ESTADÍSTICA III

TALLER 10: MODELOS ARIMA

Considere la serie en el archivo ARIMA110.SIMUL.txt, un conjunto de N=201 observaciones obtenidas por simulación. La gráfica de la serie, de su primera diferencia y de sus respectivas ACFs son presentadas en la Figura 1:

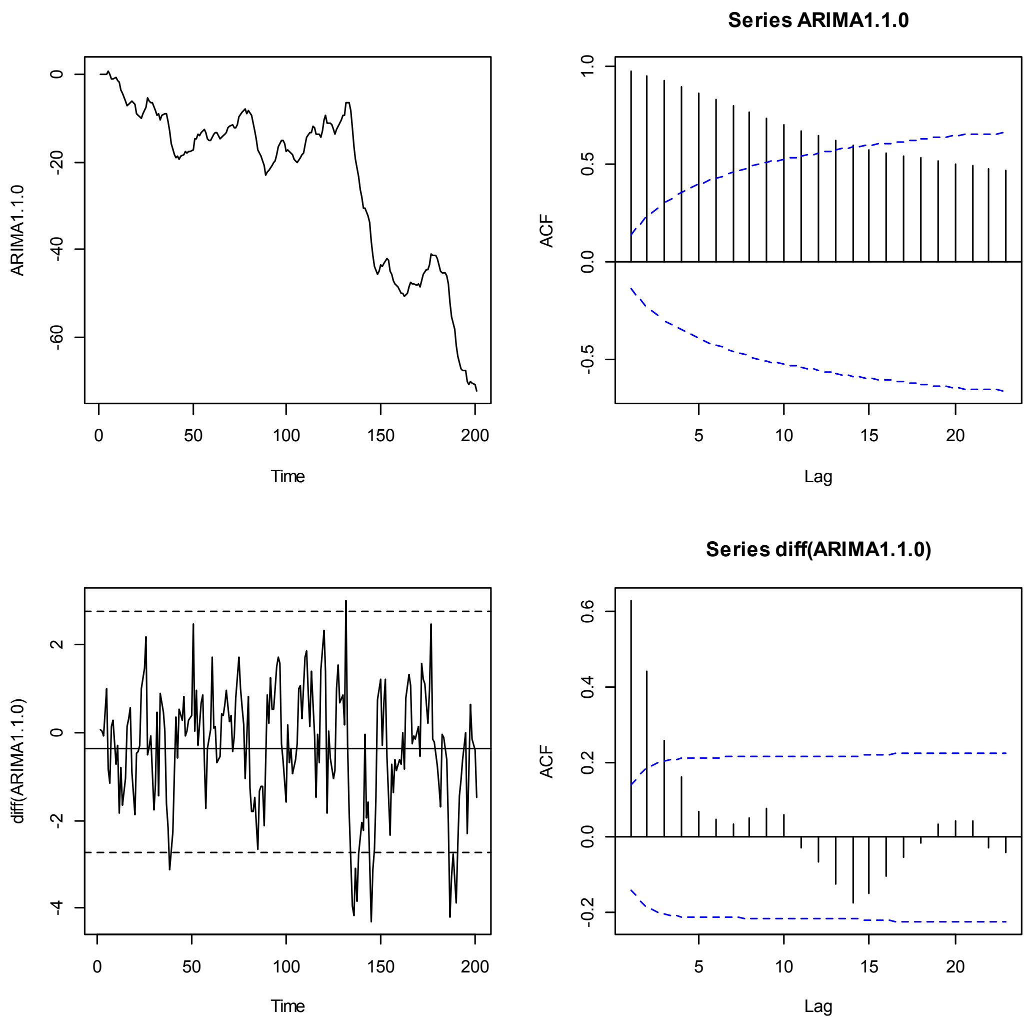


Figura 1: Serie original, su primera diferencia y sus respectivas ACFs

a) ¿La serie original proviene de un proceso estacionario? ¿por qué?
b) ¿La primera diferencia de la serie es estacionaria? ¿por qué? ¿será necesaria una segunda diferencia?
c) Considerando sólo los primeros n=191 observaciones, realice la identificación de modelos ARIMA usando las siguientes herramientas
   i. ACF y PACF de la serie de los n=191 primeros datos, diferenciada apropiadamente
   ii. EACF de la serie de los n=191 primeros datos, diferenciada apropiadamente con máximo p=7, q=13.
   iii. La función `auto.arima()` aplicada a la serie de los n=191 primeros datos, diferenciada apropiadamente
   iv. La función `auto.arima()` aplicada a la serie de los n=191 primeros datos
   v. La función `autoarmafit()` aplicada a la serie de los n=191 primeros datos, diferenciada apropiadamente
   vi. La función `armasubsets()` aplicada a la serie de los n=191 primeros datos, diferenciada apropiadamente

