HW3\_*2*.R

USER

Tue Dec 11 00:34:23 2018

library(lmtest)

## Loading required package: zoo

##   
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

library(forecast)  
rm(list=ls())  
setwd("C:\\Users\\USER\\Documents\\Github\\Econometrics\\시계열 분석\\HW3")  
rv.data<-read.csv("RV\_IV\_data.csv", header=TRUE)  
#============================#  
# VKOSPI = Y, KOSPI.RV = X #  
#============================#  
# MAKE DATA  
y.t = rv.data$VKOSPI[1:nrow(rv.data)] # VKOSPI  
head(y.t); length(y.t)

## [1] 33.04 33.98 33.17 34.02 38.05 37.56

## [1] 3558

x.t = rv.data$KOSPI[1:(nrow(rv.data))] # KOSPI.RV  
head(x.t); length(x.t)

## [1] 0.01243283 0.01506374 0.01632494 0.01859559 0.03435511 0.01659640

## [1] 3558

# AR 모형 BIC : p=9  
AR.bic = c()  
for( p in 1:10){  
 AR.fit = arima(y.t, order = c(p,0,0))  
 AR.bic[p] = AIC(AR.fit,k = log(length(y.t)))  
}  
par(mfrow=c(1,2))  
plot(AR.bic, type = "b", pch =19, main = "BIC", ylab = "", xlab = "p")  
abline(v = which.min(AR.bic), col=2, lty = 2) # AR(9)  
  
y\_t = y.t[10:length(y.t)]  
y\_t\_1 = y.t[9:(length(y.t)-1)]  
y\_t\_2 = y.t[8:(length(y.t)-2)]  
y\_t\_3 = y.t[7:(length(y.t)-3)]  
y\_t\_4 = y.t[6:(length(y.t)-4)]  
y\_t\_5 = y.t[5:(length(y.t)-5)]  
y\_t\_6 = y.t[4:(length(y.t)-6)]  
y\_t\_7 = y.t[3:(length(y.t)-7)]  
y\_t\_8 = y.t[2:(length(y.t)-8)]  
y\_t\_9 = x.t[1:(length(x.t)-9)]  
  
 x\_t = x.t[10:length(x.t)]  
x\_t\_1 = x.t[9:(length(x.t)-1)]  
x\_t\_2 = x.t[8:(length(x.t)-2)]  
x\_t\_3 = x.t[7:(length(x.t)-3)]  
x\_t\_4 = x.t[6:(length(x.t)-4)]  
x\_t\_5 = x.t[5:(length(x.t)-5)]  
x\_t\_6 = x.t[4:(length(x.t)-6)]  
x\_t\_7 = x.t[3:(length(x.t)-7)]  
x\_t\_8 = x.t[2:(length(x.t)-8)]  
x\_t\_9 = x.t[1:(length(x.t)-9)]  
  
y.data = cbind(y\_t\_1 = y\_t\_1, y\_t\_2= y\_t\_2, y\_t\_3 = y\_t\_3, y\_t\_4 = y\_t\_4,  
 y\_t\_5 = y\_t\_5, y\_t\_6= y\_t\_6, y\_t\_7 = y\_t\_7, y\_t\_8 = y\_t\_8,  
 y\_t\_9 = y\_t\_9)  
x.data = cbind(x\_t\_1 = x\_t\_1, x\_t\_2= x\_t\_2, x\_t\_3 = x\_t\_3, x\_t\_4 = x\_t\_4,  
 x\_t\_5 = x\_t\_5, x\_t\_6= x\_t\_6, x\_t\_7 = x\_t\_7, x\_t\_8 = x\_t\_8,  
 x\_t\_9 = x\_t\_9)  
  
# ADL 모형 BIC : p=5, q=1  
ADL.bic = matrix(0, ncol = 9, nrow = 9)  
for( p in 1:9){  
 for(q in 1:9){   
 ADL.fit = lm(y\_t ~ y.data[,1:p] + x.data[,1:q])   
 ADL.bic[p,q] = AIC(ADL.fit,k = log(length(y.t)))  
 }  
}  
colnames(ADL.bic) = c("q=1", "q=2", "q=3", "q=4", "q=5", "q=6", "q=7", "q=8", "q=9")  
row.names(ADL.bic) = c("p=1", "p=2", "p=3", "p=4", "p=5", "p=6", "p=7", "p=8", "p=9")  
ADL.bic; min(ADL.bic) # ADL(5,1)

## q=1 q=2 q=3 q=4 q=5 q=6 q=7  
## p=1 13232.15 13239.76 13245.36 13234.71 13241.76 13246.03 13252.76  
## p=2 13207.66 13213.54 13221.51 13214.31 13221.66 13227.57 13234.78  
## p=3 13202.57 13209.38 13216.42 13214.03 13221.73 13227.97 13235.59  
## p=4 13163.22 13170.84 13178.62 13186.57 13194.43 13201.83 13209.75  
## p=5 13154.38 13162.54 13170.57 13178.20 13180.37 13188.54 13196.70  
## p=6 13161.27 13169.41 13177.27 13185.00 13186.74 13194.67 13202.68  
## p=7 13164.47 13172.65 13180.64 13187.84 13190.22 13197.98 13205.86  
## p=8 13168.50 13176.67 13184.76 13191.70 13195.31 13202.92 13210.92  
## p=9 13162.36 13170.35 13177.73 13185.73 13186.40 13194.57 13200.30  
## q=8 q=9  
## p=1 13249.65 13232.99  
## p=2 13233.08 13217.56  
## p=3 13234.46 13219.90  
## p=4 13211.29 13198.34  
## p=5 13199.10 13189.20  
## p=6 13204.39 13193.90  
## p=7 13209.54 13200.19  
## p=8 13216.77 13208.35  
## p=9 13208.35 13208.35

