Forecast the Airlines Passengers 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 Model Driven

########## Airlines Data Set #########

library(readr)

Airlines <- read\_xlsx('D:\\Data Science\\Excelr\\Assignments\\Assignment\\Forecasting\\Airlines+Data.xlsx')

View(Airlines) # Seasonality 12 months

windows()

plot(Airlines$Passengers,type="l")

# So creating 11 dummy variables

X<- data.frame(outer(rep(month.abb,length = 96), month.abb,"==") + 0 )# Creating dummies for 12 months

View(X)

colnames(X)<-month.abb # Assigning month names

View(X)

airlinesdata<-cbind(Airlines,X)

View(airlinesdata)

airlinesdata["t"]<- 1:96

View(airlinesdata)

airlinesdata["log\_passengers"]<-log(airlinesdata["Passengers"])

airlinesdata["t\_square"]<-airlinesdata["t"]\*airlinesdata["t"]

##Data Partition

train<-airlinesdata[1:84,]

test<-airlinesdata[85:96,]

########################### LINEAR MODEL #############################

linear\_model<-lm(Passengers~t,data=train)

summary(linear\_model)

linear\_pred<-data.frame(predict(linear\_model,interval='predict',newdata =test))

View(linear\_pred)

rmse\_linear<-sqrt(mean((test$Passengers-linear\_pred$fit)^2,na.rm = T))

rmse\_linear

######################### Exponential #################################

expo\_model<-lm(log\_passengers~t,data=train)

summary(expo\_model)

expo\_pred<-data.frame(predict(expo\_model,interval='predict',newdata=test))

rmse\_expo<-sqrt(mean((test$Passengers-exp(expo\_pred$fit))^2,na.rm = T))

rmse\_expo

######################### Quadratic ####################################

Quad\_model<-lm(Passengers~t+t\_square,data=train)

summary(Quad\_model)

Quad\_pred<-data.frame(predict(Quad\_model,interval='predict',newdata=test))

rmse\_Quad<-sqrt(mean((test$Passengers-Quad\_pred$fit)^2,na.rm=T))

rmse\_Quad

######################### Additive Seasonality #########################

sea\_add\_model<-lm(Passengers~Jan+Feb+Mar+Apr+May+Jun+Jul+Aug+Sep+Oct+Nov,data=train)

summary(sea\_add\_model)

sea\_add\_pred<-data.frame(predict(sea\_add\_model,newdata=test,interval='predict'))

rmse\_sea\_add<-sqrt(mean((test$Passengers-sea\_add\_pred$fit)^2,na.rm = T))

rmse\_sea\_add

######################## Additive Seasonality with Linear #################

Add\_sea\_Linear\_model<-lm(Passengers~t+Jan+Feb+Mar+Apr+May+Jun+Jul+Aug+Sep+Oct+Nov,data=train)

summary(Add\_sea\_Linear\_model)

Add\_sea\_Linear\_pred<-data.frame(predict(Add\_sea\_Linear\_model,interval='predict',newdata=test))

rmse\_Add\_sea\_Linear<-sqrt(mean((test$Passengers-Add\_sea\_Linear\_pred$fit)^2,na.rm=T))

rmse\_Add\_sea\_Linear

######################## Additive Seasonality with Quadratic #################

Add\_sea\_Quad\_model<-lm(Passengers~t+t\_square+Jan+Feb+Mar+Apr+May+Jun+Jul+Aug+Sep+Oct+Nov,data=train)

summary(Add\_sea\_Quad\_model)