d) Para cada modelo identificado, escriba su ecuación teórica, ajuste el modelo con los n=191 primeros datos y valide supuestos
e) Pronostique los últimos 10 datos de la serie
f) Compare los modelos con base en ajuste, pronóstico y validez de supuestos (gráficos residuos, ACF, PACF, Ljung-Box y normalidad con Shapiro Wilk). ¿Cuál modelo se recomienda?

<u>Resultados</u>

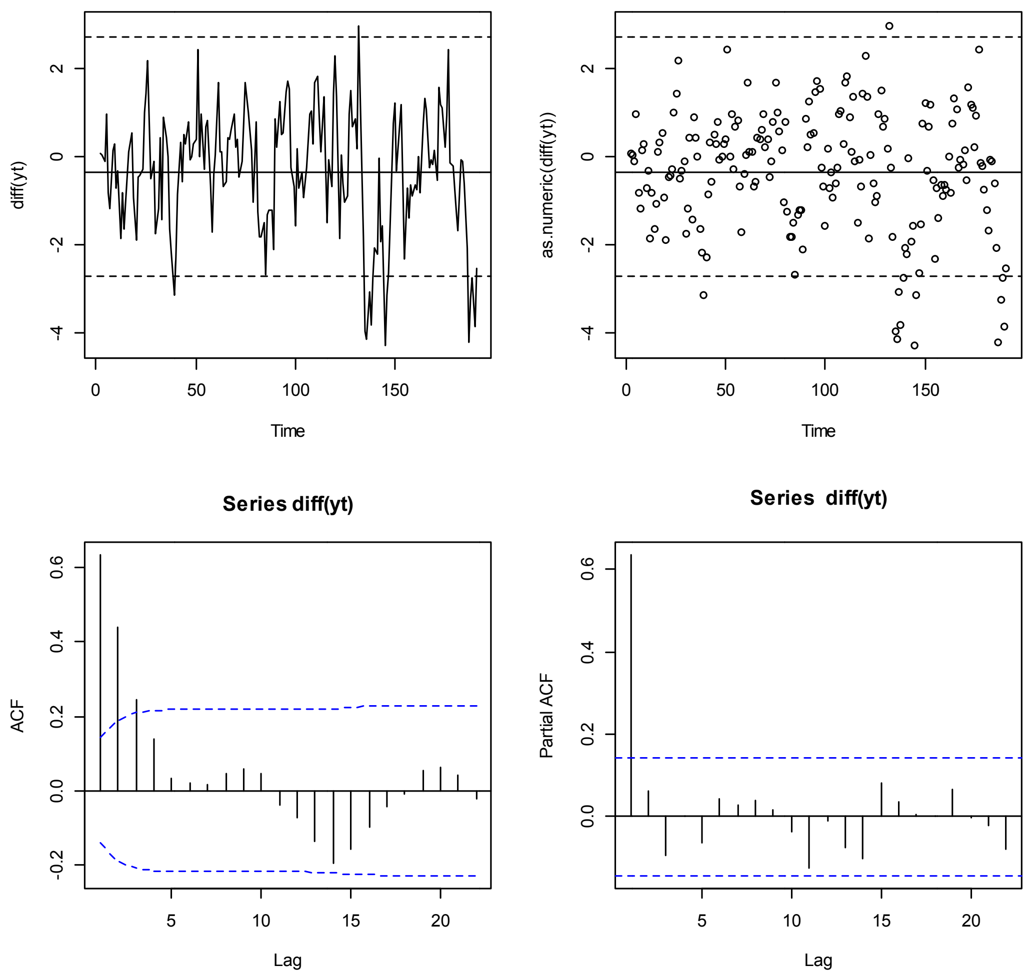
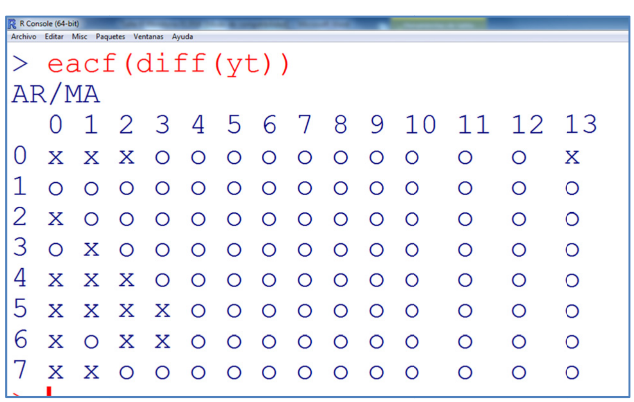
