INTRODUCCIÓN A SERIES DE TIEMPO

LUIS ORTIZ-CEVALLOS

2021-09-21

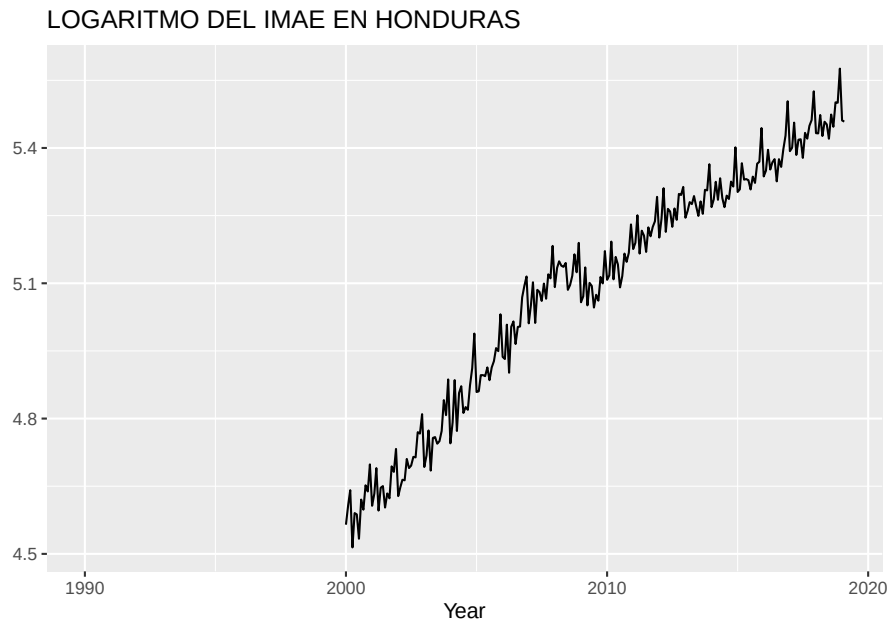
Contents

Chapter 1

¿Qué es una de serie de tiempo?

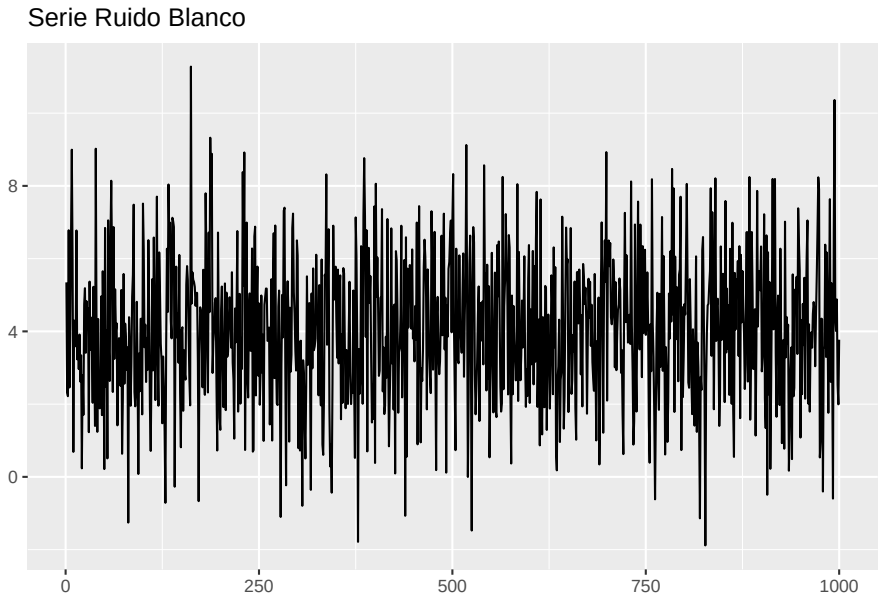
```
library("readr")
library("xts")
library("zoo")
library("astsa")
library("forecast")
library("ggplot2")
library("forecast")
library("ggfortify")
library("stargazer")
library("urca")
library("dynlm")
library("scales")
library("quantmod")
TRIM<-as.xts(read.zoo("FINAL_HN.csv", index.column = 1, sep = ";", header=TRUE, format = "%d/%m/%
MES<-as.xts(read.zoo("MES_HN.csv", index.column = 1, sep = ";", header=TRUE, format = "%d/%m/%Y")
IMAE<-MES$IMAE
P<-ggplot2::autoplot(log(IMAE))+xlab("Year")+
ggtitle("LOGARITMO DEL IMAE EN HONDURAS")
```

```
P
```



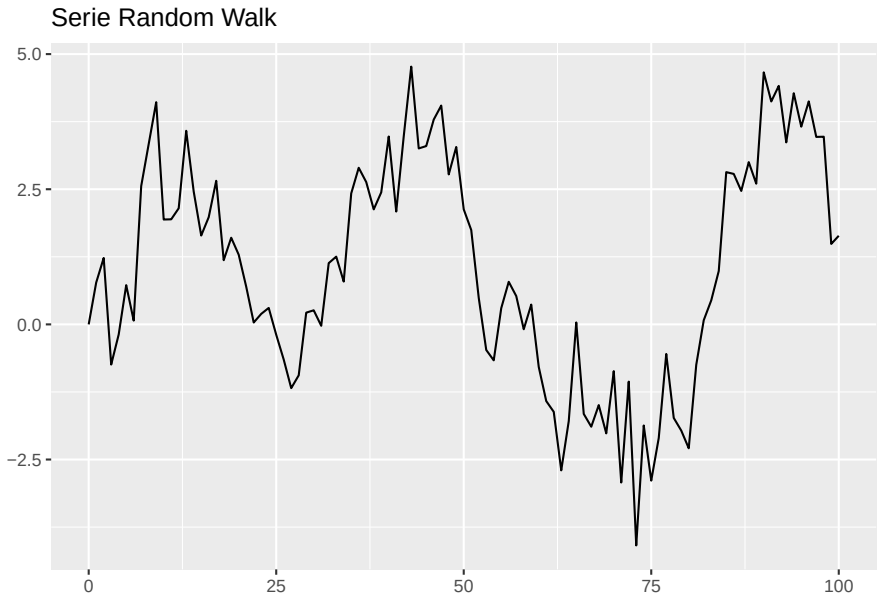

```
X_WN<-arima.sim(list(order=c(0,0,0)), n=1000, mean=4, sd=2)
autoplot(X_WN)+
ggtitle("Serie Ruido Blanco")
```

1.1 Serie Ruido Blanco (WN)

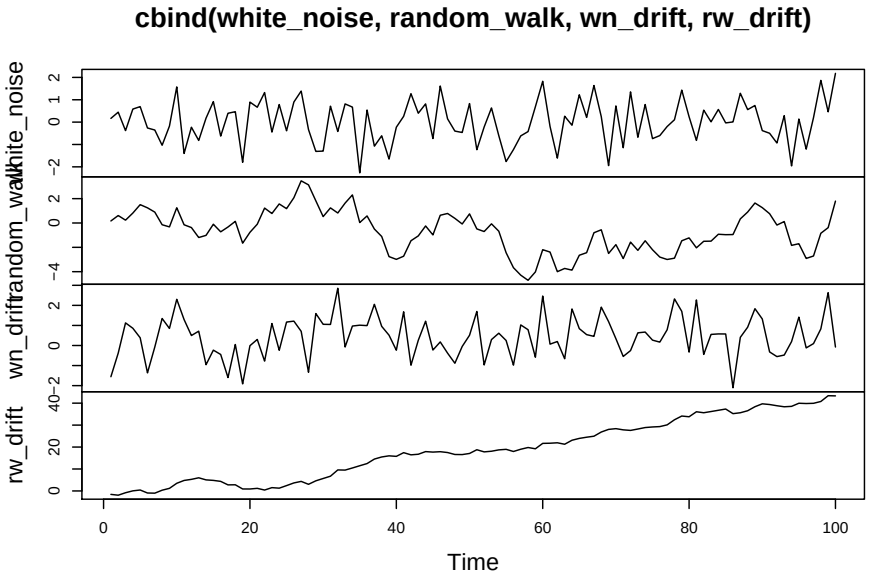



```
X_RW<-arima.sim(list(order=c(0,1,0)), n=100)
autoplot(X_RW)+
ggtitle("Serie Random Walk")
```

1.2 Serie Random Walk (RW)



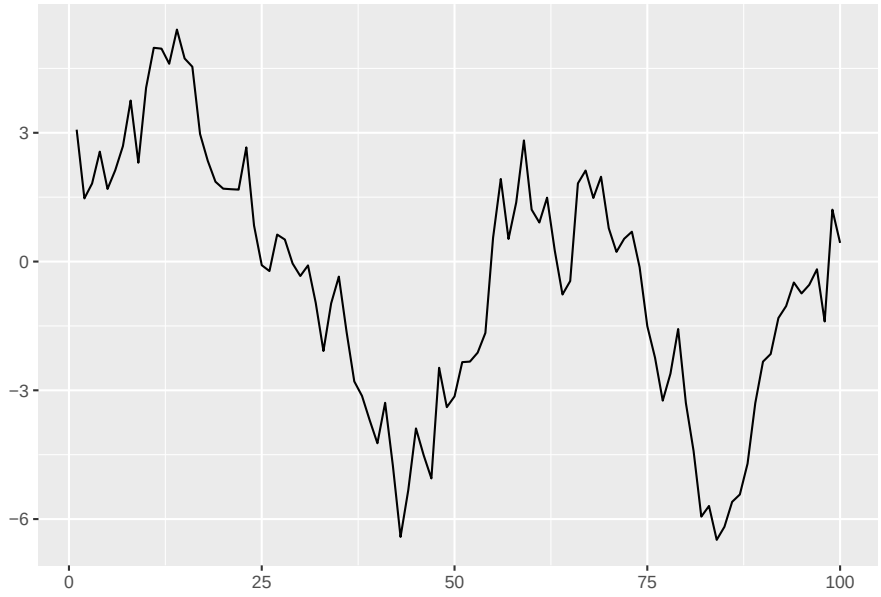

```
white_noise <- arima.sim(list(order = c(0, 0, 0)), n=100)
random_walk <- cumsum(white_noise)
wn_drift <- arima.sim(list(order = c(0, 0, 0)), n=100, mean=0.4)
rw_drift <- cumsum(wn_drift)
plot.ts(cbind(white_noise, random_walk, wn_drift, rw_drift))
```



