#\*\* title: "ch7\_SARIMA" \*\*

#install.packages("forecast")

#install.packages("tseries")

#install.packages("randtests")

#install.packages("astsa")

#install.packages("lmtest")

library(forecast)

library(tseries)

library(randtests)

library(astsa)

library(lmtest)

## Example\_ data C2:

## Checking Stability in the mean and variance .

data<- read.csv(file.choose(),header = T)

d=ts(data,frequency = 4) #time-series objects with seasonal periods 4 .

plot(d,main="Figure (1)") ; abline(h =mean(d),col="red") shapiro.test(d)

shapiro.test(d)

# Apply Box Cox transformation to stabilize the variance.

(lambda <-BoxCox.lambda( d ))

d.B<-BoxCox( d ,lambda)

shapiro.test(d.B)

ggtsdisplay(d.B,lag.max=20, main="Figure (2)" )

#plot(d.B) ; abline(h =mean(d.B),col="red")

# Taking First difference to stabilize the mean and variance\*\*

d.diff<-diff(d,differences =1 )

ggtsdisplay(d.diff,lag.max=20 , main="Figure (3)")

shapiro.test(d.diff)

#plot(d.diff) ; abline(h =mean(d.diff),col="red")

# Finding the appropriate model using ACF and PACF plot:\*\*

#From Figure (2) and Figure (3) we suggest the model SARIMA(1,0,0)(1,1,0)4, SARIMA(1,1,1)(2,0,1)4, SARIMA(2,0,1)(1,1,1)4 ...

#\* Model 1: SARIMA(1,0,0)(1,1,0)[4] model

model1<-Arima(d,order = c(1,0,0),seasonal = c(1,1,0))

#Testing the coefficients & Diagnosing Residuals

coeftest(model1)

tsdiag(model1)

checkresiduals(model1, lag= 12)

checkresiduals(model1, lag= 24,plot=FALSE)

checkresiduals(model1, lag= 36,plot=FALSE)

checkresiduals(model1, lag= 48,plot=FALSE)

runs.test(model1$r)

shapiro.test(model1$residuals)

t.test(model1$r)

#\* Model 2: SARIMA(1,1,1)(2,0,1)[4] model

model2<-Arima(d,order = c(1,1,1),seasonal = c(2,0,1))

#Testing the coefficients & Diagnosing Residuals

coeftest(model2)

tsdiag(model2)

checkresiduals(model2, lag= 12)

checkresiduals(model2, lag= 24,plot=FALSE)

checkresiduals(model2, lag= 36,plot=FALSE)

checkresiduals(model2, lag= 48,plot=FALSE)

runs.test(model2$r)

shapiro.test(model2$residuals)

t.test(model2$r)

#\* Model 3: SARIMA(1,0,0)(2,0,1)[4] model

model3<-Arima(d,order = c(1,0,0),seasonal = c(2,0,1),lambda = lambda, biasadj =TRUE)

#Testing the coefficients & Diagnosing Residuals

coeftest(model3)

tsdiag(model3)

checkresiduals(model3, lag= 12)

checkresiduals(model3, lag= 24,plot=FALSE)

checkresiduals(model3, lag= 36,plot=FALSE)

checkresiduals(model3, lag= 48,plot=FALSE)

runs.test(model3$r)

shapiro.test(model3$residuals)

t.test(model3$r)

#\* Model 4: SARIMAARIMA(0,0,1)(2,0,1)[4]

model4<-Arima(d,order = c(0,0,1),seasonal = c(2,0,1),lambda = lambda, biasadj =TRUE)

#Testing the coefficients & Diagnosing Residuals

coeftest(model4)

tsdiag(model4)

checkresiduals(model4, lag= 12)

checkresiduals(model4, lag= 24,plot=FALSE)

checkresiduals(model4, lag= 36,plot=FALSE)

checkresiduals(model4, lag= 48,plot=FALSE)

runs.test(model4$r)

shapiro.test(model4$residuals)

t.test(model4$r)

#\* best model:

model1$aic

model2$aic

model3$aic

model4$aic

#\* Forecasting using SARIMA(1,1,1)(2,0,1)4:

best.model<-Arima(d,order = c(1,1,1),seasonal = c(2,0,1))

(f=forecast(best.model, h=10,level = c(80, 95)))

autoplot(f)

#~ end \_\_\_\_\_\_

## NOTE:Use the lambda argument in the modelling function and don't transform your data yourself. All the modelling functions in the forecast package will do the BoxCox transformation for you, and back-transform the forecasts when you need them (including bias-adjustment if required).

#biasadj =TRUE: return to raw data, after using BoxCox transformation