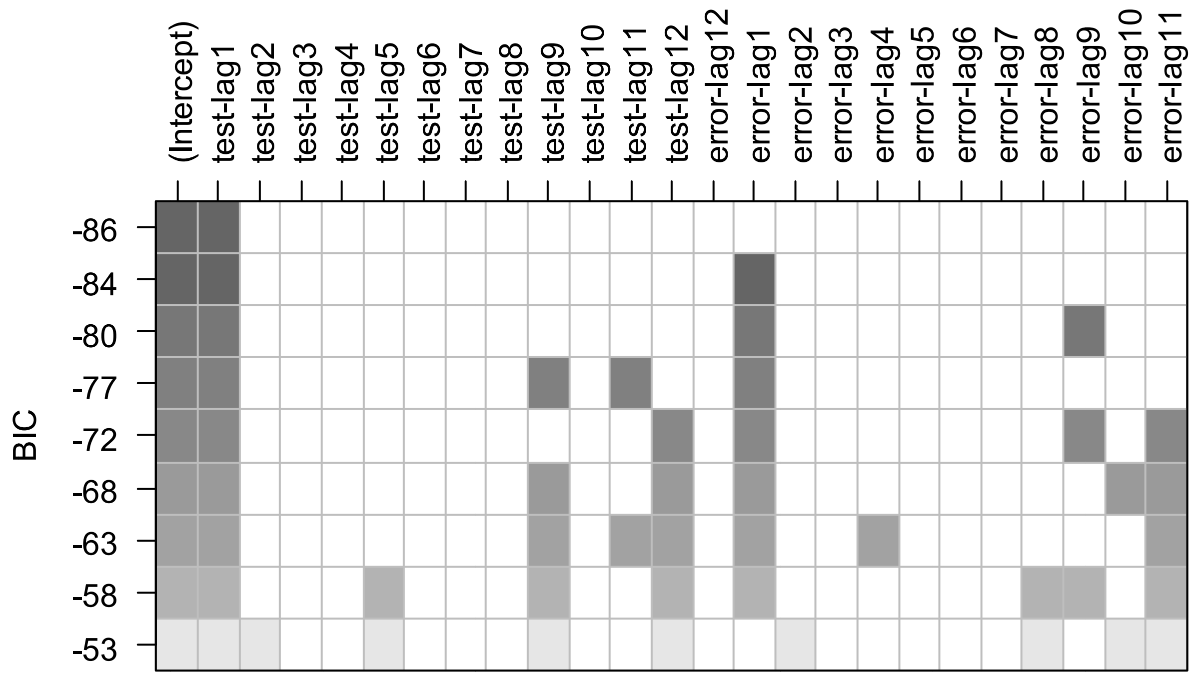


Figura 2: Primera diferencia con los n=191 primeros datos, su ACF y PACF



```
> eacf(diff(yt))
AR/MA
  0 1 2 3 4 5 6 7 8 9 10 11 12 13
0 x x x o o o o o o o o  o  o  x
1 o o o o o o o o o o o  o  o  o
2 x o o o o o o o o o o  o  o  o
3 o x o o o o o o o o o  o  o  o
4 x x x o o o o o o o o  o  o  o
5 x x x x o o o o o o o  o  o  o
6 x o x x o o o o o o o  o  o  o
7 x x o o o o o o o o o  o  o  o
```