1.3 Proceso ARMA

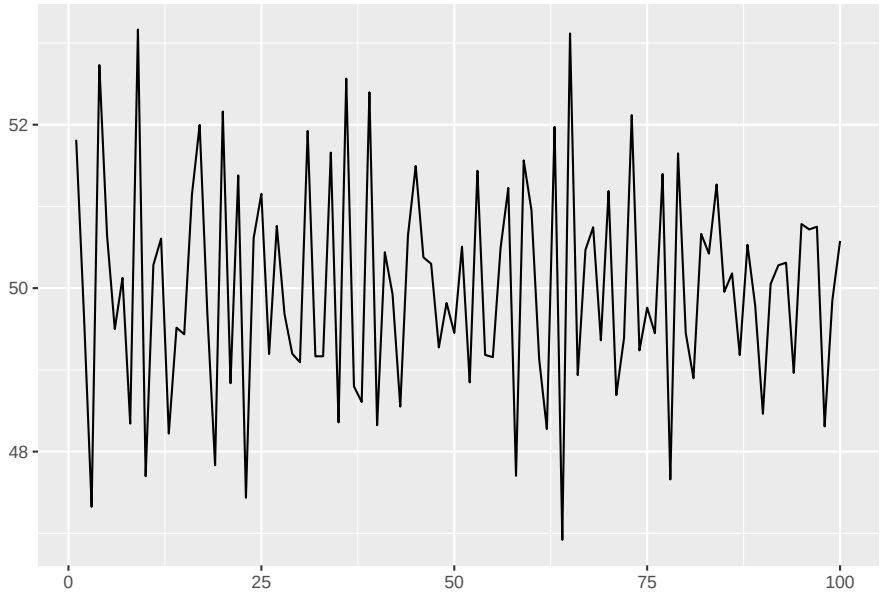
Simulando un proceso AR(1)

```
X_AR1<-arima.sim(list(order=c(1,0,0), ar=c(0.90)), n=100)
autoplot(X_AR1)
```



Simulando un proceso AR(2)

```
X_MA1<-arima.sim(list(order=c(0,0,1), ma=c(-0.98)), n=100)+50
autoplot(X_MA1)
```



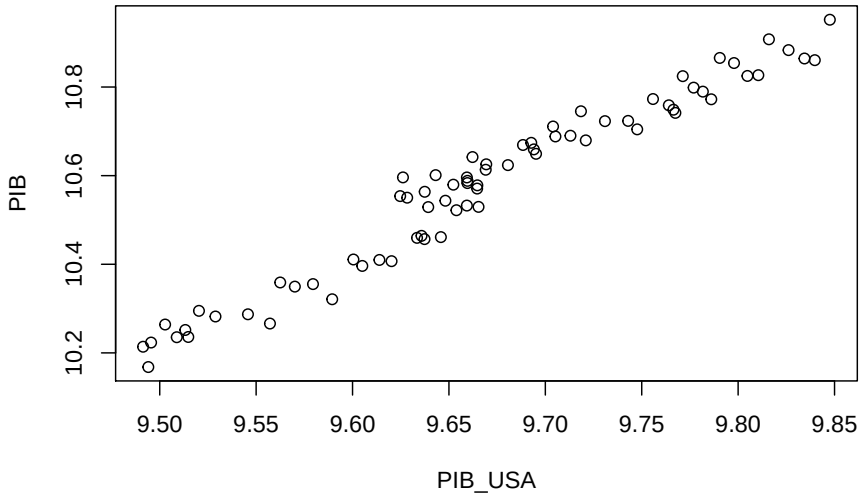
Correlación entre el nivel del PIB de Honduras y el de USA

```
USA<-coredata(log(TRIM$PIB_USA["2001-01-01/"]))
HN<-coredata(log(TRIM$PIB["2001-01-01/"]))
cor(USA,HN)
```

```
##                PIB
## PIB_USA 0.9775886
```

Scatter plot

```
plot(cbind(USA, HN))
```



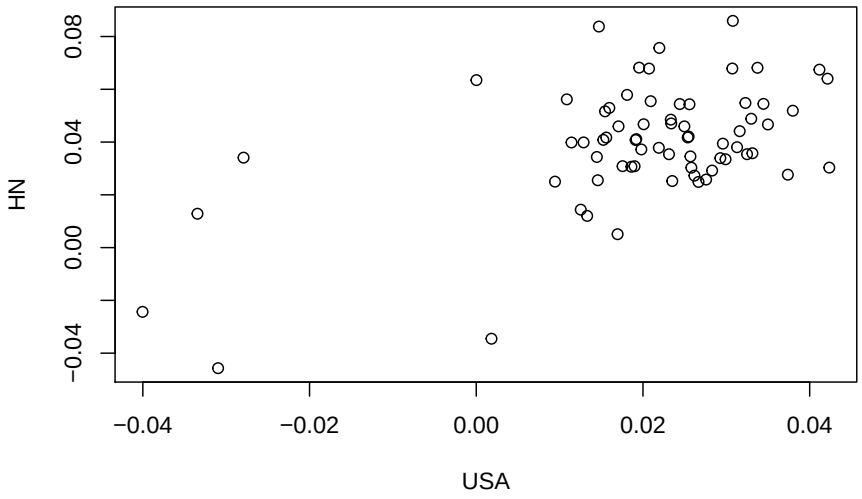
Correlación entre el la tasa de crecimiento del PIB de Honduras y el de USA

```
USA<-coredata(diff(USA, lag=4))
HN<-coredata(diff(HN, lag=4))
cor(USA,HN)
```

```
##                PIB
## PIB_USA 0.5405966
```

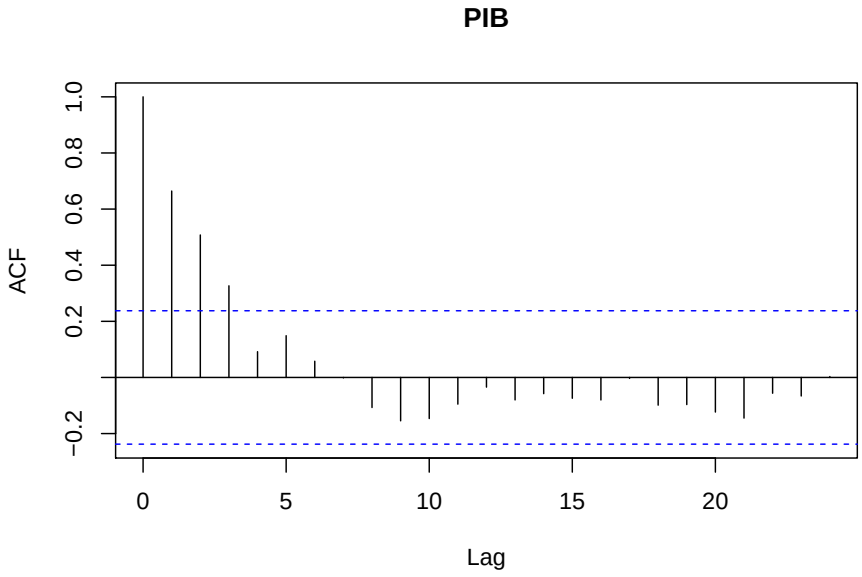
Scatter plot

```
plot(USA, HN)
```



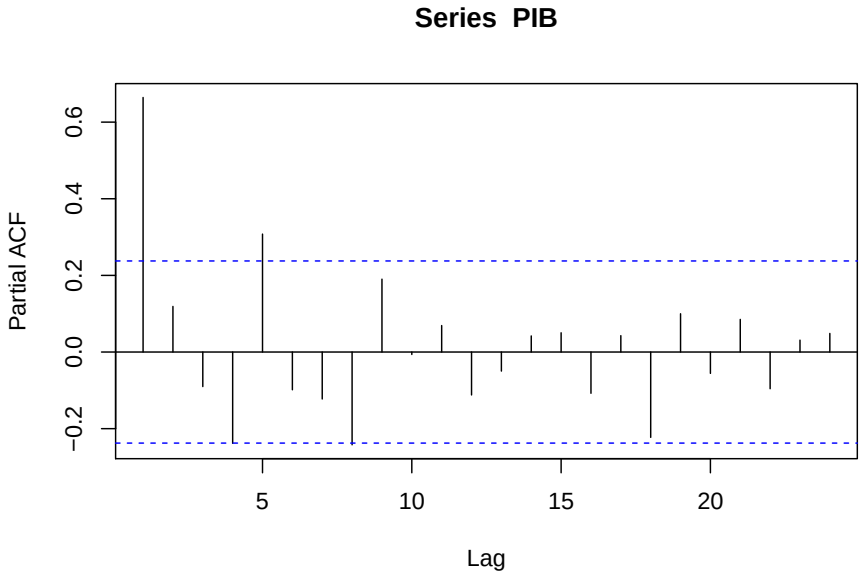
Función de autocorrelación del PIB de Honduras

```
PIB<-as.ts(HN)
acf(PIB, lag.max = 24, plot=TRUE)
```



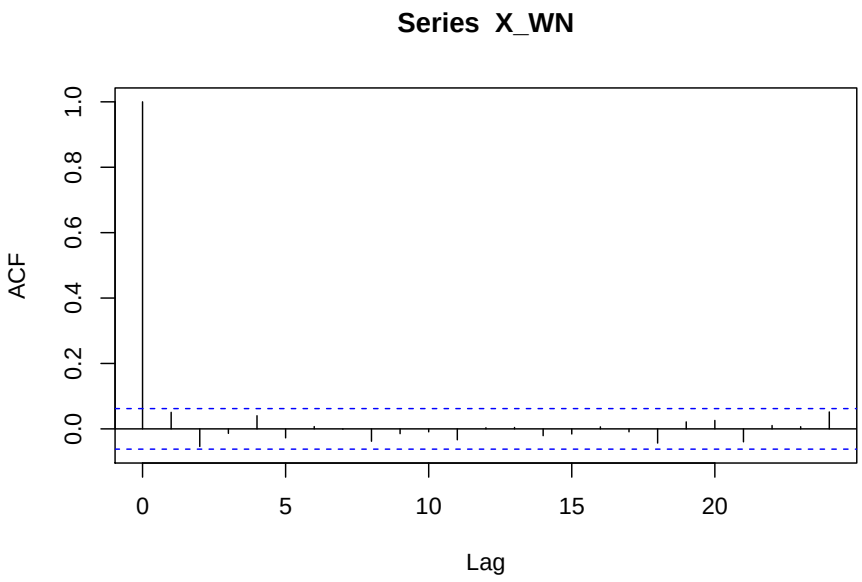
Función de autocorrelación parcial del PIB de Honduras

```
PIB<-as.ts(HN)
pacf(PIB, lag.max = 24, plot=TRUE)
```



