot(forecast(hw\_a,h=4))

hwa\_mape<-MAPE(hwa\_pred$fit,test)\*100

# with alpha = 0.2, beta = 0.1

# Assuming time series data has level and trend parameter

hw\_ab<-HoltWinters(train,alpha = 0.2,beta = 0.1,gamma = F)

hwab\_pred<-data.frame(predict(hw\_ab,n.ahead = 4))

# by looking at the plot the forecasted values are still missing some characters exhibited by train data

plot(forecast(hw\_ab,h=4))

hwab\_mape<-MAPE(hwab\_pred$fit,test)\*100

# with alpha = 0.2, beta = 0.1, gamma = 0.1

# Assuming time series data has level,trend and seasonality

hw\_abg<-HoltWinters(train,alpha = 0.2,beta = 0.1,gamma = 0.1)

hwabg\_pred<-data.frame(predict(hw\_abg,n.ahead = 4))

# by looking at the plot the characters of forecasted values are closely following historical data

plot(forecast(hw\_abg,h=4))

hwabg\_mape<-MAPE(hwabg\_pred$fit,test)\*100

# With out optimum values

hw\_na<-HoltWinters(train,beta = F,gamma = F)

hwna\_pred<-data.frame(predict(hw\_na,n.ahead = 4))

hwna\_pred

plot(forecast(hw\_na,h=4))

hwna\_mape<-MAPE(hwna\_pred$fit,test)\*100

hw\_nab<-HoltWinters(train,gamma=F)

hwnab\_pred<-data.frame(predict(hw\_nab,n.ahead=4))

hwnab\_pred

plot(forecast(hw\_nab,h=4))

hwnab\_mape<-MAPE(hwnab\_pred$fit,test)\*100

hw\_nabg<-HoltWinters(train)

hwnabg\_pred<-data.frame(predict(hw\_nabg,n.ahead =4))

hwnabg\_pred

plot(forecast(hw\_nabg,h=4))

hwnabg\_mape<-MAPE(hwnabg\_pred$fit,test)\*100

df\_mape<-data.frame(c("hwa\_mape","hwab\_mape","hwabg\_mape","hwna\_mape","hwnab\_mape","hwnabg\_mape"),c(hwa\_mape,hwab\_mape,hwabg\_mape,hwna\_mape,hwnab\_mape,hwnabg\_mape))

colnames(df\_mape)<-c("MAPE","VALUES")

View(df\_mape)

# Based on the MAPE value who choose holts winter exponential tecnique which assumes the time series

# Data level, trend, seasonality characters with default values of alpha, beta and gamma

new\_model <- HoltWinters(sales)

plot(forecast(new\_model,n.ahead=8))

# Forecasted values for the next 4 quarters

forecast\_new <- data.frame(predict(new\_model,n.ahead=4))

forecast\_new

######## ARIMA Model #############

# Converting data into time series object

sales<-ts(cocacolasales$Sales,frequency = 4,start=c(86))

View(sales)

plot(sales)

# dividing entire data into training and testing data

train<-sales[1:38]

test<-sales[39:42] # Considering only 4 Quarters of data for testing because data itself is Quarterly

# seasonal data

# converting time series object

train<-ts(train,frequency = 4)

test<-ts(test,frequency = 4)

plot(train)

acf(train)

pacf(train)

# Auto.Arima model on the price agg data

library(forecast)

model\_AA <- auto.arima(train)

model\_AA

pred\_AA <- data.frame(forecast(model\_AA))

acf(model\_AA$residuals)

pacf(model\_AA$residuals)

