Stationary

NGUYEN QUANG DONG

November 9, 2022

library(foreign)  
library(dynlm)

## Loading required package: zoo

##   
## Attaching package: 'zoo'

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

library(sandwich)  
library(car)

## Loading required package: carData

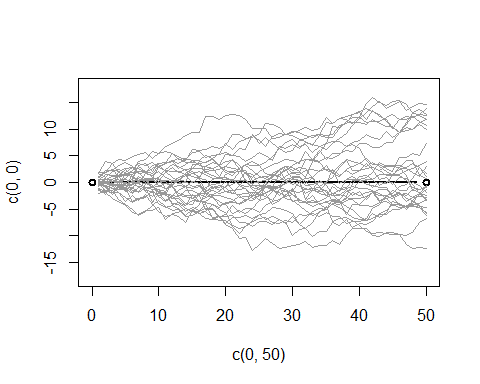
library(carData)  
library(zoo)  
library(ggplot2)  
library(forecast)

## Registered S3 method overwritten by 'quantmod':  
## method from  
## as.zoo.data.frame zoo

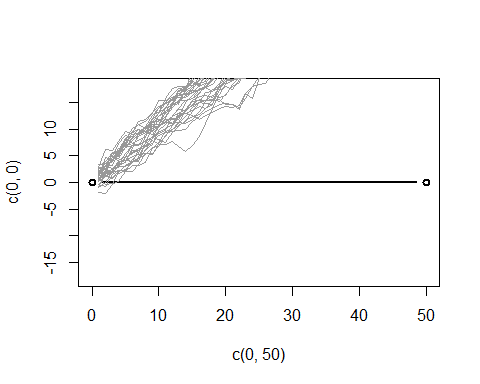
library(tseries)  
library(urca)  
library(dynlm)

## TẠO Random Walk

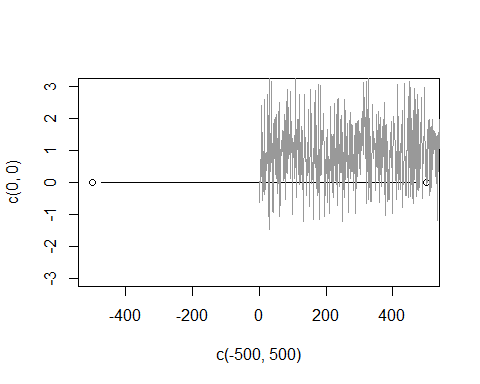
# Tạo trục tọa độ x= (0,50), y=(-18,18)  
plot(c(0,50),c(0,0),type="b", lwd=2,ylim=c(-18,18))  
#lopp over draw  
for (r in 1:30){  
 # i.i.d stadard normal shock  
 e=rnorm(50)  
 # Random Walk as cummulative sum shocks  
 y=ts(cumsum(e))  
 # Add line to graph  
 lines(y,col=gray(.6))  
}

 ## TẠO Random Walk with Drift

plot(c(0,50),c(0,0),type="b", lwd=2,ylim=c(-18,18))  
#lopp over draw  
for (r in 1:30){  
 # i.i.d stadard normal shock  
 e=rnorm(50)  
 # Random Walk as cummulative sum shocks  
 y=ts(cumsum(1+e))  
 # Add line to graph  
 lines(y,col=gray(.6))  
}

 ## SAI PHÂN CỦA Random Walk with Drift

# Tao 1000 twf phan bo chuan hoa  
u=rnorm(1000, 0, 1)  
plot(c(-500,500), c(0,0),type="b",ylim=c(-3,3))  
#lopp over draw  
for (r in 1:1000){  
 # i.i.d stadard normal shock  
 # Random Walk as cummulative sum shocks, with drift  
 y=ts(cumsum(1+u))  
 #First Difference  
 Dy=diff(y)  
 # Add line to graph  
 lines(Dy,col=gray(.6))  
}

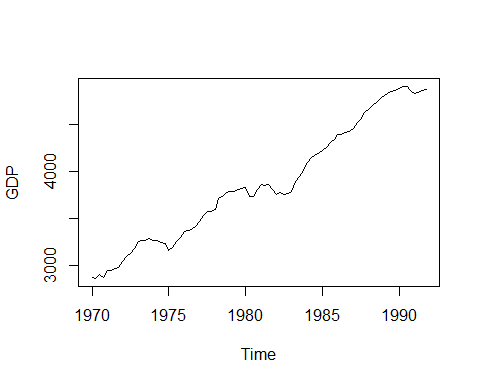


## NHẬP SỐ LIỆU VÀ TẠO TIME SERIER

CH12BT20=read.table("D:/dataR/ch12bt20.txt", header=TRUE)  
# CH12BT20   
GDP=CH12BT20$GDP  
GDP=ts(GDP, start=c(1970,1),end=c(1991,4),frequency=4)  
# GDP  
PDI=CH12BT20$PDI  
PDI=ts(PDI, start=c(1970,1),end=c(1991,4),frequency=4)  
# PDI