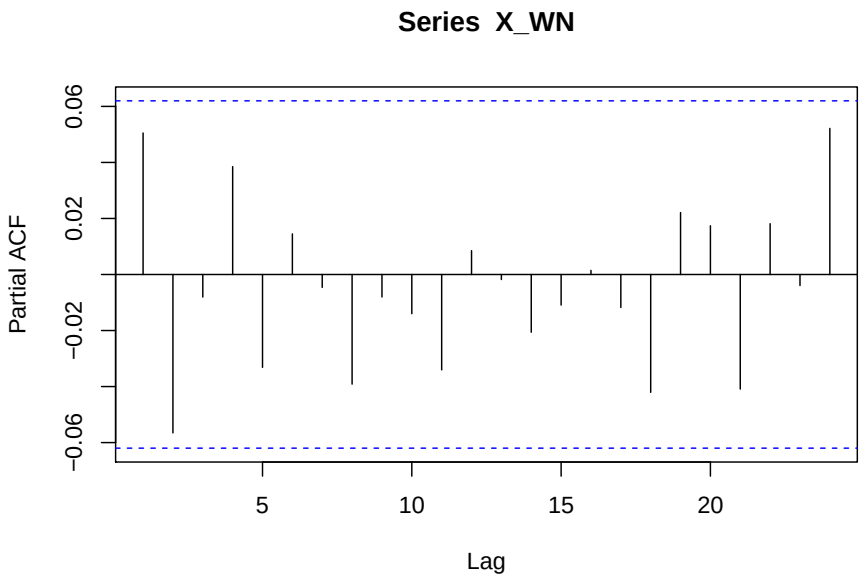
Función de autocorrelación de un proceso ruído blanco

```
acf(X_WN, lag.max = 24, plot=TRUE)
```



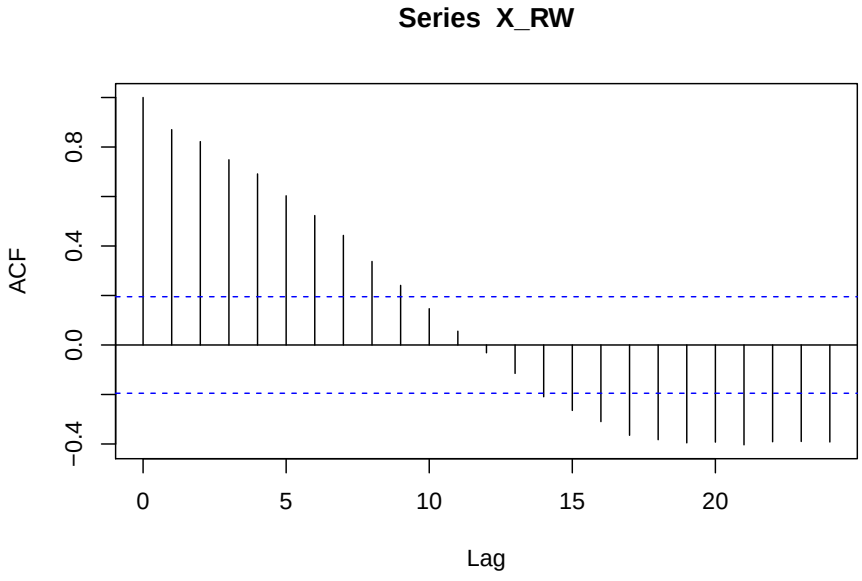
Función de autocorrelación parcial de un proceso ruído blanco

```
pacf(X_WN, lag.max = 24, plot=TRUE)
```



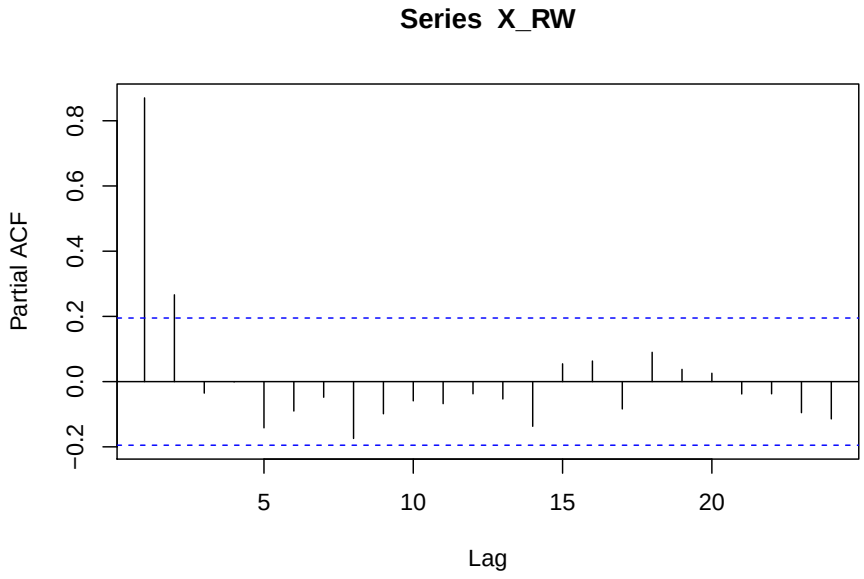
Función de autocorrelación de un proceso RW

```
acf(X_RW, lag.max = 24, plot=TRUE)
```



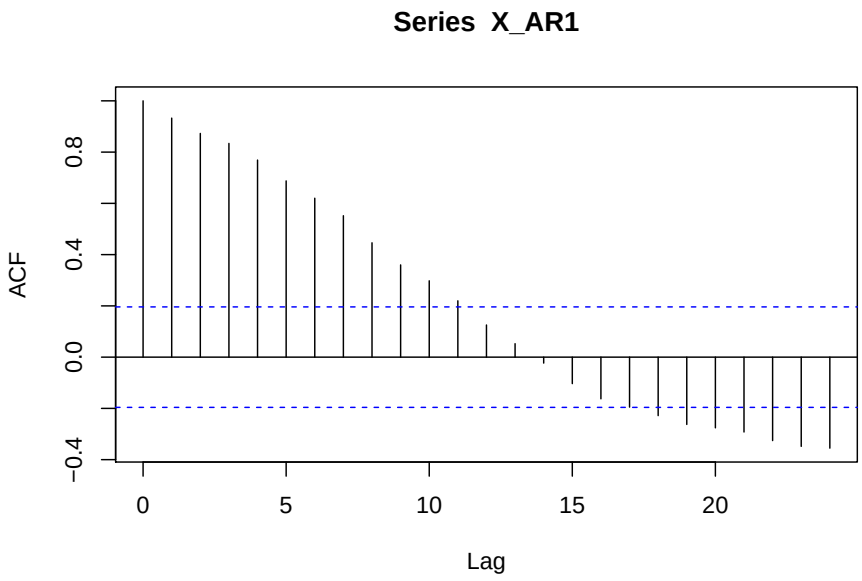
Función de autocorrelación parcial de un proceso RW

```
pacf(X_RW, lag.max = 24, plot=TRUE)
```



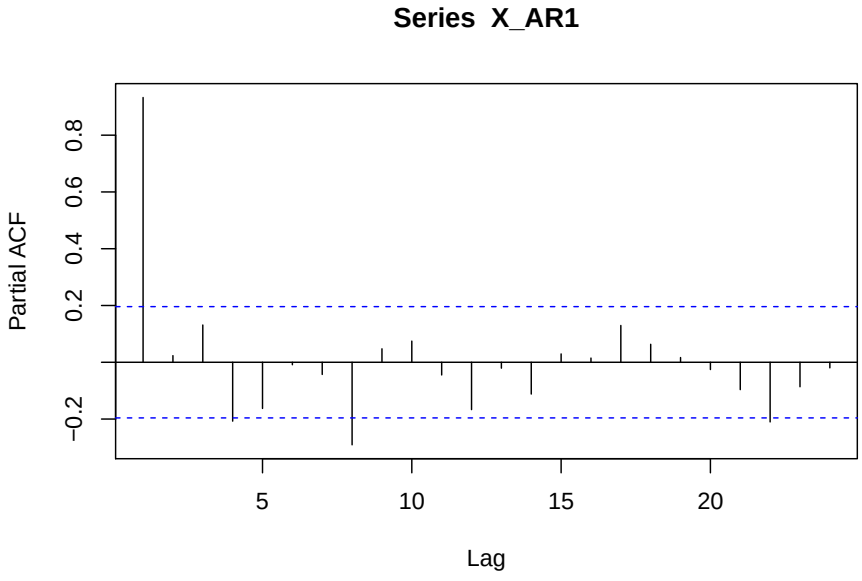
Función de autocorrelación de un proceso AR(1)

```
acf(X_AR1, lag.max = 24, plot=TRUE)
```



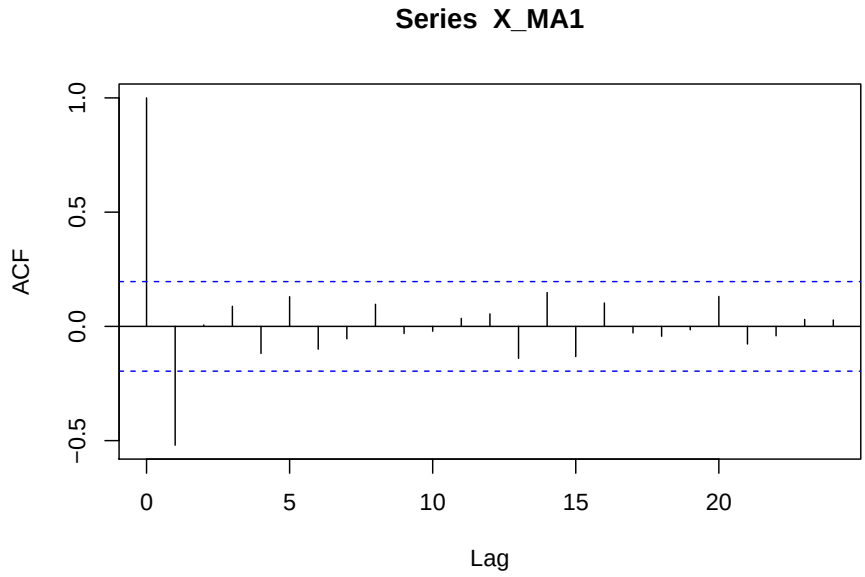
Función de autocorrelación parcial de un proceso AR(1)

```
pacf(X_AR1, lag.max = 24, plot=TRUE)
```



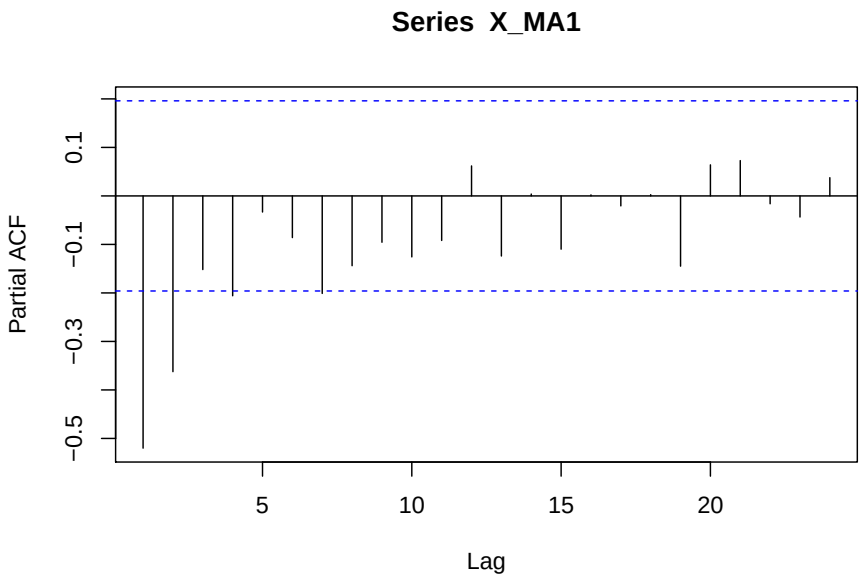
Función de autocorrelación de un proceso MA(1)

```
acf(X_MA1, lag.max = 24, plot=TRUE)
```