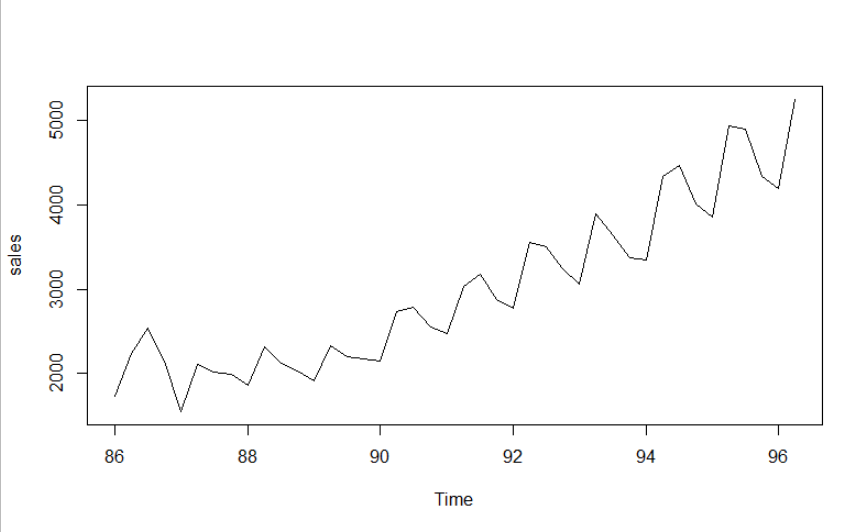
windows()

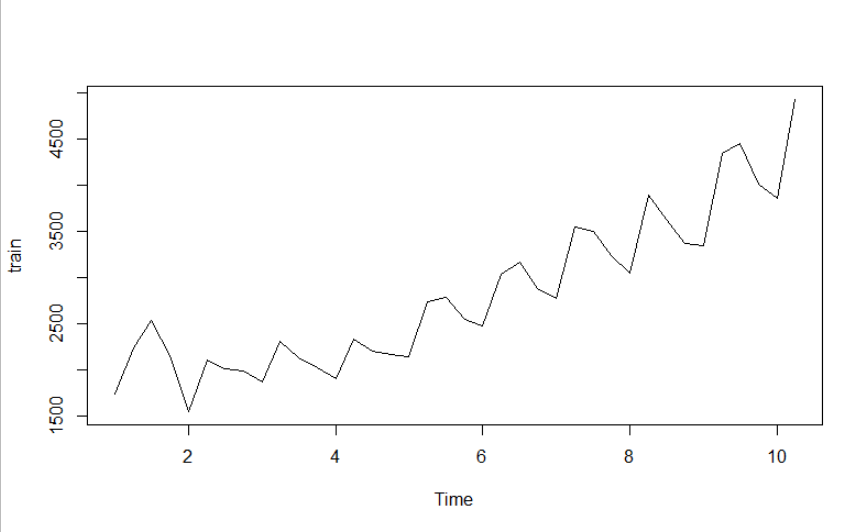
plot(forecast(model\_AA,h=12),xaxt="n")