Add\_sea\_Quad\_pred<-data.frame(predict(Add\_sea\_Quad\_model,interval='predict',newdata=test))

rmse\_Add\_sea\_Quad<-sqrt(mean((test$Passengers-Add\_sea\_Quad\_pred$fit)^2,na.rm=T))

rmse\_Add\_sea\_Quad

######################## Multiplicative Seasonality #########################

multi\_sea\_model<-lm(log\_passengers~Jan+Feb+Mar+Apr+May+Jun+Jul+Aug+Sep+Oct+Nov,data = train)

summary(multi\_sea\_model)

multi\_sea\_pred<-data.frame(predict(multi\_sea\_model,newdata=test,interval='predict'))

rmse\_multi\_sea<-sqrt(mean((test$Passengers-exp(multi\_sea\_pred$fit))^2,na.rm = T))

rmse\_multi\_sea

######################## Multiplicative Seasonality Linear trend ##########################

multi\_add\_sea\_model<-lm(log\_passengers~t+Jan+Feb+Mar+Apr+May+Jun+Jul+Aug+Sep+Oct+Nov,data = train)

summary(multi\_add\_sea\_model)

multi\_add\_sea\_pred<-data.frame(predict(multi\_add\_sea\_model,newdata=test,interval='predict'))

rmse\_multi\_add\_sea<-sqrt(mean((test$Passengers-exp(multi\_add\_sea\_pred$fit))^2,na.rm = T))

rmse\_multi\_add\_sea

# Preparing table on model and it's RMSE values

table\_rmse<-data.frame('Model'=c("rmse\_linear","rmse\_expo","rmse\_Quad","rmse\_sea\_add","rmse\_Add\_sea\_Quad","rmse\_multi\_sea","rmse\_multi\_add\_sea"),'RMSE'=c(rmse\_linear,rmse\_expo,rmse\_Quad,rmse\_sea\_add,rmse\_Add\_sea\_Quad,rmse\_multi\_sea,rmse\_multi\_add\_sea))

View(table\_rmse)

colnames(table\_rmse)<-c("model","RMSE")

View(table\_rmse)

# Use entire data : Multiplicative Seasonality Linear trend has least RMSE value

new\_model <- lm(log\_passengers~t+Jan+Feb+Mar+Apr+May+Jun+Jul+Aug+Sep+Oct+Nov,data=airlinesdata)

#predict(new\_model,n.ahead=1)

# Getting residuals

resid <- residuals(new\_model)

resid[1:10]

windows()

hist(resid)

windows()

acf(resid,lag.max = 10)

# By principal of parcimony we will consider lag - 1 as we have so

# many significant lags

# Building Autoregressive model on residuals consider lag-1

k <- arima(resid, order=c(1,0,0))

windows();

acf(k$residuals,lag.max = 15)

pred\_res<- predict(arima(resid,order=c(1,0,0)),n.ahead = 12)

str(pred\_res)

pred\_res$pred

acf(k$residuals)

write.csv(airlinesdata,file="airlinesnew.csv",col.names = F,row.names = F)

####################### Predicting new data #############################

library(readxl)

test\_data<-read\_excel(file.choose(),1) #Load Predict\_new\_Airlines Data.xlsx

View(test\_data)

#test\_data<-Predict\_new

pred\_new<-data.frame(predict(new\_model,newdata=test\_data,interval = 'predict'))

View(pred\_new)

#pred\_re<-pred\_res$pred[1:12]

pred\_new$fit <- pred\_new$fit+pred\_res$pred[1:12]

View(pred\_new)

**Results :**

> linear\_model<-lm(Passengers~t,data=train)

> summary(linear\_model)

Call:

lm(formula = Passengers ~ t, data = train)

Residuals:

Min 1Q Median 3Q Max

-55.419 -17.202 -0.705 16.546 88.438

Coefficients:

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

(Intercept) 106.2708 5.9287 17.93 <2e-16 \*\*\*

t 2.1429 0.1212 17.69 <2e-16 \*\*\*

---

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

Residual standard error: 26.93 on 82 degrees of freedom

Multiple R-squared: 0.7923, Adjusted R-squared: 0.7898

F-statistic: 312.8 on 1 and 82 DF, p-value: < 2.2e-16

> linear\_pred<-data.frame(predict(linear\_model,interval='predict',newdata =test))

> View(linear\_pred)

> rmse\_linear<-sqrt(mean((test$Passengers-linear\_pred$fit)^2,na.rm = T))

> rmse\_linear

[1] 53.19924

> expo\_model<-lm(log\_passengers~t,data=train)

> summary(expo\_model)