Función de autocorrelación parcial de un proceso MA(1)

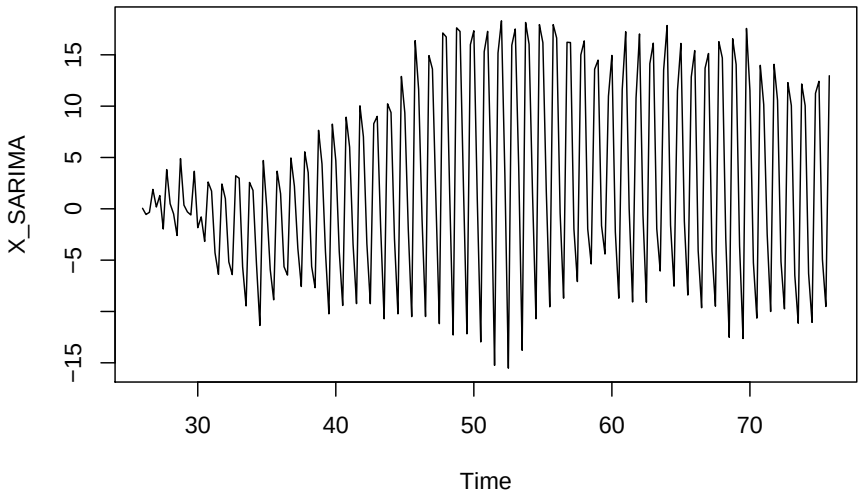
```
pacf(X_MA1, lag.max = 24, plot=TRUE)
```



1.4 Simulación de procesos

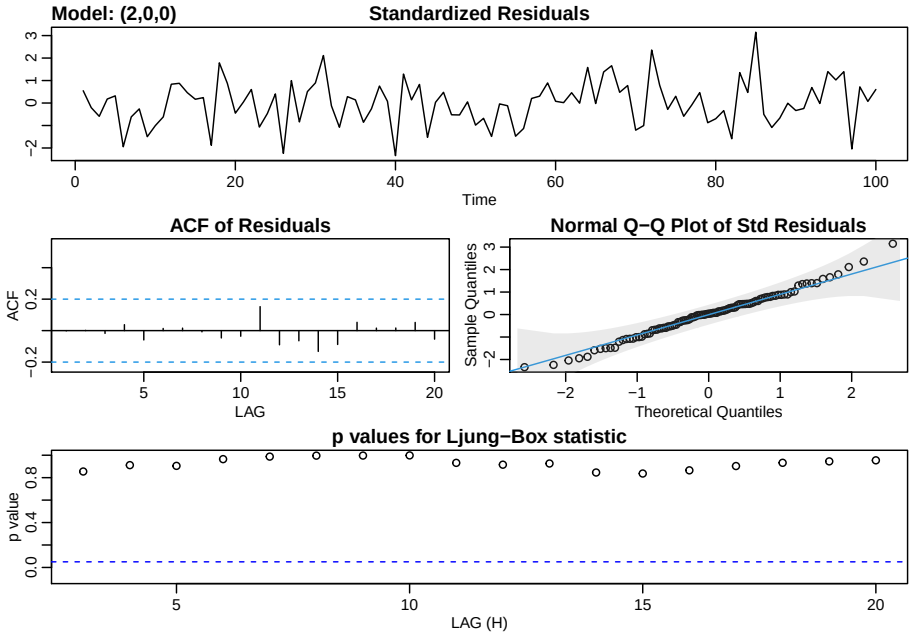
###Estimación de un procesos SARIMA(1,0,1,1,1,1)

```
model    <- Arima(ts(rnorm(100),freq=4), order=c(1,0,1), seasonal=c(1,1,1),
            fixed=c(phi=0.0, theta=-0.0, Phi=0.0, Theta=-0.0))
X_SARIMA<- simulate(model, nsim=200)
plot(X_SARIMA)
```




```
x<-arima.sim(list(order=c(0,0,2), ma=c(1.5,-0.75)), n=100)+50
x_fit<-sarima(x, p=2, d=0, q=0)
```

1.4.1 Estimación de un proceso ARIMA


Model: (2,0,0)
Standardized Residuals
Time
ACF of Residuals
ACF
LAG
Normal Q−Q Plot of Std Residuals
Sample Quantiles
Theoretical Quantiles
p values for Ljung−Box statistic
p value
LAG (H)

```
x_fit$ttable
```

Chapter 2

Pronósticos

2.1 Modelos introductorios

Pronósticos Naive del IMAE de Honduras vs data observada

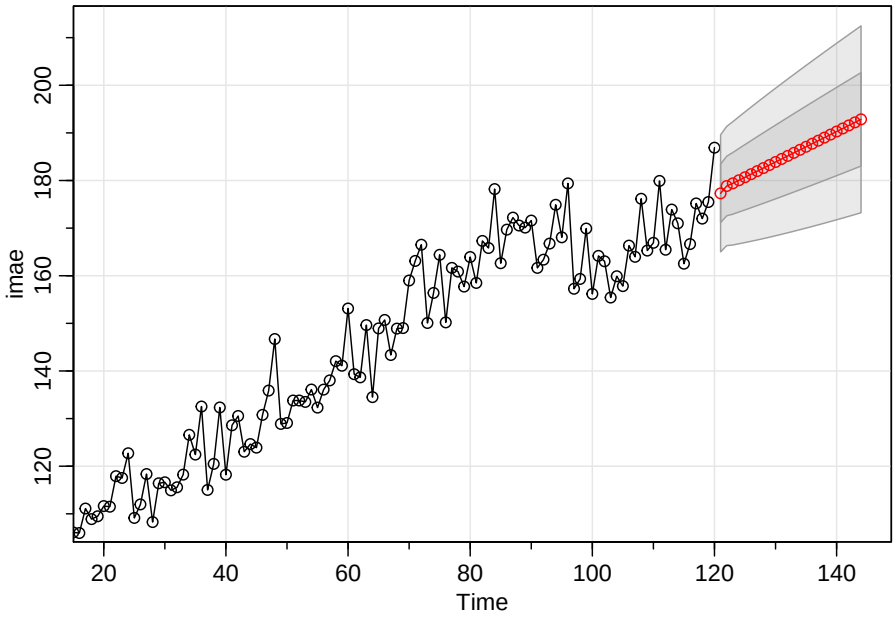
```
imae<-log(MES$IMAE["2001-01-01/2010-12-01"])
IMAE_NAIVE<-naive(imae)
imaef<-ts(fitted(IMAE_NAIVE), frequency=12, start=c(2001/01/01))
imaef<-as.xts(imaef)
autoplot(ts(cbind(imae, imaef), start = c(2001/01/01), frequency = 12 ),
         facets = FALSE)+xlab("Years")
```

Pronósticos del IMAE de Honduras 24 meses en adelante a partir de un proceso SARIMA(1,1,1)

```
imae<-IMAE["2001-01-01/2010-12-01"]
imaef<-IMAE["/2012-12-01"]
resultado<-sarima.for(imae, n.ahead=24,1,1,1)
```

Table 2.1: Regresión entre el nivel del PIB de Honduras con respecto al de USA

term	estimate	std.error	statistic	p.value
(Intercept)	-9.6056	0.4685	-20.5041	0
log(TRIM$PIB_USA)	2.0873	0.0485	43.0384	0



2.2 Modelos para hacer pronósticos del PIB de Honduras

Modelo de regresión

```
library(knitr)
library(dplyr)
library(broom)
library(AER)
TRIM<-as.xts(read.zoo("FINAL_HN_P.csv", index.column = 1, sep = ";", header=TRUE, format = "%d/%m
M.ols <- lm(log(TRIM$PIB) ~ log(TRIM$PIB_USA))
kable(tidy(M.ols), digits=4, align='c',caption="Regresión entre el nivel del PIB de Honduras con
```

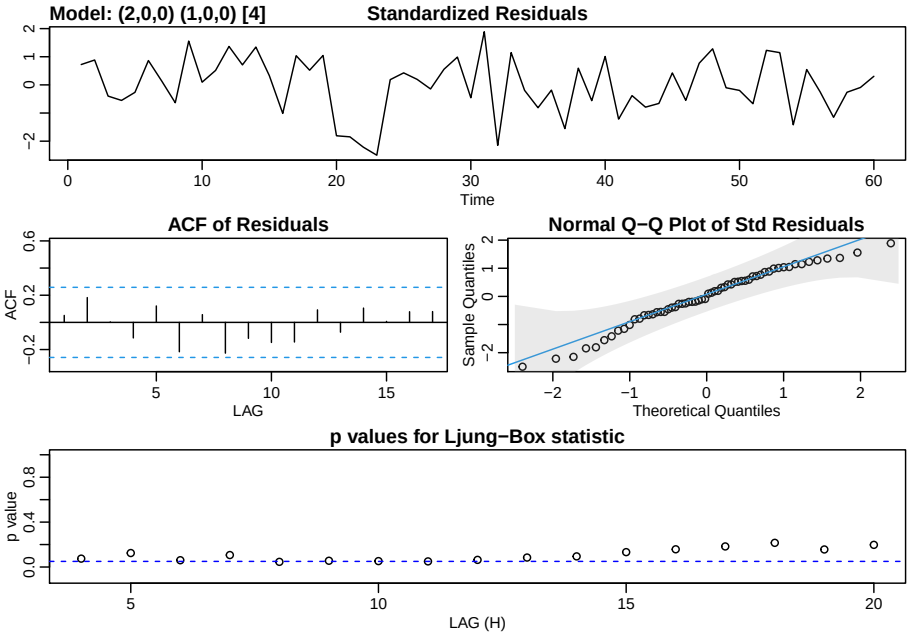
Modelo de regresión para el PIB de Honduras

```
INDEX  <-factor(index(TRIM))
dummies<-model.matrix(~INDEX)
```

```
TRIM   <-merge(TRIM, dummies, join="left")
Y      <-window(diff(log(TRIM$PIB), lag=4)*100, start="2004-03-01", end="2018-12-01")
Y_USA  <-window(diff(log(TRIM$PIB_USA), lag=4)*100, start="2004-03-01", end="2018-12-0
DUM_HN <-window(TRIM[, c("INDEX2005.09.01", "INDEX2006.12.01", "INDEX2008.06.01")], sta
i_HN   <-window(diff(TRIM$TASA_P, lag=1)*100, start="2004-03-01", end="2018-12-01")
REG_HN <- merge(DUM_HN, Y_USA, join="left")
REG_HN <- merge(REG_HN, i_HN,  join="left")
PIB_HN <-sarima(Y, 2,0,0,P=1, D=0, Q=0, 4, xreg=REG_HN)
```

```
## initial  value 0.542818
## iter   2 value 0.399627
## iter   3 value 0.369226
## iter   4 value 0.292117
## iter   5 value 0.266433
## iter   6 value 0.252289
## iter   7 value 0.225647
## iter   8 value 0.225239
## iter   9 value 0.217205
## iter  10 value 0.210556
## iter  11 value 0.209208
## iter  12 value 0.204386
## iter  13 value 0.204299
## iter  14 value 0.204282
## iter  15 value 0.204281
## iter  16 value 0.204281
## iter  17 value 0.204281
## iter  18 value 0.204281
## iter  19 value 0.204281
## iter  19 value 0.204281
## iter  19 value 0.204281
## final  value 0.204281
## converged
## initial  value 0.186966
## iter   2 value 0.186057
## iter   3 value 0.185621
## iter   4 value 0.185490
## iter   5 value 0.185264
## iter   6 value 0.185174
## iter   7 value 0.185124
## iter   8 value 0.185091
## iter   9 value 0.185069
## iter  10 value 0.185068
## iter  11 value 0.185068
## iter  12 value 0.185068
## iter  13 value 0.185068
```

```
## iter  13 value 0.185068
## iter  13 value 0.185068
## final  value 0.185068
## converged
```




```
PIB_HN$ttable
```