Figura 3: EACF de la primera diferencia con los n=191 primeros datos



Observe que la primera diferencia de los primero 191 datos tiene una media muestral que es distinta de cero!!!

```
> mean(diff(yt))
[1] -0.3400249
```

```
> autoarmafit(diff(yt))


Case No. 1

 AR coefficient Standard deviation
 0.561092       1.149094
 0.052312       0.820622

 MA coefficient Standard deviation
 -0.089643      1.149149
 -0.081511      0.126970

 AIC    23.251782
 Innovation variance    1.083582
 Final gradient 5.523876e-02    -1.315073e-02    -6.681693e-02    -1.421383e-02
```

```
> auto.arima(diff(yt)) #auto.arima sobre serie diferencia
Series: diff(yt)
ARIMA(1,0,0) with non-zero mean

Coefficients:
         ar1     mean
      0.6405  -0.3566
s.e.  0.0558   0.2123

sigma^2 estimated as 1.139:  log likelihood=-281.25
AIC=568.51   AICc=568.64   BIC=578.25
> auto.arima(yt) #auto.arima sobre serie sin diferenciar
Series: yt
ARIMA(1,1,0) with drift

Coefficients:
         ar1    drift
      0.6405  -0.3566
s.e.  0.0558   0.2123

sigma^2 estimated as 1.139:  log likelihood=-281.25
AIC=568.51   AICc=568.64   BIC=578.25
>
```

RESUMEN AJUSTES (Valores P bajo aproximación N(0,1))

AJUSTE MODELO 1: ARIMA(1,1,0) sin deriva

Parámetros	Estimación	Error estándar	Z_0	$P(\|Z\|>\|Z_0\|)$
ϕ_1	0.6616	0.0546	12.12001	$< 2.2\times10^{-16}$

Shapiro-Wilk normality test
W = 0.9953, p-value = 0.8154

AJUSTE MODELO 2: ARIMA(1,1,0) con deriva

Parámetros	Estimación	Error estándar	Z_0	$P(\|Z\|>\|Z_0\|)$
ϕ_1	0.6405	0.0559	11.45301	$< 2.2\times10^{-16}$
δ	-0.3566	0.2129	-1.67490	0.0940

Shapiro-Wilk normality test
W = 0.9953, p-value = 0.8148

AJUSTE MODELO 3: ARIMA(2,1,1) sin deriva

Parámetros	Estimación	Error estándar	Z_0	$P(\|Z\|>\|Z_0\|)$
ϕ_1	-0.0934	0.2440	-0.38270	0.7019
ϕ_2	0.5539	0.1497	3.69920	0.0002
θ_1	0.7063	0.2629	2.68656	0.0072

Shapiro-Wilk normality test
W = 0.994, p-value = 0.6343

AJUSTE MODELO 3b: ARIMA(2,1,1) con deriva

Parámetros	Estimación	Error estándar	Z_0	$P(\|Z\|>\|Z_0\|)$
ϕ_1	-0.12301	0.23245	-0.5292	0.596655
ϕ_2	0.54270	0.13871	3.9124	9.139×10^{-05}
θ_1	0.71922	0.24844	2.8950	3.792×10^{-03}
δ	-0.35972	0.22457	-1.6018	0.109201

Shapiro-Wilk normality test
W = 0.9944, p-value = 0.6907

AJUSTE MODELO 4: ARIMA(1,1,1) sin deriva

Parámetros	Estimación	Error estándar	Z_0	$P(\|Z\|>\|Z_0\|)$
ϕ_1	0.7240	0.0724	9.99915	1.537×10^{-23}
θ_1	-0.1083	0.0979	-1.10543	0.2690

Shapiro-Wilk normality test
W = 0.9936, p-value = 0.574

AJUSTE MODELO 4b: ARIMA(1,1,1) con deriva

Parámetros	Estimación	Error estándar	Z_0	$P(\|Z\|>\|Z_0\|)$
ϕ_1	0.697849	0.076279	9.1487	$< 2.2\times10^{-16}$
θ_1	-0.095204	0.099282	-0.9589	0.3376
δ	-0.361760	0.227633	-1.5892	0.1120