Call:

lm(formula = log\_passengers ~ t, data = train)

Residuals:

Min 1Q Median 3Q Max

-0.28906 -0.07775 -0.01528 0.07901 0.25104

Coefficients:

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

(Intercept) 4.770262 0.027693 172.26 <2e-16 \*\*\*

t 0.011087 0.000566 19.59 <2e-16 \*\*\*

---

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

Residual standard error: 0.1258 on 82 degrees of freedom

Multiple R-squared: 0.8239, Adjusted R-squared: 0.8218

F-statistic: 383.7 on 1 and 82 DF, p-value: < 2.2e-16

> expo\_pred<-data.frame(predict(expo\_model,interval='predict',newdata=test))

> rmse\_expo<-sqrt(mean((test$Passengers-exp(expo\_pred$fit))^2,na.rm = T))

> rmse\_expo

[1] 46.05736

> Quad\_model<-lm(Passengers~t+t\_square,data=train)

> summary(Quad\_model)

Call:

lm(formula = Passengers ~ t + t\_square, data = train)

Residuals:

Min 1Q Median 3Q Max

-56.985 -15.652 -3.801 16.360 83.241

Coefficients:

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

(Intercept) 1.148e+02 8.996e+00 12.758 < 2e-16 \*\*\*

t 1.549e+00 4.885e-01 3.172 0.00214 \*\*

t\_square 6.982e-03 5.569e-03 1.254 0.21350

---

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

Residual standard error: 26.83 on 81 degrees of freedom

Multiple R-squared: 0.7963, Adjusted R-squared: 0.7912

F-statistic: 158.3 on 2 and 81 DF, p-value: < 2.2e-16

> Quad\_pred<-data.frame(predict(Quad\_model,interval='predict',newdata=test))

> rmse\_Quad<-sqrt(mean((test$Passengers-Quad\_pred$fit)^2,na.rm=T))

> rmse\_Quad

[1] 48.05189

> sea\_add\_model<-lm(Passengers~Jan+Feb+Mar+Apr+May+Jun+Jul+Aug+Sep+Oct+Nov,data=train)

> summary(sea\_add\_model)

Call:

lm(formula = Passengers ~ Jan + Feb + Mar + Apr + May + Jun +

Jul + Aug + Sep + Oct + Nov, data = train)

Residuals:

Min 1Q Median 3Q Max

-91.571 -51.393 2.143 38.464 124.429

Coefficients:

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

(Intercept) 189.429 21.746 8.711 7.21e-13 \*\*\*

Jan -20.143 30.753 -0.655 0.515

Feb -19.286 30.753 -0.627 0.533

Mar 8.000 30.753 0.260 0.796

Apr 1.857 30.753 0.060 0.952

May 1.143 30.753 0.037 0.970

Jun 25.143 30.753 0.818 0.416

Jul 50.143 30.753 1.630 0.107

Aug 49.286 30.753 1.603 0.113

Sep 24.143 30.753 0.785 0.435

Oct -1.000 30.753 -0.033 0.974

Nov -24.286 30.753 -0.790 0.432

---

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

Residual standard error: 57.53 on 72 degrees of freedom

Multiple R-squared: 0.1674, Adjusted R-squared: 0.04015

F-statistic: 1.316 on 11 and 72 DF, p-value: 0.2337

> sea\_add\_pred<-data.frame(predict(sea\_add\_model,newdata=test,interval='predict'))

> rmse\_sea\_add<-sqrt(mean((test$Passengers-sea\_add\_pred$fit)^2,na.rm = T))

> rmse\_sea\_add

[1] 132.8198

> Add\_sea\_Linear\_model<-lm(Passengers~t+Jan+Feb+Mar+Apr+May+Jun+Jul+Aug+Sep+Oct+Nov,data=train)

> summary(Add\_sea\_Linear\_model)

Call:

lm(formula = Passengers ~ t + Jan + Feb + Mar + Apr + May + Jun +

Jul + Aug + Sep + Oct + Nov, data = train)

Residuals:

Min 1Q Median 3Q Max

-33.952 -8.679 -0.286 6.976 46.714

Coefficients:

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

(Intercept) 85.80952 5.87255 14.612 < 2e-16 \*\*\*

t 2.15873 0.06117 35.289 < 2e-16 \*\*\*

Jan 3.60317 7.22378 0.499 0.61947

Feb 2.30159 7.21834 0.319 0.75077

Mar 27.42857 7.21341 3.802 0.00030 \*\*\*

Apr 19.12698 7.20900 2.653 0.00983 \*\*

May 16.25397 7.20511 2.256 0.02716 \*

Jun 38.09524 7.20173 5.290 1.30e-06 \*\*\*

Jul 60.93651 7.19887 8.465 2.30e-12 \*\*\*

Aug 57.92063 7.19653 8.048 1.36e-11 \*\*\*

Sep 30.61905 7.19471 4.256 6.26e-05 \*\*\*

Oct 3.31746 7.19341 0.461 0.64608

Nov -22.12698 7.19263 -3.076 0.00298 \*\*

---

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

Residual standard error: 13.46 on 71 degrees of freedom

Multiple R-squared: 0.9551, Adjusted R-squared: 0.9475

F-statistic: 125.8 on 12 and 71 DF, p-value: < 2.2e-16

> Add\_sea\_Linear\_pred<-data.frame(predict(Add\_sea\_Linear\_model,interval='predict',newdata=test))

> rmse\_Add\_sea\_Linear<-sqrt(mean((test$Passengers-Add\_sea\_Linear\_pred$fit)^2,na.rm=T))

> rmse\_Add\_sea\_Linear

[1] 35.34896

> Add\_sea\_Quad\_model<-lm(Passengers~t+t\_square+Jan+Feb+Mar+Apr+May+Jun+Jul+Aug+Sep+Oct+Nov,data=train)

> summary(Add\_sea\_Quad\_model)

Call:

lm(formula = Passengers ~ t + t\_square + Jan + Feb + Mar + Apr +

May + Jun + Jul + Aug + Sep + Oct + Nov, data = train)

Residuals:

Min 1Q Median 3Q Max

-32.297 -7.694 0.392 7.735 40.922

Coefficients:

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

(Intercept) 95.003657 6.438877 14.755 < 2e-16 \*\*\*

t 1.507479 0.233462 6.457 1.20e-08 \*\*\*

t\_square 0.007662 0.002660 2.881 0.005263 \*\*

Jan 3.603175 6.878882 0.524 0.602071

Feb 2.378205 6.873752 0.346 0.730393

Mar 27.566483 6.869176 4.013 0.000148 \*\*\*

Apr 19.310867 6.865106 2.813 0.006367 \*\*

May 16.468498 6.861505 2.400 0.019052 \*

Jun 38.325091 6.858349 5.588 4.11e-07 \*\*\*

Jul 61.166361 6.855628 8.922 3.67e-13 \*\*\*

Aug 58.135165 6.853340 8.483 2.36e-12 \*\*\*

Sep 30.802930 6.851500 4.496 2.68e-05 \*\*\*

Oct 3.455372 6.850131 0.504 0.615547

Nov -22.050366 6.849272 -3.219 0.001949 \*\*

---

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

Residual standard error: 12.81 on 70 degrees of freedom

Multiple R-squared: 0.9598, Adjusted R-squared: 0.9524

F-statistic: 128.7 on 13 and 70 DF, p-value: < 2.2e-16

> Add\_sea\_Quad\_pred<-data.frame(predict(Add\_sea\_Quad\_model,interval='predict',newdata=test))

> rmse\_Add\_sea\_Quad<-sqrt(mean((test$Passengers-Add\_sea\_Quad\_pred$fit)^2,na.rm=T))

> rmse\_Add\_sea\_Quad

[1] 26.36082

> multi\_sea\_model<-lm(log\_passengers~Jan+Feb+Mar+Apr+May+Jun+Jul+Aug+Sep+Oct+Nov,data = train)

> summary(multi\_sea\_model)

Call:

lm(formula = log\_passengers ~ Jan + Feb + Mar + Apr + May + Jun +

Jul + Aug + Sep + Oct + Nov, data = train)