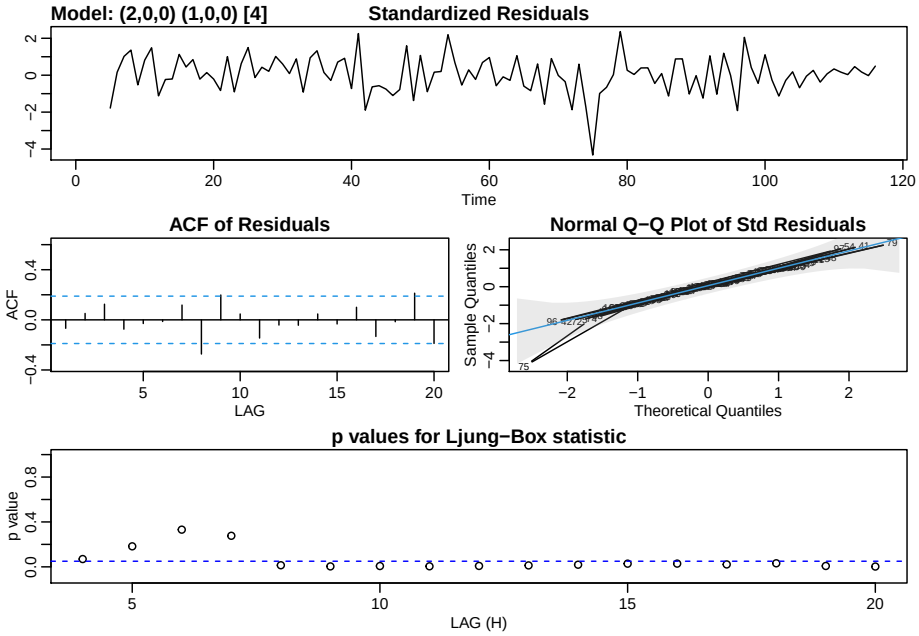
```
##                 Estimate     SE t.value p.value
## ar1               0.5867 0.1333  4.4015  0.0001
## ar2               0.2160 0.1364  1.5839  0.1194
## sar1             -0.3799 0.1277 -2.9750  0.0045
## intercept         2.4738 0.6710  3.6869  0.0006
## INDEX2005.09.01   3.3296 0.9658  3.4473  0.0011
## INDEX2006.12.01   2.0824 1.0148  2.0521  0.0453
## INDEX2008.06.01   2.2639 1.0536  2.1487  0.0364
## PIB_USA           0.7381 0.1920  3.8438  0.0003
## TASA_P            0.0056 0.0028  1.9794  0.0532
```

Modelo de regresión para el PIB de USA

```
Y_USA     <-window(diff(log(TRIM$PIB_USA), lag=4)*100, start="1990-03-01", end="2018-12-01")
DUM_USA   <-window(TRIM[, c("INDEX2008.12.01", "INDEX2009.12.01")], start="1990-03-01", end="2018
PIB_USA   <-sarima(Y_USA, 2,0,0,P=1, D=0, Q=0, 4, xreg=DUM_USA )
```

```
## initial  value 0.434672
## iter   2 value 0.189957
## iter   3 value 0.021610
```

```
## iter   4 value -0.116248
## iter   5 value -0.251754
## iter   6 value -0.332615
## iter   7 value -0.410324
## iter   8 value -0.433150
## iter   9 value -0.436896
## iter  10 value -0.439417
## iter  11 value -0.440979
## iter  12 value -0.441051
## iter  13 value -0.441096
## iter  14 value -0.441109
## iter  15 value -0.441110
## iter  16 value -0.441110
## iter  17 value -0.441111
## iter  18 value -0.441115
## iter  19 value -0.441117
## iter  20 value -0.441118
## iter  21 value -0.441119
## iter  22 value -0.441119
## iter  22 value -0.441119
## iter  22 value -0.441119
## final  value -0.441119
## converged
```

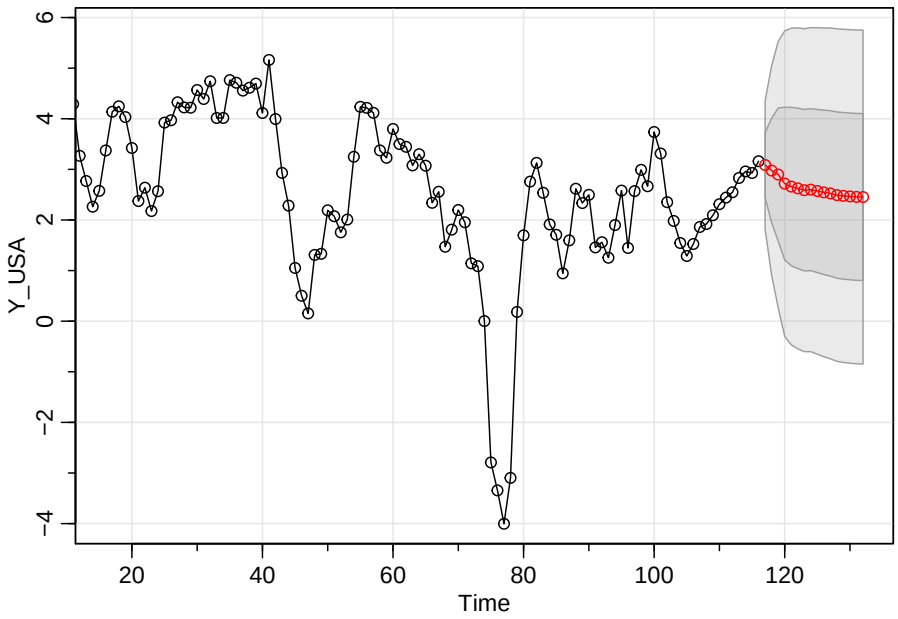


```
PIB_USA$ttable
```

```
##                  Estimate     SE t.value p.value
## ar1                1.2831 0.0888 14.4513  0.0000
## ar2               -0.3694 0.0899 -4.1111  0.0001
## sar1              -0.3721 0.0925 -4.0228  0.0001
## intercept          2.4027 0.4862  4.9417  0.0000
## INDEX2008.12.01    0.3768 0.3851  0.9784  0.3301
## INDEX2009.12.01   -0.1896 0.3823 -0.4958  0.6210
```

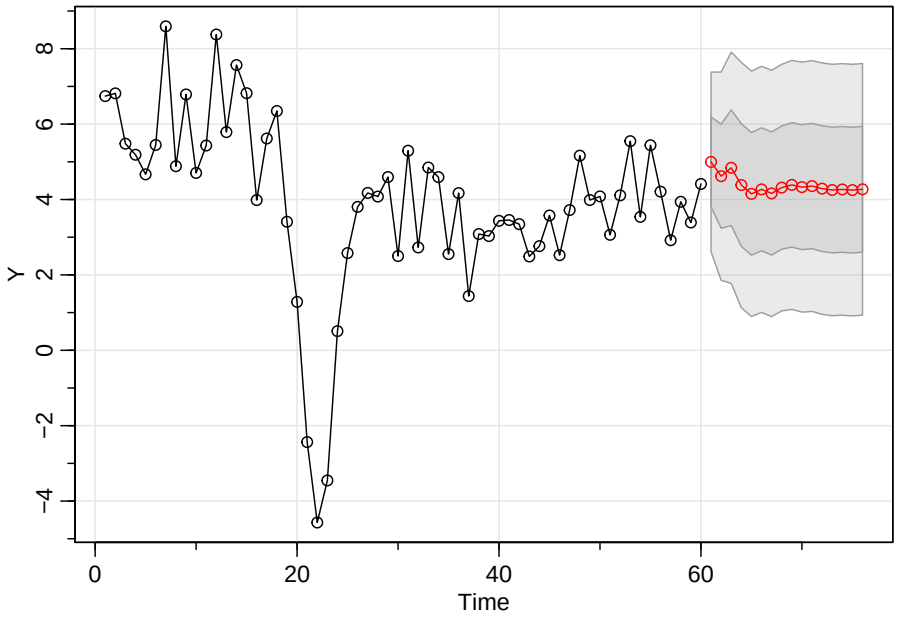
Pronóstico del PIB de USA

```
DUM_USA_N <-window(TRIM[, c("INDEX2008.12.01", "INDEX2009.12.01")], start="2019-03-01", end="2022
Y_USA_N   <-sarima.for(Y_USA,16,2,0,0,1,0,0,4, xreg=DUM_USA, newxreg=DUM_USA_N)
```