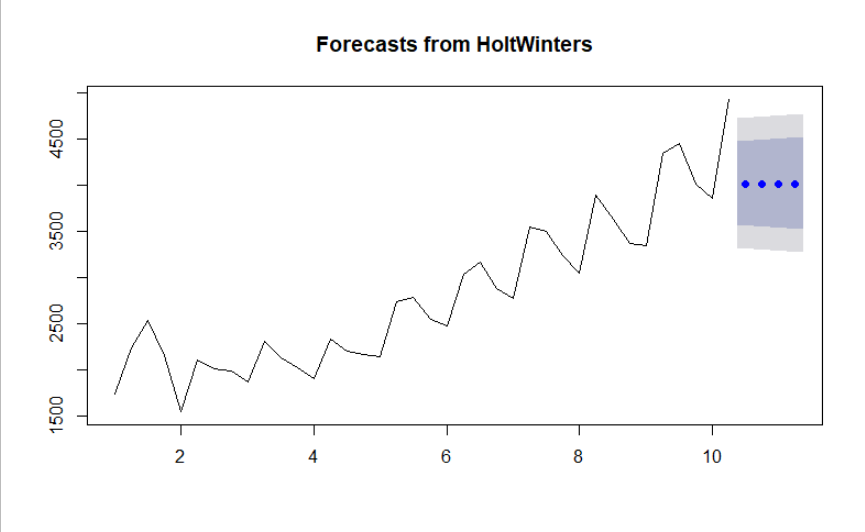
**Results :**

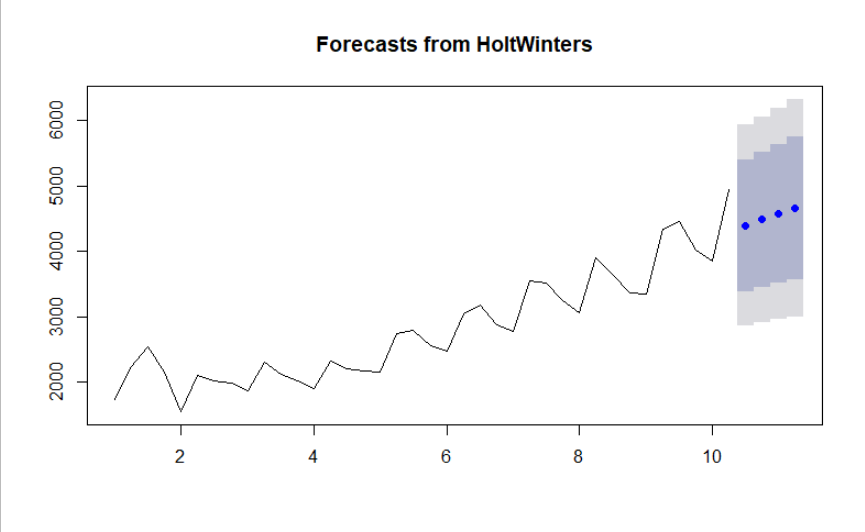
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| > hw\_na<-HoltWinters(train,beta = F,gamma = F)  > hwna\_pred<-data.frame(predict(hw\_na,n.ahead = 4))  > hwna\_pred  fit  1 4456.709  2 4456.709  3 4456.709  4 4456.709  > plot(forecast(hw\_na,h=4))  > hwna\_mape<-MAPE(hwna\_pred$fit,test)\*100  > hw\_nab<-HoltWinters(train,gamma=F)  > hwnab\_pred<-data.frame(predict(hw\_nab,n.ahead=4))  > hwnab\_pred  fit  1 4763.920  2 4946.695  3 5129.470  4 5312.244  > plot(forecast(hw\_nab,h=4))  > hwnab\_mape<-MAPE(hwnab\_pred$fit,test)\*100  > hw\_nabg<-HoltWinters(train)  > hwnabg\_pred<-data.frame(predict(hw\_nabg,n.ahead =4))  > hwnabg\_pred  fit  1 4876.456  2 4451.734  3 4353.283  4 5408.914  > plot(forecast(hw\_nabg,h=4))  > hwnabg\_mape<-MAPE(hwnabg\_pred$fit,test)\*100  > df\_mape<-data.frame(c("hwa\_mape","hwab\_mape","hwabg\_mape","hwna\_mape","hwnab\_mape","hwnabg\_mape"),c(hwa\_mape,hwab\_mape,hwabg\_mape,hwna\_mape,hwnab\_mape,hwnabg\_mape))  > colnames(df\_mape)<-c("MAPE","VALUES")  > View(df\_mape)   | **MAPE** | | **VALUES** | | | --- | --- | --- | --- | |  |  | |  | | **1** | hwa\_mape | | 16.126335 | | **2** | hwab\_mape | | 8.928086 | | **3** | hwabg\_mape | | 3.549841 | | **4** | hwna\_mape | | 9.093032 | | **5** | hwnab\_mape | | 8.627520 | | **6** | hwnabg\_mape | | 2.397211 |   > new\_model <- HoltWinters(sales)  > plot(forecast(new\_model,n.ahead=8))  > # Forecasted values for the next 4 quarters  > forecast\_new <- data.frame(predict(new\_model,n.ahead=4))  > forecast\_new  fit  1 5215.150  2 4672.568  3 4556.262  4 5630.019  > # Converting data into time series object  > sales<-ts(cocacolasales$Sales,frequency = 4,start=c(86))  > View(sales)  > plot(sales)  > # dividing entire data into training and testing data  > train<-sales[1:38]  > test<-sales[39:42] # Considering only 4 Quarters of data for testing because data itself is Quarterly  > # seasonal data  > # converting time series object  > train<-ts(train,frequency = 4)  > test<-ts(test,frequency = 4)  > plot(train)  > acf(train)  > pacf(train)  > # Auto.Arima model on the price agg data  > library(forecast)  > model\_AA <- auto.arima(train)  > model\_AA  Series: train  ARIMA(0,1,0)(0,1,0)[4]  sigma^2 estimated as 32602: log likelihood=-218.29  AIC=438.59 AICc=438.72 BIC=440.09  > pred\_AA <- data.frame(forecast(model\_AA))  > acf(model\_AA$residuals)  > pacf(model\_AA$residuals)  > plot(forecast(model\_AA,h=12),xaxt="n") |
|  |
| |  | | --- | |  | |