Residuals:

Min 1Q Median 3Q Max

-0.43858 -0.28085 0.04875 0.22435 0.46174

Coefficients:

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

(Intercept) 5.208117 0.111153 46.856 <2e-16 \*\*\*

Jan -0.112831 0.157194 -0.718 0.475

Feb -0.097238 0.157194 -0.619 0.538

Mar 0.047126 0.157194 0.300 0.765

Apr 0.011394 0.157194 0.072 0.942

May 0.001676 0.157194 0.011 0.992

Jun 0.120037 0.157194 0.764 0.448

Jul 0.227301 0.157194 1.446 0.153

Aug 0.227676 0.157194 1.448 0.152

Sep 0.120513 0.157194 0.767 0.446

Oct -0.006967 0.157194 -0.044 0.965

Nov -0.138701 0.157194 -0.882 0.381

---

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

Residual standard error: 0.2941 on 72 degrees of freedom

Multiple R-squared: 0.1548, Adjusted R-squared: 0.02568

F-statistic: 1.199 on 11 and 72 DF, p-value: 0.3036

> multi\_sea\_pred<-data.frame(predict(multi\_sea\_model,newdata=test,interval='predict'))

> rmse\_multi\_sea<-sqrt(mean((test$Passengers-exp(multi\_sea\_pred$fit))^2,na.rm = T))

> rmse\_multi\_sea

[1] 140.0632

> multi\_add\_sea\_model<-lm(log\_passengers~t+Jan+Feb+Mar+Apr+May+Jun+Jul+Aug+Sep+Oct+Nov,data = train)

> summary(multi\_add\_sea\_model)

Call:

lm(formula = log\_passengers ~ t + Jan + Feb + Mar + Apr + May +

Jun + Jul + Aug + Sep + Oct + Nov, data = train)

Residuals:

Min 1Q Median 3Q Max

-0.142864 -0.031286 0.000823 0.031275 0.105860

Coefficients:

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

(Intercept) 4.6712626 0.0216310 215.952 < 2e-16 \*\*\*

t 0.0111845 0.0002253 49.637 < 2e-16 \*\*\*

Jan 0.0101977 0.0266082 0.383 0.702676

Feb 0.0146065 0.0265881 0.549 0.584480

Mar 0.1477860 0.0265700 5.562 4.41e-07 \*\*\*

Apr 0.1008701 0.0265537 3.799 0.000304 \*\*\*

May 0.0799669 0.0265394 3.013 0.003582 \*\*

Jun 0.1871442 0.0265270 7.055 9.31e-10 \*\*\*

Jul 0.2832229 0.0265164 10.681 < 2e-16 \*\*\*

Aug 0.2724138 0.0265078 10.277 1.08e-15 \*\*\*

Sep 0.1540663 0.0265011 5.814 1.61e-07 \*\*\*

Oct 0.0154017 0.0264963 0.581 0.562893

Nov -0.1275166 0.0264934 -4.813 8.11e-06 \*\*\*

---

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

Residual standard error: 0.04956 on 71 degrees of freedom

Multiple R-squared: 0.9763, Adjusted R-squared: 0.9723

F-statistic: 244 on 12 and 71 DF, p-value: < 2.2e-16

> multi\_add\_sea\_pred<-data.frame(predict(multi\_add\_sea\_model,newdata=test,interval='predict'))

> rmse\_multi\_add\_sea<-sqrt(mean((test$Passengers-exp(multi\_add\_sea\_pred$fit))^2,na.rm = T))

> rmse\_multi\_add\_sea

[1] 10.51917

> table\_rmse<-data.frame('Model'=c("rmse\_linear","rmse\_expo","rmse\_Quad","rmse\_sea\_add","rmse\_Add\_sea\_Quad","rmse\_multi\_sea","rmse\_multi\_add\_sea"),'RMSE'=c(rmse\_linear,rmse\_expo,rmse\_Quad,rmse\_sea\_add,rmse\_Add\_sea\_Quad,rmse\_multi\_sea,rmse\_multi\_add\_sea))