Pronóstico del PIB de Honduras

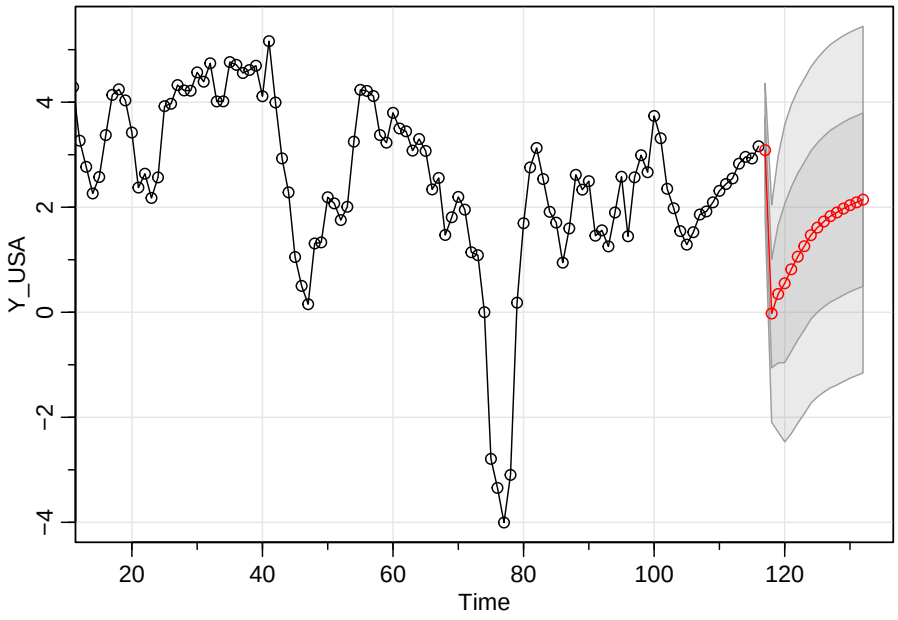
```
dates <- seq(as.Date("2019-03-01"), length = 16, by = "quarter")
DUM_HN_N <-window(TRIM[, c("INDEX2005.09.01", "INDEX2006.12.01", "INDEX2008.06.01")], start="2019
Y_USA_N   <- xts(x=Y_USA_N$pred, order.by = dates)
REG_HN_N<- merge(DUM_HN_N, Y_USA_N, join="left")
data <- rep(1, 16)
i_HN_N = xts(x = data, order.by = dates)
REG_HN_N<- merge(REG_HN_N, i_HN_N, join="left")
Y_N<-sarima.for(Y,16,2,0,0,1,0,0,4, xreg=REG_HN, newxreg=REG_HN_N)
```



2.3 Simulación de shock en el PIB de USA

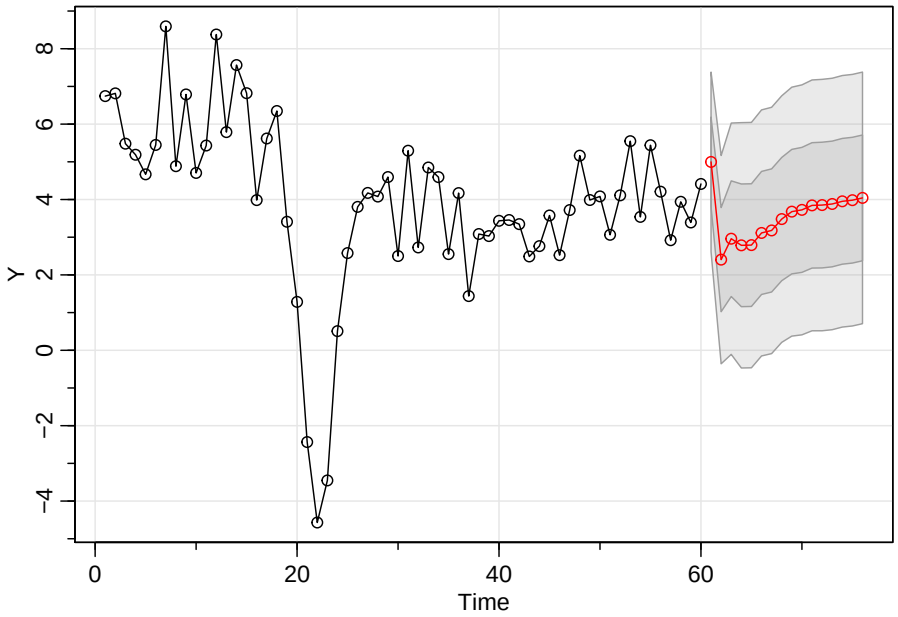
Simulación

```
dates <- seq(as.Date("2019-03-01"), length = 16, by = "quarter")
shock <-c()
shock[1]<- 0
shock[2]<- -3*(1/-0.1896)
for(i in 3:16 ){
  shock[i]<-0.85*shock[i-1]
}
shock_Y_USA= xts(x = shock, order.by = dates)
REG_SHOCK<-window(TRIM[, c("INDEX2008.12.01")], start="2019-03-01", end="2022-12-01")
REG_SHOCK<- merge(REG_SHOCK, shock_Y_USA, join="left")
Y_USA_SHOCK<-sarima.for(Y_USA,16,2,0,0,1,0,0,4, xreg=DUM_USA, newxreg=REG_SHOCK)
```



Transimisión del shock al PIB de Honduras

```
Y_USA_S <- xts(x=Y_USA_SHOCK$pred, order.by = dates)
REG_HN_S<- merge(DUM_HN_N, Y_USA_S, join="left")
REG_HN_S<- merge(REG_HN_S, i_HN_N, join="left")
Y_S<-      sarima.for(Y,16,2,0,0,1,0,0,4, xreg=REG_HN, newxreg=REG_HN_S)
```

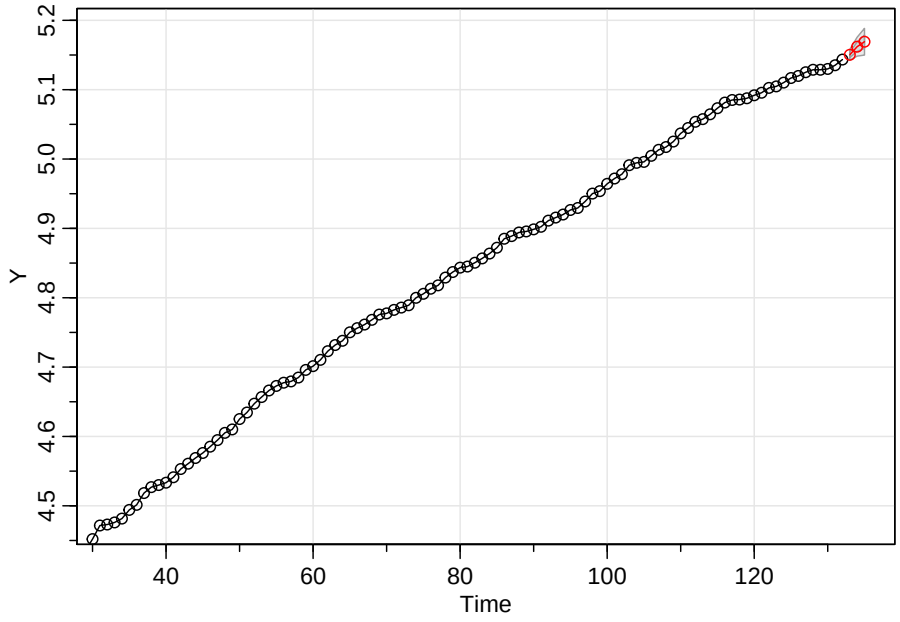

Y
8
6
4
2
0
−2
−4
0
20
40
60
Time

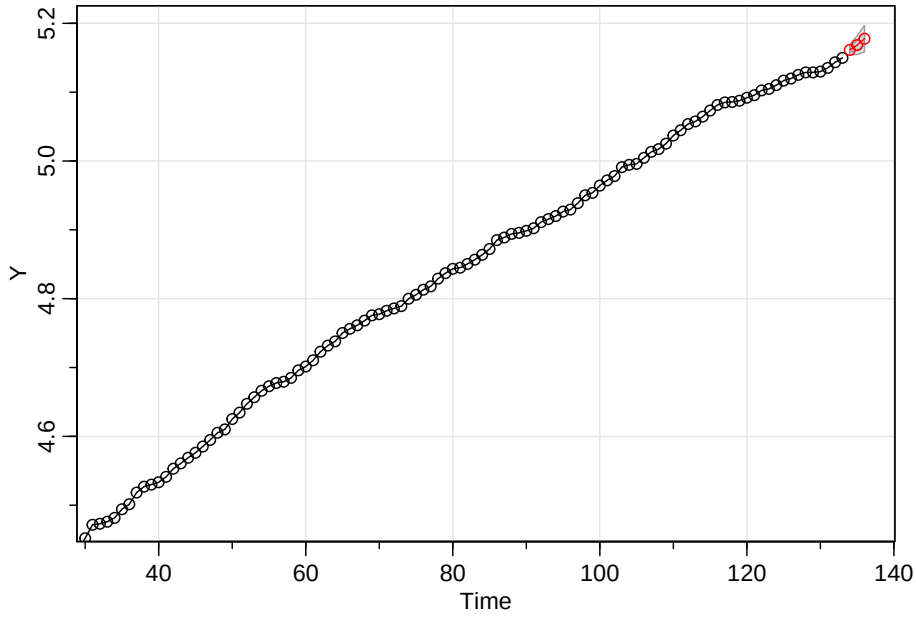
Chapter 3

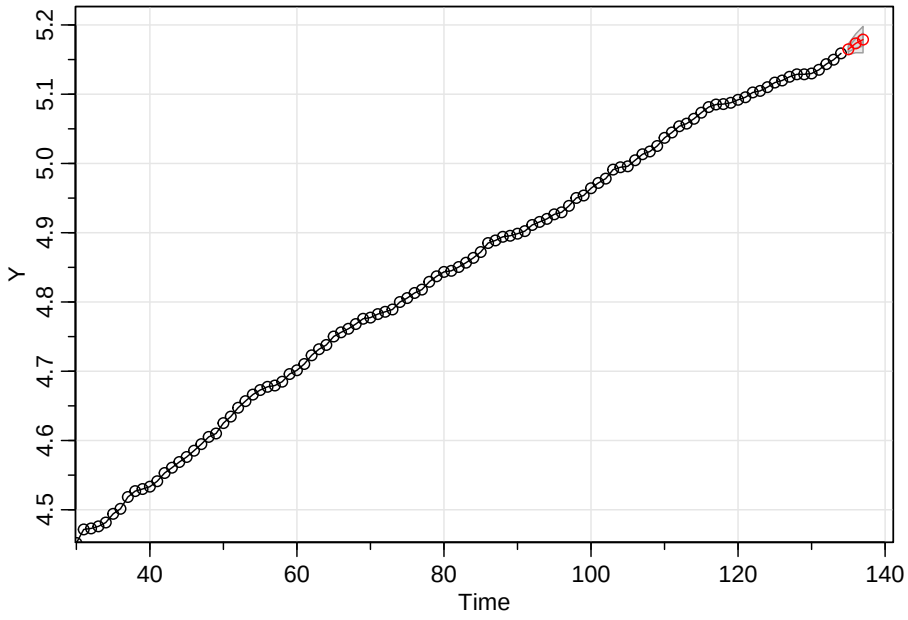
Ejercicio fuera de muestra ipc de Honduras