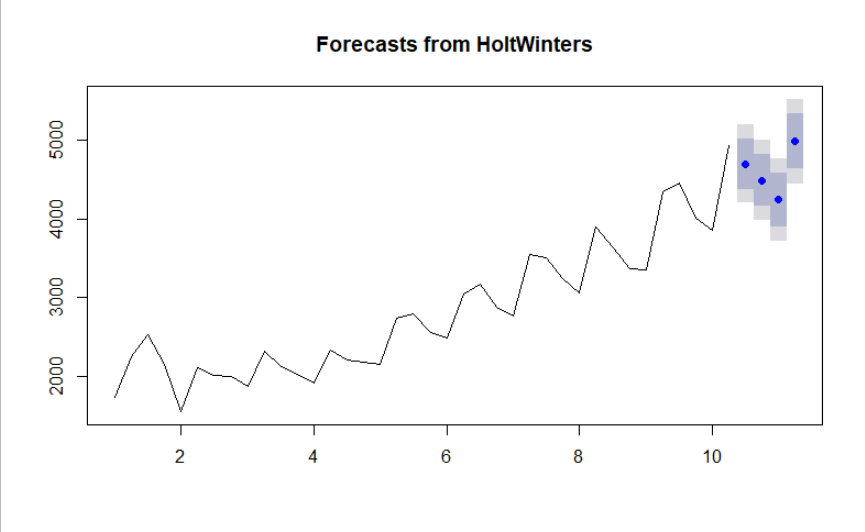
**Plots :**

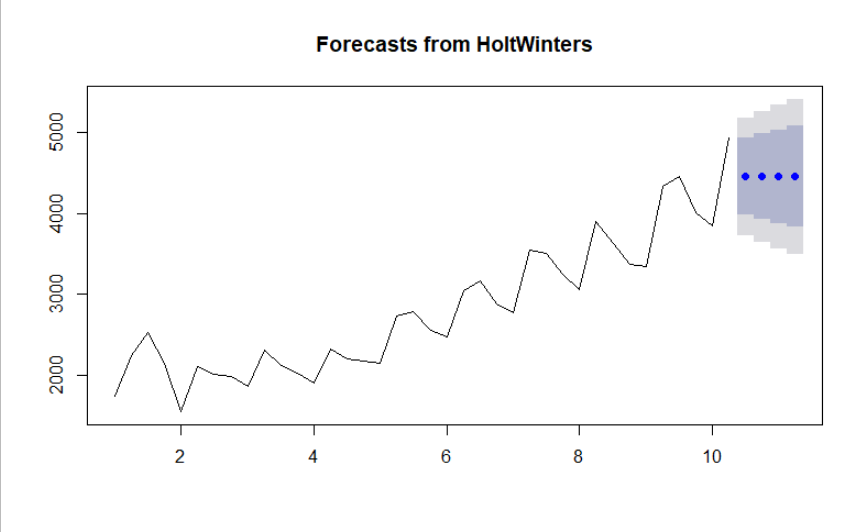


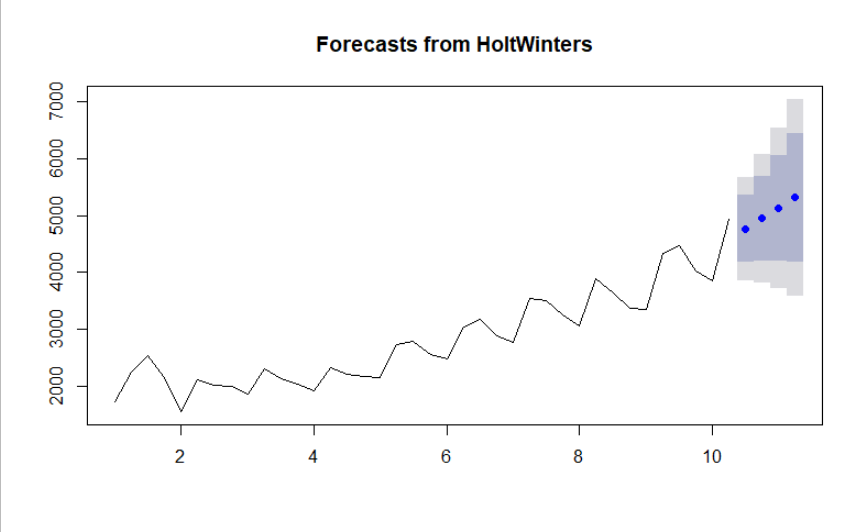


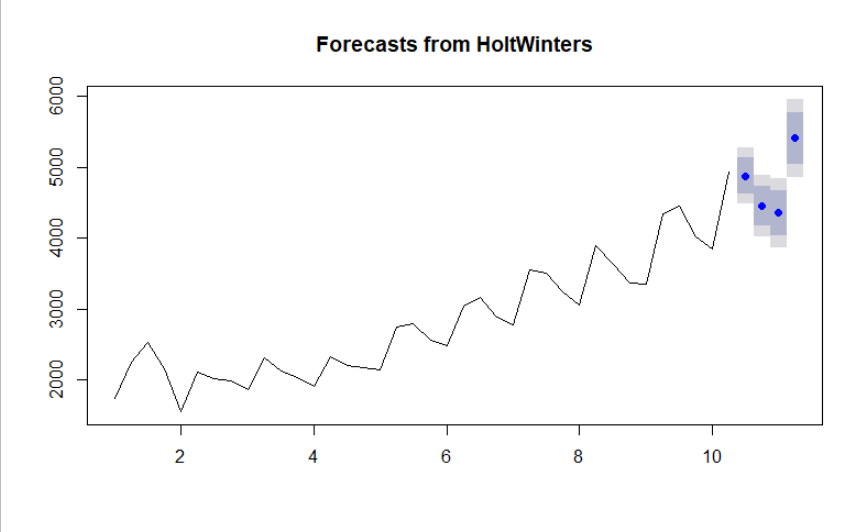