> View(table\_rmse)

> colnames(table\_rmse)<-c("model","RMSE")

> View(table\_rmse)

| **model** | | **RMSE** | |
| --- | --- | --- | --- |
|  |  | |  |
| **1** | **rmse\_linear** | | **53.19924** |
| **2** | **rmse\_expo** | | **46.05736** |
| **3** | **rmse\_Quad** | | **48.05189** |
| **4** | **rmse\_sea\_add** | | **132.81978** |
| **5** | **rmse\_Add\_sea\_Quad** | | **26.36082** |
| **6** | **rmse\_multi\_sea** | | **140.06320** |
| **7** | **rmse\_multi\_add\_sea** | | **10.51917** |

> # Use entire data : Additive seasonality with Quadratic has least RMSE value

> new\_model <- lm(log\_passengers~t+Jan+Feb+Mar+Apr+May+Jun+Jul+Aug+Sep+Oct+Nov,data=airlinesdata)

> #predict(new\_model,n.ahead=1)

> # Getting residuals

> resid <- residuals(new\_model)

> resid[1:10]

1 2 3 4 5 6 7 8 9

0.02553626 0.06719747 0.03613576 0.04599859 -0.01150038 -0.02585831 -0.04017503 -0.03805845 -0.01237700

10

-0.01579672

> hist(resid)

> acf(resid,lag.max = 10)

> # By principal of parcimony we will consider lag - 1 as we have so

> # many significant lags

> # Building Autoregressive model on residuals consider lag-1

> k <- arima(resid, order=c(1,0,0))

> acf(k$residuals,lag.max = 15)

> pred\_res<- predict(arima(resid,order=c(1,0,0)),n.ahead = 12)

> str(pred\_res)

List of 2

$ pred: Time-Series [1:12] from 97 to 108: -0.01373 -0.00918 -0.00613 -0.00408 -0.00271 ...

$ se : Time-Series [1:12] from 97 to 108: 0.0322 0.0388 0.0414 0.0425 0.043 ...

> pred\_res$pred

Time Series:

Start = 97

End = 108

Frequency = 1

[1] -1.372754e-02 -9.181337e-03 -6.129532e-03 -4.080897e-03 -2.705676e-03 -1.782508e-03 -1.162798e-03

[8] -7.467949e-04 -4.675377e-04 -2.800761e-04 -1.542356e-04 -6.976056e-05

> acf(k$residuals)

> ####################### Predicting new data #############################

> library(readxl)

> test\_data<-read\_excel(file.choose(),1) #Load Predict\_new\_Airlines data.xlsx

> View(test\_data)

> #test\_data<-Predict\_new

> pred\_new<-data.frame(predict(new\_model,newdata=test\_data,interval = 'predict'))

> View(pred\_new)

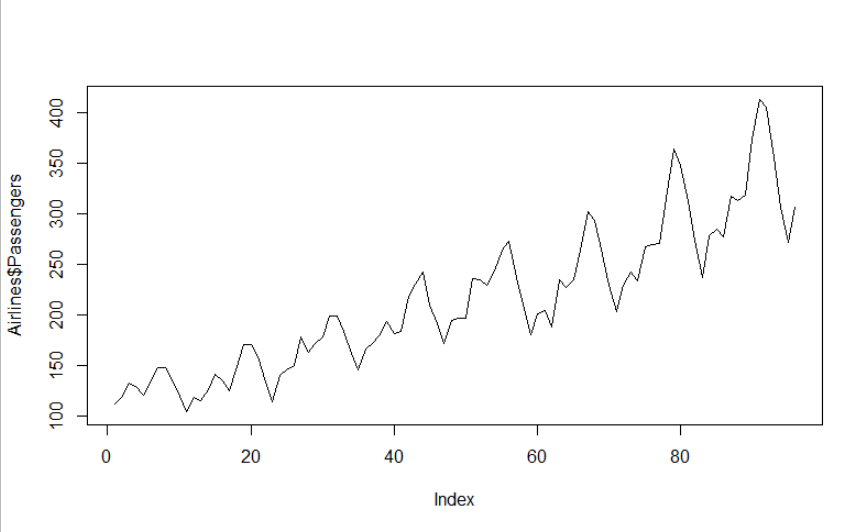
> #pred\_re<-pred\_res$pred[1:12]