## NHẬN DẠNG CHUỖI GDP: XÁC ĐỊNH CÓ XU THÊ, HỆ SỐ CHẶN?

plot(GDP)



## KIỂM ĐỊNH ADF CHO GDP-KIỂM ĐỊNH BƯỚC NGẪU NHIÊN CHO GDP - MO HINH

# Cách 1  
reg1=dynlm(d(GDP)~L(GDP)+L(d(GDP))+L(d(GDP),2)+L(d(GDP),3)+L(d(GDP),4)+trend(GDP))  
  
summary(reg1)

##   
## Time series regression with "ts" data:  
## Start = 1971(2), End = 1991(4)  
##   
## Call:  
## dynlm(formula = d(GDP) ~ L(GDP) + L(d(GDP)) + L(d(GDP), 2) +   
## L(d(GDP), 3) + L(d(GDP), 4) + trend(GDP))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -109.042 -14.433 5.334 17.139 90.714   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 300.97181 105.44016 2.854 0.00555 \*\*  
## L(GDP) -0.10624 0.03884 -2.735 0.00775 \*\*  
## L(d(GDP)) 0.36594 0.10798 3.389 0.00111 \*\*  
## L(d(GDP), 2) 0.10127 0.11257 0.900 0.37117   
## L(d(GDP), 3) 0.03813 0.11270 0.338 0.73606   
## L(d(GDP), 4) 0.11565 0.11102 1.042 0.30087   
## trend(GDP) 10.37163 3.84508 2.697 0.00860 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 33.06 on 76 degrees of freedom  
## Multiple R-squared: 0.2119, Adjusted R-squared: 0.1497   
## F-statistic: 3.406 on 6 and 76 DF, p-value: 0.004974

# Kiểm định ADF BIẾN KHÔNG XU THẾ, KHÔNG CÓ HỆ SỐ CHẶN   
adf.test(GDP)

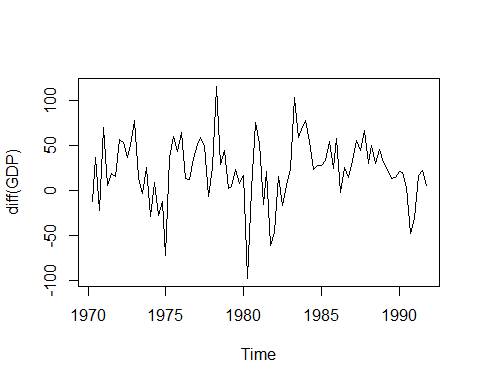
##   
## Augmented Dickey-Fuller Test  
##   
## data: GDP  
## Dickey-Fuller = -2.7352, Lag order = 4, p-value = 0.2738  
## alternative hypothesis: stationary

# DẠNG TỔNG QUÁT: ur.df(y, type = c("none", "drift", "trend"), lags = 1, selectlags = c("Fixed", "AIC", "BIC"))  
  
  
reg2=(ur.df(GDP,type=c("trend"), 1))  
summary(reg2)

##   
## ###############################################   
## # Augmented Dickey-Fuller Test Unit Root Test #   
## ###############################################   
##   
## Test regression trend   
##   
##   
## Call:  
## lm(formula = z.diff ~ z.lag.1 + 1 + tt + z.diff.lag)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -115.183 -15.702 5.961 18.990 91.566   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 234.97291 98.58764 2.383 0.019465 \*   
## z.lag.1 -0.07866 0.03551 -2.215 0.029513 \*   
## tt 1.89220 0.87917 2.152 0.034317 \*   
## z.diff.lag 0.35579 0.10269 3.465 0.000847 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 33.68 on 82 degrees of freedom  
## Multiple R-squared: 0.1526, Adjusted R-squared: 0.1216   
## F-statistic: 4.923 on 3 and 82 DF, p-value: 0.003406  
##   
##   
## Value of test-statistic is: -2.2153 6.2201 2.4724   
##   
## Critical values for test statistics:   
## 1pct 5pct 10pct  
## tau3 -4.04 -3.45 -3.15  
## phi2 6.50 4.88 4.16  
## phi3 8.73 6.49 5.47

## KIỂM ĐỊNH TÍNH DỪNG CHO SAI PHÂN

# Nhan dang sai phan  
  
plot(diff(GDP))