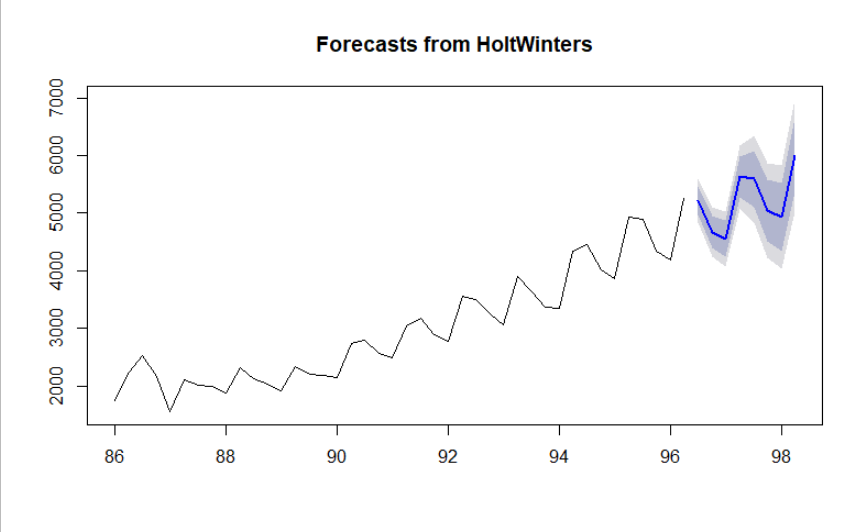


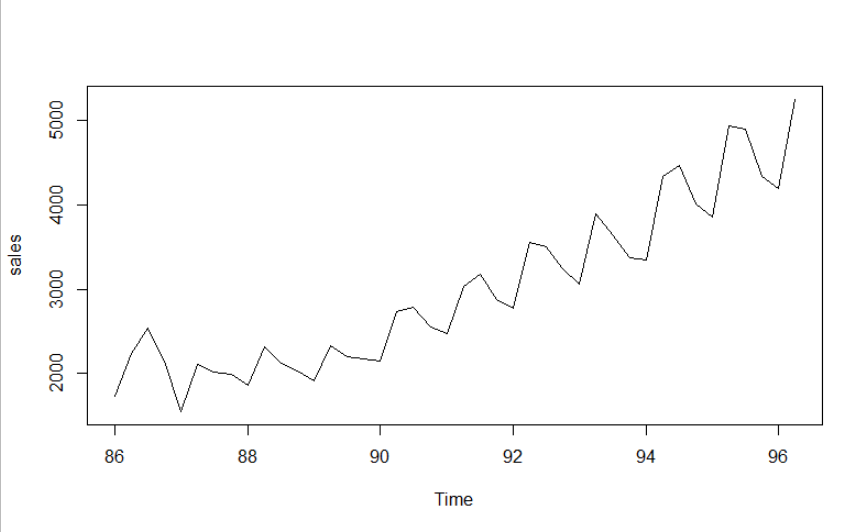


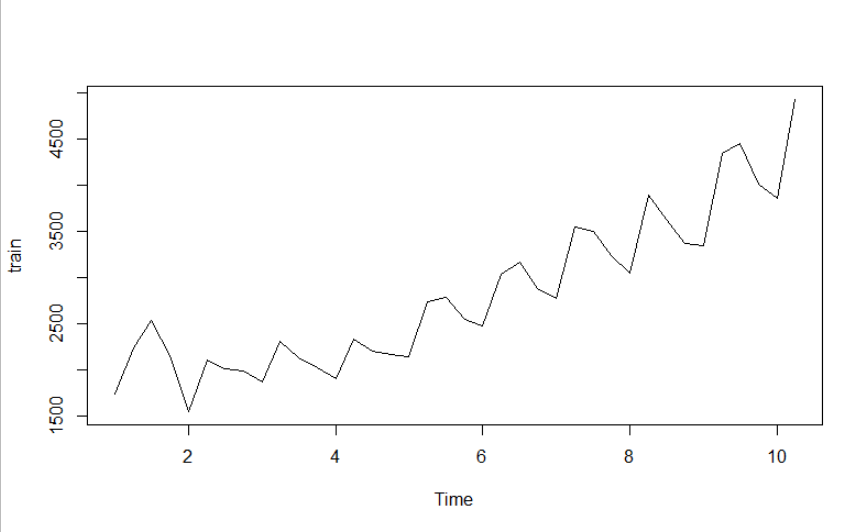


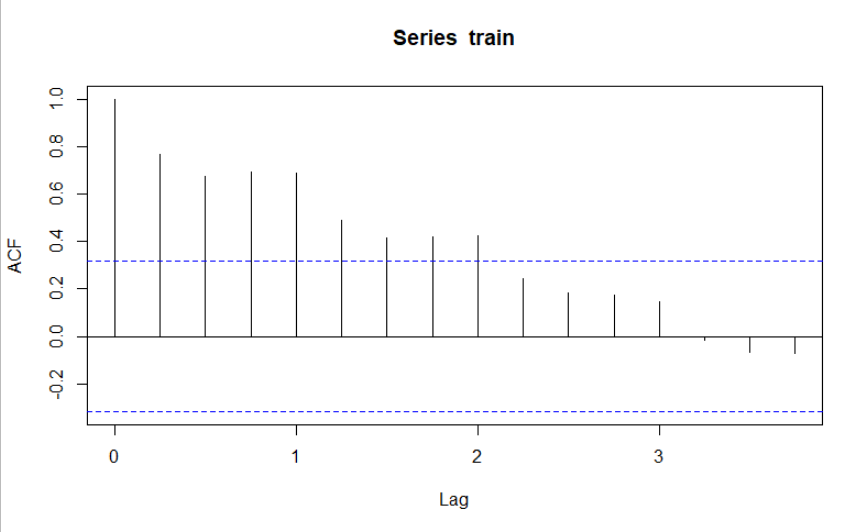