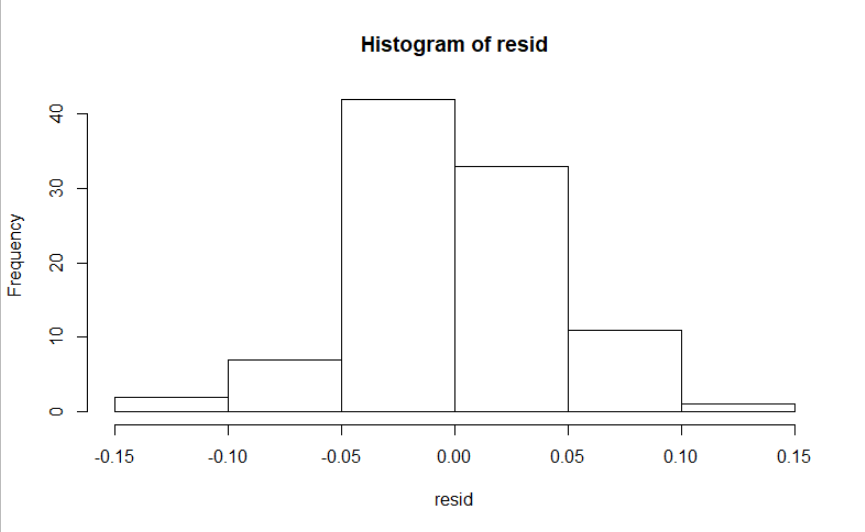
> pred\_new$fit <- pred\_new$fit+pred\_res$pred[1:12]

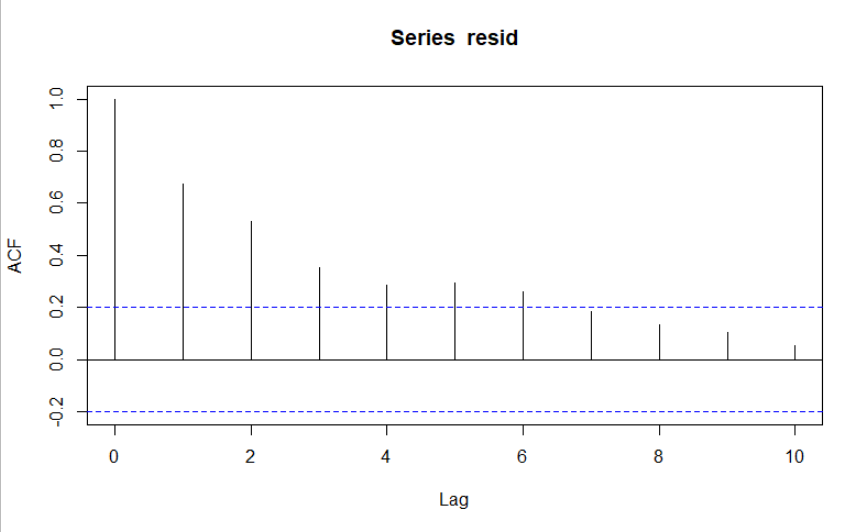
> View(pred\_new)