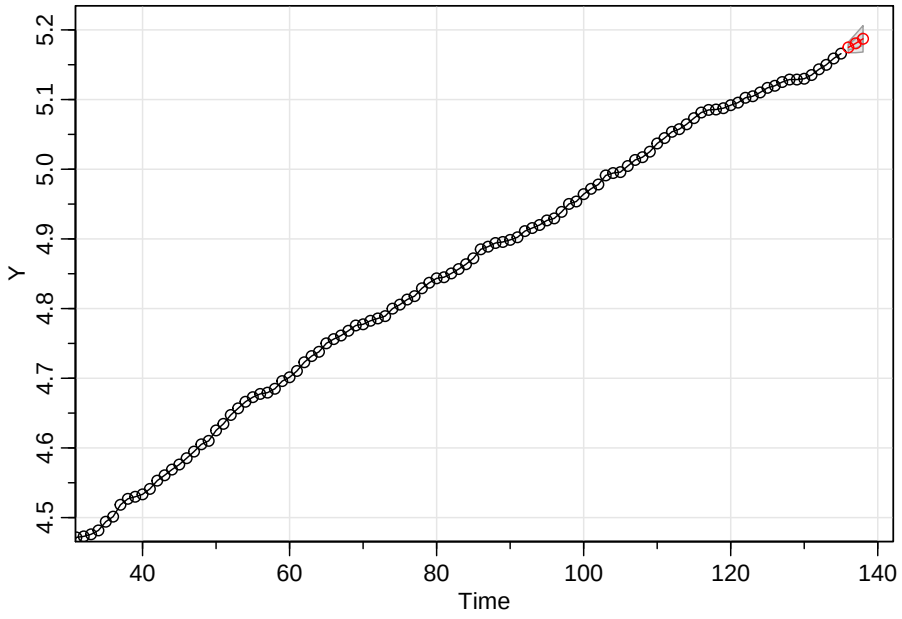
3.1 Ipc de Honduras

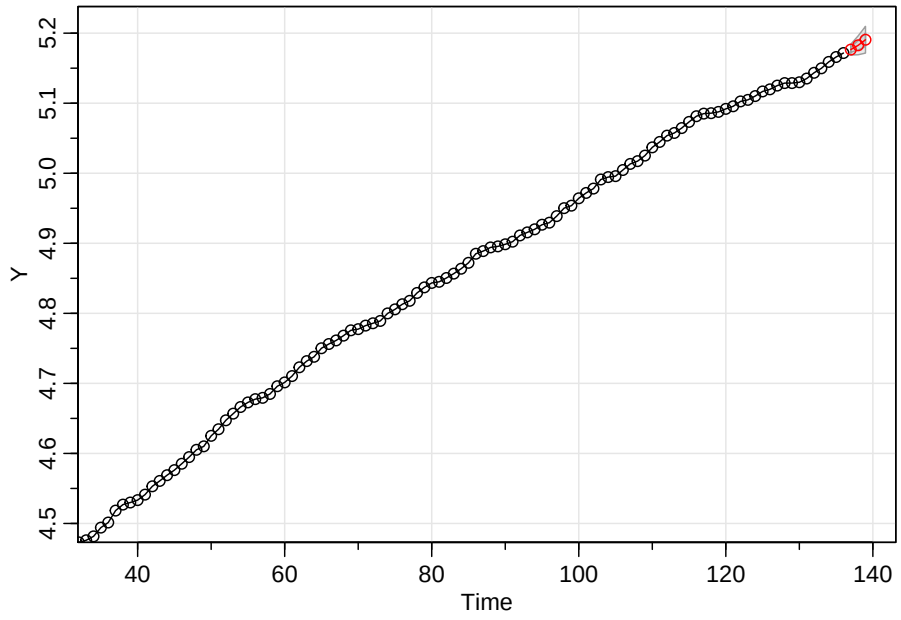
```
IPC<-MES$IPC
FECHA             <-index(MES)
t_inicial         <-first(FECHA,'1 month')
index_final       <- last(index(FECHA))
fecha_contador    <-seq(as.Date(t_inicial), length =index_final, by = "months")
counter           <-c(1:index_final)
contador          <-xts(x=counter, order.by = fecha_contador)
inicio_estimacion<-coredata(contador["1996-01-01"])[1]
final_estimacion <-coredata(contador["2006-12-01"])[1]
H                 <-3
dates_F           <- seq(as.Date(FECHA[inicio_estimacion]), length =final_estimacion-inicio_estima
assign(paste('IPC_', H, sep=''), xts(x=window(log(MES$IPC), start=FECHA[inicio_estimacion], end=F

for(i in 1:10){
Y                 <-window(log(MES$IPC), start=FECHA[inicio_estimacion], end=FECHA[final_estimacio
Y_F               <-sarima.for(Y,H,3,0,11,0,1,0,12, xreg=NULL, newxreg=NULL)
dates_out         <-seq(as.Date(FECHA[final_estimacion+i]), length =H, by = "months")
Y_F_P             <-xts(x=Y_F$pred, order.by = dates_out)
assign(paste('IPC_', H, sep=''),rbind(get(paste('IPC_', H, sep='')), Y_F_P[H]) )
}
```