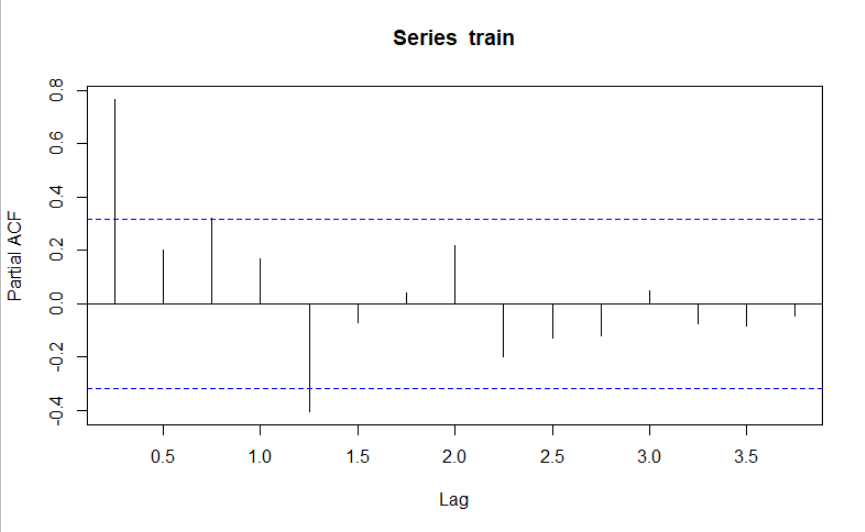


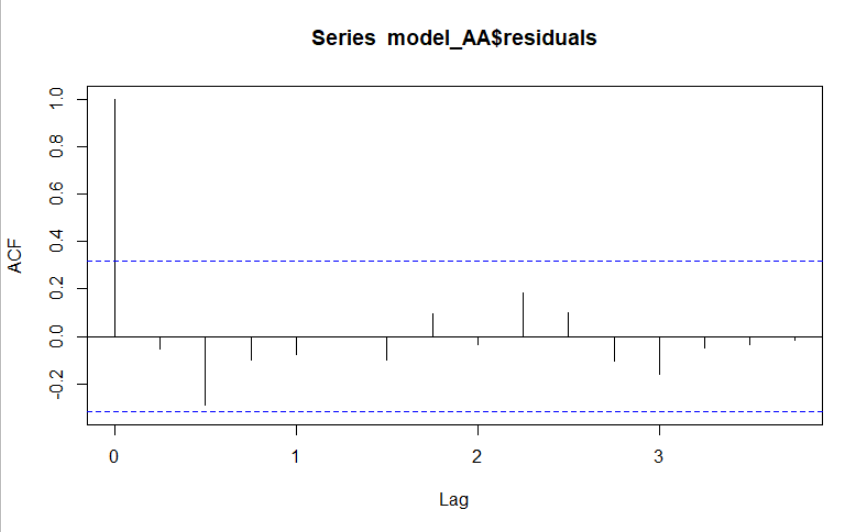


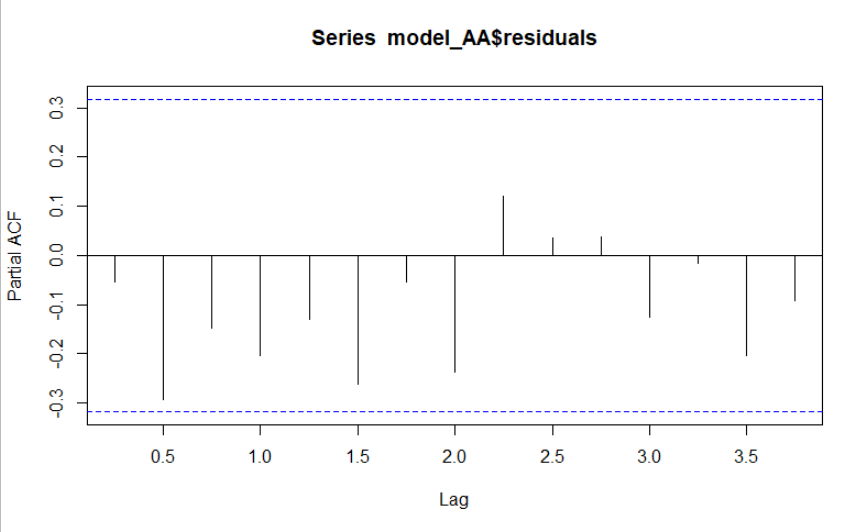


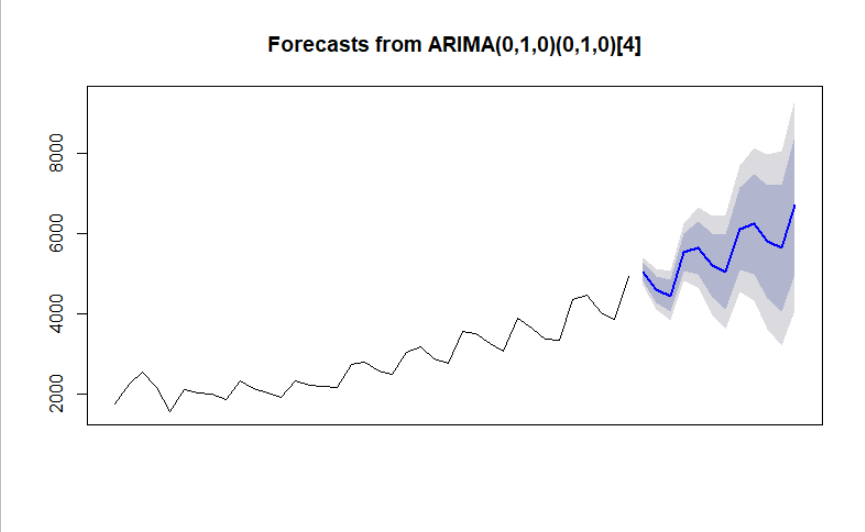












**Inference :**

The MAPE values of various models are as below :

| **MAPE** | | **VALUES** | |
| --- | --- | --- | --- |
|  |  | |  |
| **1** | hwa\_mape | | 16.126335 |
| **2** | hwab\_mape | | 8.928086 |
| **3** | hwabg\_mape | | 3.549841 |
| **4** | hwna\_mape | | 9.093032 |
| **5** | hwnab\_mape | | 8.627520 |
| **6** | hwnabg\_mape | | 2.397211 |