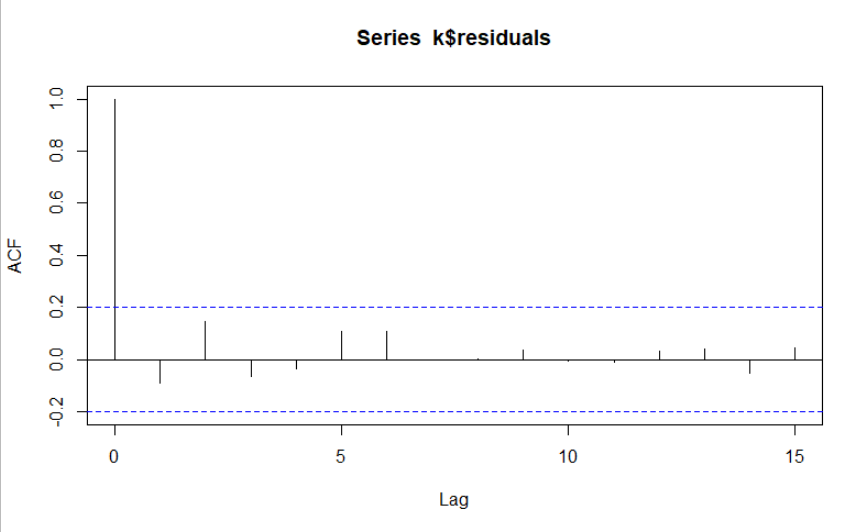
| **fit** | | **lwr** | | **upr** | |
| --- | --- | --- | --- | --- | --- |
|  |  | |  | |  |
| **1** | 5.757028 | | 5.669907 | | 5.871603 |
| **2** | 5.772098 | | 5.680432 | | 5.882128 |
| **3** | 5.918329 | | 5.823611 | | 6.025307 |
| **4** | 5.887525 | | 5.790758 | | 5.992454 |
| **5** | 5.882378 | | 5.784235 | | 5.985932 |
| **6** | 6.007143 | | 5.908078 | | 6.109774 |
| **7** | 6.114017 | | 6.014332 | | 6.216028 |
| **8** | 6.112316 | | 6.012215 | | 6.213911 |
| **9** | 6.002357 | | 5.901976 | | 6.103672 |
| **10** | 5.872433 | | 5.771865 | | 5.973561 |
| **11** | 5.742108 | | 5.641414 | | 5.843110 |
| **12** | 5.878739 | | 5.777961 | | 5.979657 |

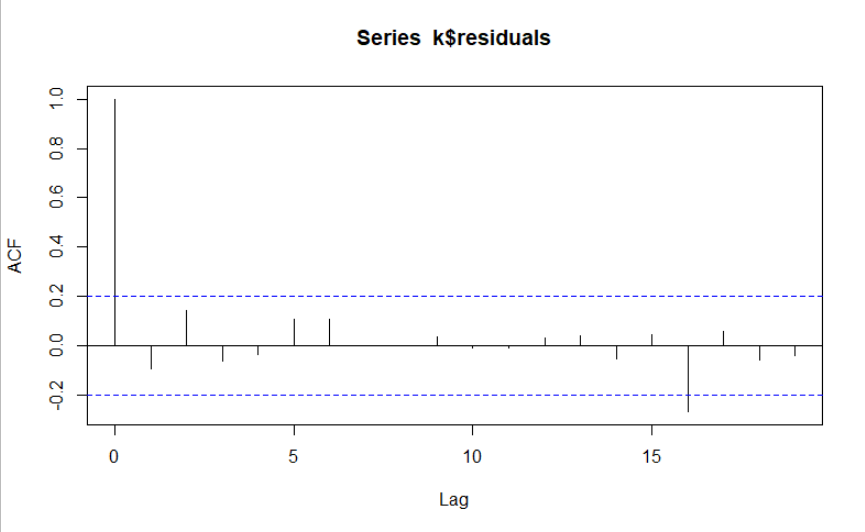
**Plots :**











**Inference :**

The RMSE values of various models are as below :

| **model** | | **RMSE** | |
| --- | --- | --- | --- |
|  |  | |  |
| **1** | **rmse\_linear** | | **53.19924** |
| **2** | **rmse\_expo** | | **46.05736** |
| **3** | **rmse\_Quad** | | **48.05189** |
| **4** | **rmse\_sea\_add** | | **132.81978** |
| **5** | **rmse\_Add\_sea\_Quad** | | **26.36082** |
| **6** | **rmse\_multi\_sea** | | **140.06320** |
| **7** | **rmse\_multi\_add\_sea** | | **10.51917** |

The lowest RMSE value was of Multiplicative Seasonality Linear trend model. So we used this model for forecasting.