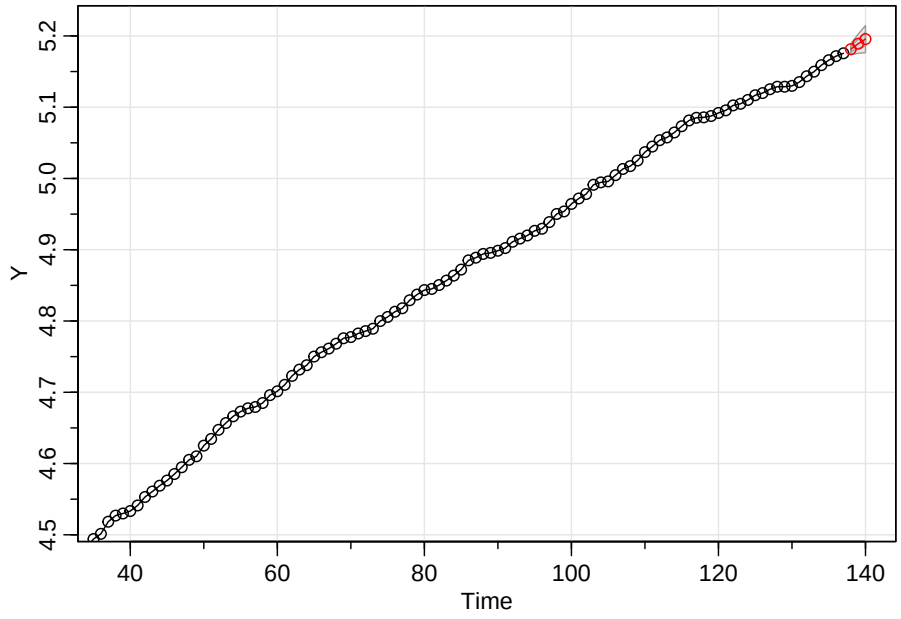

Y
4.5
4.6
4.7
4.8
4.9
5.0
5.1
5.2
40
60
80
100
120
Time

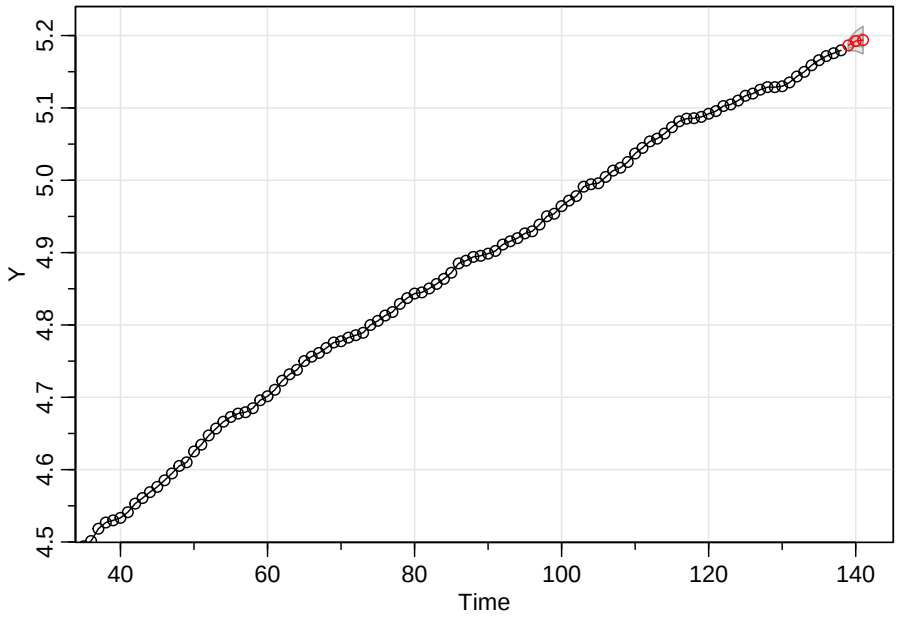

Y
4.6
4.8
5.0
5.2
40
60
80
100
120
140
Time

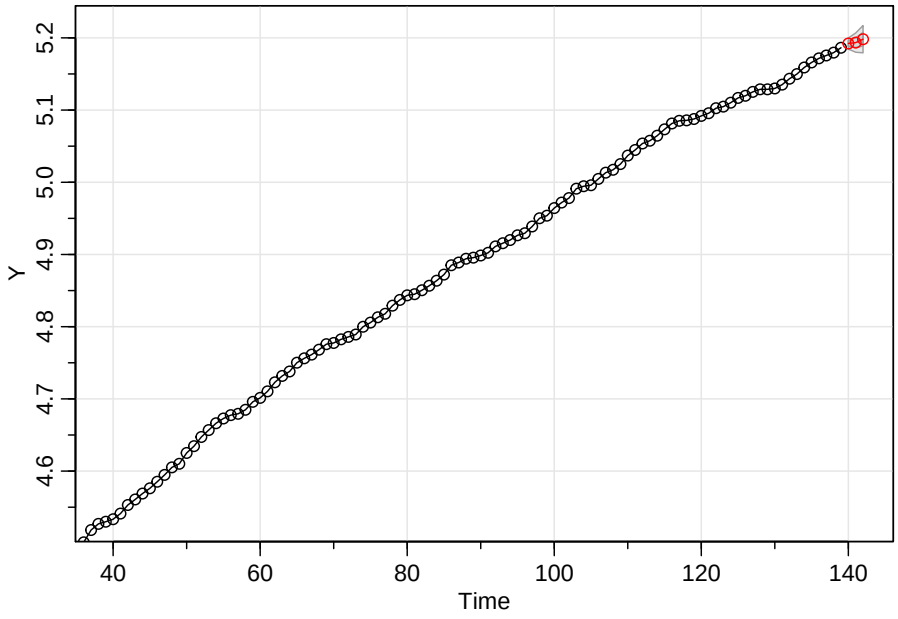

Y
4.5
4.6
4.7
4.8
4.9
5.0
5.1
5.2
40
60
80
100
120
140
Time

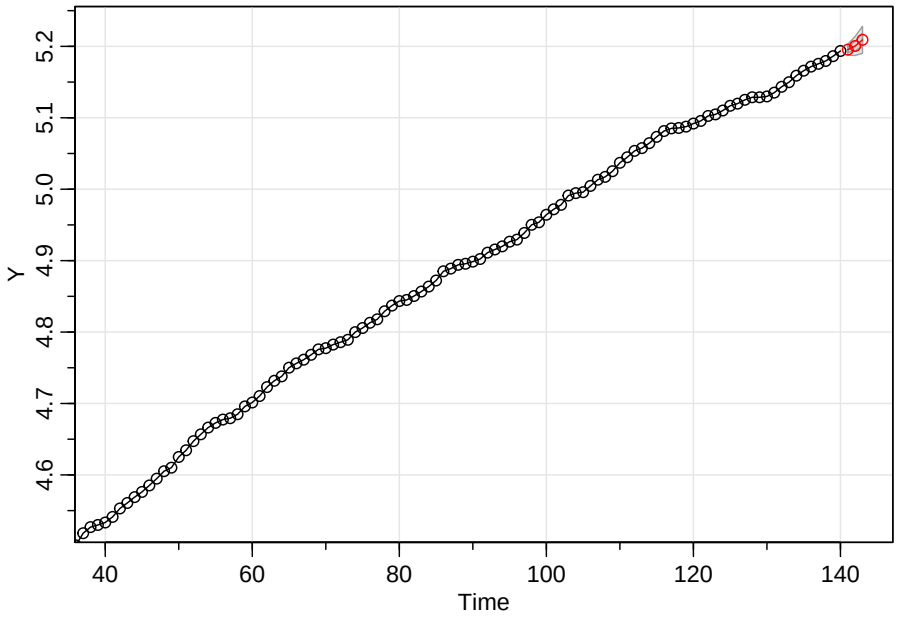

Y
4.5
4.6
4.7
4.8
4.9
5.0
5.1
5.2
40
60
80
100
120
140
Time

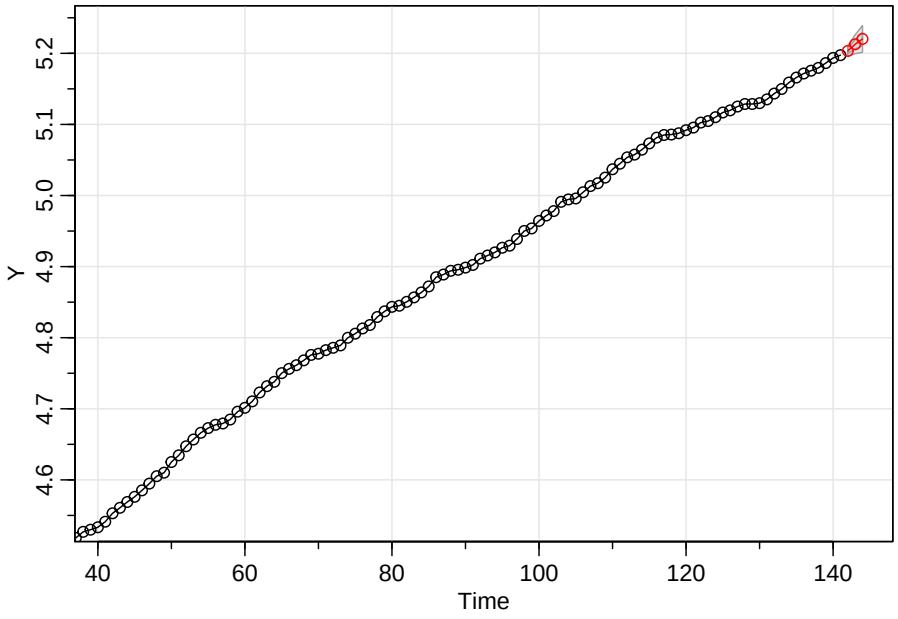

Y
4.5
4.6
4.7
4.8
4.9
5.0
5.1
5.2
40
60
80
100
120
140
Time


Y
4.5
4.6
4.7
4.8
4.9
5.0
5.1
5.2
40
60
80
100
120
140
Time


Y
4.5
4.6
4.7
4.8
4.9
5.0
5.1
5.2
40
60
80
100
120
140
Time


Y
4.6
4.7
4.8
4.9
5.0
5.1
5.2
40
60
80
100
120
140
Time






```
compara          <-merge(log(MES$IPC),get(paste('IPC_', H, sep='')),join='right',fill=0
compara
```

```
##                  IPC    IPC.1
## 1996-01-01 4.015125 4.015125
```

```
## 1996-02-01 4.043140 4.043140
## 1996-03-01 4.058452 4.058452
## 1996-04-01 4.078832 4.078832
## 1996-05-01 4.099510 4.099510
## 1996-06-01 4.124369 4.124369
## 1996-07-01 4.146943 4.146943
## 1996-08-01 4.174600 4.174600
## 1996-09-01 4.191478 4.191478
## 1996-10-01 4.206069 4.206069
## 1996-11-01 4.225648 4.225648
## 1996-12-01 4.236681 4.236681
## 1997-01-01 4.256881 4.256881
## 1997-02-01 4.280811 4.280811
## 1997-03-01 4.288338 4.288338
## 1997-04-01 4.294164 4.294164
## 1997-05-01 4.306671 4.306671
## 1997-06-01 4.319496 4.319496
## 1997-07-01 4.333369 4.333369
## 1997-08-01 4.345675 4.345675
## 1997-09-01 4.348796 4.348796
## 1997-10-01 4.346134 4.346134
## 1997-11-01 4.353825 4.353825
## 1997-12-01 4.356012 4.356012
## 1998-01-01 4.364260 4.364260
## 1998-02-01 4.388778 4.388778
## 1998-03-01 4.405547 4.405547
## 1998-04-01 4.422721 4.422721
## 1998-05-01 4.434569 4.434569
## 1998-06-01 4.452000 4.452000
## 1998-07-01 4.471568 4.471568
## 1998-08-01 4.473026 4.473026
## 1998-09-01 4.476017 4.476017
## 1998-10-01 4.481571 4.481571
## 1998-11-01 4.494095 4.494095
## 1998-12-01 4.501361 4.501361
## 1999-01-01 4.518372 4.518372
## 1999-02-01 4.526923 4.526923
## 1999-03-01 4.529910 4.529910
## 1999-04-01 4.533346 4.533346
## 1999-05-01 4.541166 4.541166
## 1999-06-01 4.553044 4.553044
## 1999-07-01 4.560712 4.560712
## 1999-08-01 4.568910 4.568910
## 1999-09-01 4.576166 4.576166
## 1999-10-01 4.585253 4.585253
## 1999-11-01 4.594759 4.594759
```

```
## 1999-12-01 4.605170 4.605170
## 2000-01-01 4.610123 4.610123
## 2000-02-01 4.625116 4.625116
## 2000-03-01 4.634529 4.634529
## 2000-04-01 4.647271 4.647271
## 2000-05-01 4.656813 4.656813
## 2000-06-01 4.666265 4.666265
## 2000-07-01 4.672829 4.672829
## 2000-08-01 4.677491 4.677491
## 2000-09-01 4.679350 4.679350
## 2000-10-01 4.684905 4.684905
## 2000-11-01 4.695925 4.695925
## 2000-12-01 4.701389 4.701389
## 2001-01-01 4.710431 4.710431
## 2001-02-01 4.722953 4.722953
## 2001-03-01 4.731803 4.731803
## 2001-04-01 4.737951 4.737951
## 2001-05-01 4.750136 4.750136
## 2001-06-01 4.756173 4.756173
## 2001-07-01 4.761319 4.761319
## 2001-08-01 4.768139 4.768139
## 2001-09-01 4.775756 4.775756
## 2001-10-01 4.777441 4.777441
## 2001-11-01 4.782479 4.782479
## 2001-12-01 4.785824 4.785824
## 2002-01-01 4.789157 4.789157
## 2002-02-01 4.799914 4.799914
## 2002-03-01 4.805659 4.805659
## 2002-04-01 4.812997 4.812997
## 2002-05-01 4.817859 4.817859
## 2002-06-01 4.829113 4.829113
## 2002-07-01 4.837075 4.837075
## 2002-08-01 4.843399 4.843399
## 2002-09-01 4.844974 4.844974
## 2002-10-01 4.850467 4.850467
## 2002-11-01 4.856707 4.856707
## 2002-12-01 4.863681 4.863681
## 2003-01-01 4.872139 4.872139
## 2003-02-01 4.885072 4.885072
## 2003-03-01 4.888844 4.888844
## 2003-04-01 4.894101 4.894101
## 2003-05-01 4.895598 4.895598
## 2003-06-01 4.898586 4.898586
## 2003-07-01 4.902307 4.902307
## 2003-08-01 4.911183 4.911183
## 2003-09-01 4.915592 4.915592
```

```
## 2003-10-01 4.919981 4.919981
## 2003-11-01 4.926529 4.926529
## 2003-12-01 4.929425 4.929425
## 2004-01-01 4.938781 4.938781
## 2004-02-01 4.950177 4.950177
## 2004-03-01 4.953712 4.953712
## 2004-04-01 4.964242 4.964242
## 2004-05-01 4.971894 4.971894
## 2004-06-01 4.978112 4.978112
## 2004-07-01 4.991113 4.991113
## 2004-08-01 4.994506 4.994506
## 2004-09-01 4.995860 4.995860
## 2004-10-01 5.004617 5.004617
## 2004-11-01 5.013298 5.013298
## 2004-12-01 5.017280 5.017280
## 2005-01-01 5.025195 5.025195
## 2005-02-01 5.036953 5.036953
## 2005-03-01 5.044715 5.044715
## 2005-04-01 5.053695 5.053695
## 2005-05-01 5.057519 5.057519
## 2005-06-01 5.064492 5.064492
## 2005-07-01 5.073297 5.073297
## 2005-08-01 5.081404 5.081404
## 2005-09-01 5.085124 5.085124
## 2005-10-01 5.085743 5.085743
## 2005-11-01 5.087596 5.087596
## 2005-12-01 5.091908 5.091908
## 2006-01-01 5.095589 5.095589
## 2006-02-01 5.102565 5.102565
## 2006-03-01 5.104733 5.104733
## 2006-04-01 5.110179 5.110179
## 2006-05-01 5.116795 5.116795
## 2006-06-01 5.119789 5.119789
## 2006-07-01 5.125154 5.125154
## 2006-08-01 5.128715 5.128715
## 2006-09-01 5.128715 5.128715
## 2006-10-01 5.129899 5.129899
## 2006-11-01 5.135210 5.135210
## 2006-12-01 5.143416 5.143416
## 2007-01-01 5.149817 5.149817
## 2007-02-01 5.158945 5.158945
## 2007-03-01 5.165928 5.169134
## 2007-04-01 5.171620 5.177588
## 2007-05-01 5.175585 5.178838
## 2007-06-01 5.179534 5.187207
## 2007-07-01 5.186268 5.190610
```

```
## 2007-08-01 5.193512 5.195522
## 2007-09-01 5.197391 5.193731
## 2007-10-01 5.209486 5.198121
## 2007-11-01 5.226821 5.209163
## 2007-12-01 5.228431 5.220281
```