## [1] 13154.38

## 1-step ahead forecasting  
AR9.fore = c(); ADL51.fore = c()  
for(i in 3106:(length(y.t)-1)){  
 train.data = data.frame(y.t = y.t[1:i], x.t = x.t[1:i])  
   
 y\_t = train.data[10:nrow(train.data),1]  
 y\_t\_1 = train.data[9:(nrow(train.data)-1),1]  
 y\_t\_2 = train.data[8:(nrow(train.data)-2),1]  
 y\_t\_3 = train.data[7:(nrow(train.data)-3),1]  
 y\_t\_4 = train.data[6:(nrow(train.data)-4),1]  
 y\_t\_5 = train.data[5:(nrow(train.data)-5),1]  
 y\_t\_6 = train.data[4:(nrow(train.data)-6),1]  
 y\_t\_7 = train.data[3:(nrow(train.data)-7),1]  
 y\_t\_8 = train.data[2:(nrow(train.data)-8),1]  
 y\_t\_9 = train.data[1:(nrow(train.data)-9),1]  
   
 x\_t = train.data[10:nrow(train.data),2]  
 x\_t\_1 = train.data[9:(nrow(train.data)-1),2]  
 x\_t\_2 = train.data[8:(nrow(train.data)-2),2]  
 x\_t\_3 = train.data[7:(nrow(train.data)-3),2]  
 x\_t\_4 = train.data[6:(nrow(train.data)-4),2]  
 x\_t\_5 = train.data[5:(nrow(train.data)-5),2]  
 x\_t\_6 = train.data[4:(nrow(train.data)-6),2]  
 x\_t\_7 = train.data[3:(nrow(train.data)-7),2]  
 x\_t\_8 = train.data[2:(nrow(train.data)-8),2]  
 x\_t\_9 = train.data[1:(nrow(train.data)-9),2]  
   
 # AR(9)  
 AR9.fit = lm(y\_t ~ y\_t\_1 + y\_t\_2 + y\_t\_3 + y\_t\_4 +   
 y\_t\_5 + y\_t\_6 + y\_t\_7 + y\_t\_8 + y\_t\_9)  
 AR9.fore[i-3105] = sum(AR9.fit$coef\*c(1, y\_t[length(y\_t)],   
 y\_t\_1[length(y\_t)],   
 y\_t\_2[length(y\_t)],   
 y\_t\_3[length(y\_t)],  
 y\_t\_4[length(y\_t)],  
 y\_t\_5[length(y\_t)],  
 y\_t\_6[length(y\_t)],  
 y\_t\_7[length(y\_t)],  
 y\_t\_8[length(y\_t)]))  
 # ADL(5,1)  
 ADL.fit1 = lm(y\_t ~   
 y\_t\_1 + y\_t\_2 +y\_t\_3 + y\_t\_4 + y\_t\_5 +   
 x\_t );  
 ADL51.fore[i-3105] = sum(ADL.fit1$coef\*c(1, y\_t[length(y\_t)],   
 y\_t\_1[length(y\_t)],   
 y\_t\_2[length(y\_t)],   
 y\_t\_3[length(y\_t)],  
 y\_t\_4[length(y\_t)],  
 x\_t[length(y\_t)] ))  
}  
  
# 1. VKOSPI가 KOSPI 5분 실현변동성을 GRANGER CAUSE하는가  
grangertest(y.t~x.t)

## Granger causality test  
##   
## Model 1: y.t ~ Lags(y.t, 1:1) + Lags(x.t, 1:1)  
## Model 2: y.t ~ Lags(y.t, 1:1)  
## Res.Df Df F Pr(>F)  
## 1 3554   
## 2 3555 -1 0.0184 0.8922

# 2. 예측력 비교  
# AR(8) VS ADL(5,1)   
# AR8.fore VS ADL51.fore  
# MAE  
AR9.MAE = mean(abs(AR9.fore - y.t[3107:length(y.t)]))  
ADL51.MAE = mean(abs(ADL51.fore - y.t[3107:length(y.t)]))  
AR9.MAE < ADL51.MAE # AR(9) 모형이 MAE 값이 더 작으므로

## [1] TRUE

# MSE  
AR9.MSE = mean((AR9.fore - y.t[3107:length(y.t)])^2)  
ADL51.MSE = mean((ADL51.fore - y.t[3107:length(y.t)])^2)  
AR9.MSE < ADL51.MSE # AR(9) 모형이 MSE 값이 더 작으므로

## [1] TRUE

result = matrix(c(AR9.MAE, ADL51.MAE, AR9.MSE, ADL51.MSE), nrow = 2, byrow = T)  
row.names(result) = c("MAE", "MSE")  
colnames(result) = c("AR9", "ADL(5,1)")  
result # AR(1) 모형이 더 잘 예측

## AR9 ADL(5,1)  
## MAE 0.6205788 0.6765098  
## MSE 0.9326661 1.1859905