# Unit Test for diff(GDP)  
reg3=ur.df(diff(GDP),type=c("drift"))  
  
summary(reg3)

##   
## ###############################################   
## # Augmented Dickey-Fuller Test Unit Root Test #   
## ###############################################   
##   
## Test regression drift   
##   
##   
## Call:  
## lm(formula = z.diff ~ z.lag.1 + 1 + z.diff.lag)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -117.729 -16.834 2.039 17.747 94.516   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 14.08372 4.78000 2.946 0.00418 \*\*   
## z.lag.1 -0.61172 0.12776 -4.788 7.37e-06 \*\*\*  
## z.diff.lag -0.09282 0.10915 -0.850 0.39758   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 34.42 on 82 degrees of freedom  
## Multiple R-squared: 0.3428, Adjusted R-squared: 0.3268   
## F-statistic: 21.39 on 2 and 82 DF, p-value: 3.351e-08  
##   
##   
## Value of test-statistic is: -4.7879 11.4634   
##   
## Critical values for test statistics:   
## 1pct 5pct 10pct  
## tau2 -3.51 -2.89 -2.58  
## phi1 6.70 4.71 3.86

## TINH ACF, PACF & CORELOGRAM

# So p lon nhat tính bang maxp ; p khong vuot qua n.  
maxp= (10\*log10(length(GDP)/1))  
maxp= as.integer(maxp)  
maxp

## [1] 19

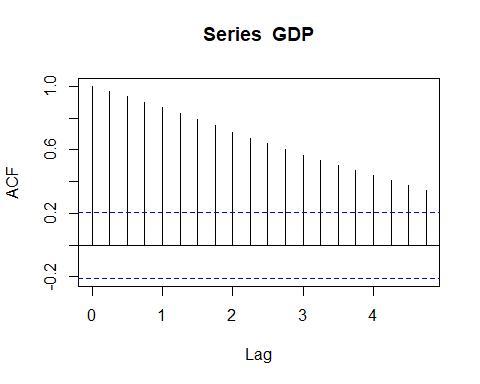
# Tinh ACF  
print("Tinh ACF")

## [1] "Tinh ACF"

acf(GDP, lag.max = maxp, plot = FALSE)

##   
## Autocorrelations of series 'GDP', by lag  
##   
## 0.00 0.25 0.50 0.75 1.00 1.25 1.50 1.75 2.00 2.25 2.50 2.75 3.00   
## 1.000 0.969 0.935 0.901 0.866 0.830 0.791 0.752 0.713 0.675 0.638 0.601 0.565   
## 3.25 3.50 3.75 4.00 4.25 4.50 4.75   
## 0.532 0.500 0.468 0.437 0.405 0.375 0.344

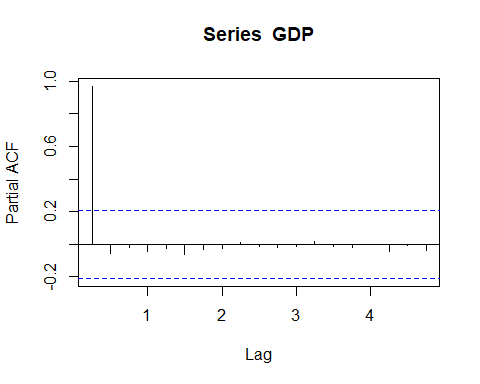
# Ve ACF  
acf(GDP, lag.max = maxp ,type=c("correlation"),plot = TRUE,na.action = na.contiguous,demean = TRUE)



# Tinh PACF  
pacf(GDP, lag.max = maxp , plot = FALSE)

##   
## Partial autocorrelations of series 'GDP', by lag  
##   
## 0.25 0.50 0.75 1.00 1.25 1.50 1.75 2.00 2.25 2.50 2.75   
## 0.969 -0.058 -0.020 -0.045 -0.024 -0.062 -0.029 -0.024 0.009 -0.010 -0.020   
## 3.00 3.25 3.50 3.75 4.00 4.25 4.50 4.75   
## -0.012 0.020 -0.012 -0.021 -0.001 -0.041 -0.005 -0.038

#Ve PACF  
pacf(GDP,lag.max = maxp,plot=TRUE,na.action = na.contiguous)



Box.test(GDP, lag = 3, type = c("Box-Pierce"), fitdf = 0)

##   
## Box-Pierce test  
##   
## data: GDP  
## X-squared = 231.08, df = 3, p-value < 2.2e-16

Box.test(GDP, lag = 3, type = c("Ljung-Box"), fitdf = 0)

##   
## Box-Ljung test  
##   
## data: GDP  
## X-squared = 241.72, df = 3, p-value < 2.2e-16