Shapiro-Wilk normality test
W = 0.99382, p-value = 0.61

AJUSTE MODELO 5: ARIMA(2,1,2) sin deriva

Parámetros	Estimación	Error estándar	Z_0	$P(\|Z\|>\|Z_0\|)$
ϕ_1	-0.080769	0.309080	-0.2613	0.79384
ϕ_2	0.535567	0.249007	2.1508	0.03149
θ_1	0.696578	0.318621	2.1862	0.02880
θ_2	0.013774	0.128256	0.1074	0.91447

Shapiro-Wilk normality test
W = 0.99406, p-value = 0.6442

AJUSTE MODELO 5b: ARIMA(2,1,2) con deriva

Parámetros	Estimación	Error estándar	Z_0	$P(\|Z\|>\|Z_0\|)$
ϕ_1	-0.094275	0.356065	-0.2648	0.79119
ϕ_2	0.500985	0.279999	1.7892	0.07358
θ_1	0.696618	0.364929	1.9089	0.05627
θ_2	0.031185	0.135208	0.2306	0.81759
δ	-0.358301	0.220323	-1.6263	0.10390

Shapiro-Wilk normality test
W = 0.99459, p-value = 0.7205

Criterios de información con $\exp\left(C_n^*(p)\right)$

	p	AIC	BIC
modelo1	1	1.148930	1.168661
modelo2	2	1.145220	1.184892
modelo3	3	1.152733	1.213148
modelo3b	4	1.150442	1.231529
modelo4	2	1.153581	1.193544
modelo4b	3	1.151691	1.212051
modelo5	4	1.164796	1.246895
modelo5b	5	1.162196	1.265478

Ecuaciones teóricas y ajustadas

Modelo 1: $(1-\phi_1 B)(1-B)Y_t = E_t$, $E_t\ un\ RB \sim N(0,\sigma^2)$. Como $(1-\phi_1 B)(1-B) = 1-(1+\phi_1)B+\phi_1 B^2$, la ecuación final es:
$Y_t = (1+\phi_1)Y_{t-1} - \phi_1 Y_{t-2} + E_t$, $E_t\ un\ RB \sim N(0,\sigma^2)$.
Ecuación ajustada: $\hat{Y}_t = 1.6616Y_{t-1} - 0.6616Y_{t-2}$,

Modelo2: $(1-\phi_1 B)(1-B)Y_t = \delta + E_t$, $E_t\ un\ RB \sim N(0,\sigma^2)$. La ecuación final es
$Y_t = \delta + (1+\phi_1)Y_{t-1} - \phi_1 Y_{t-2} + E_t$, $E_t\ un\ RB \sim N(0,\sigma^2)$.
Ecuación ajustada: $\hat{Y}_t = -0.3566 + 1.6405Y_{t-1} - 0.6405Y_{t-2}$,

Modelo 3: $(1-\phi_1 B-\phi_2 B^2)(1-B)Y_t = (1+\theta_1 B)E_t$, $E_t\ un\ RB \sim N(0,\sigma^2)$. Como
$(1-\phi_1 B-\phi_2 B^2)(1-B) = 1-(1+\phi_1)B-(\phi_2-\phi_1)B^2+\phi_2 B^3$, la ecuación final es:
$Y_t = (1+\phi_1)Y_{t-1} + (\phi_2-\phi_1)Y_{t-2} - \phi_2 Y_{t-3} + E_t + \theta_1 E_{t-1}$, $E_t\ un\ RB \sim N(0,\sigma^2)$.
Ecuación ajustada: $\hat{Y}_t = 0.9066Y_{t-1} + 0.6473Y_{t-2} - 0.5539Y_{t-3} + 0.7063\hat{E}_{t-1}$

Modelo3b: $(1-\phi_1 B-\phi_2 B^2)(1-B)Y_t = \delta + (1+\theta_1 B)E_t$, $E_t\ un\ RB \sim N(0,\sigma^2)$. La ecuación final es:
$Y_t = \delta + (1+\phi_1)Y_{t-1} + (\phi_2-\phi_1)Y_{t-2} - \phi_2 Y_{t-3} + E_t + \theta_1 E_{t-1}$, $E_t\ un\ RB \sim N(0,\sigma^2)$.
Ecuación ajustada: $\hat{Y}_t = -0.35972 + 0.87699Y_{t-1} + 0.66571Y_{t-2} - 0.54270Y_{t-3} + 0.71922\hat{E}_{t-1}$.

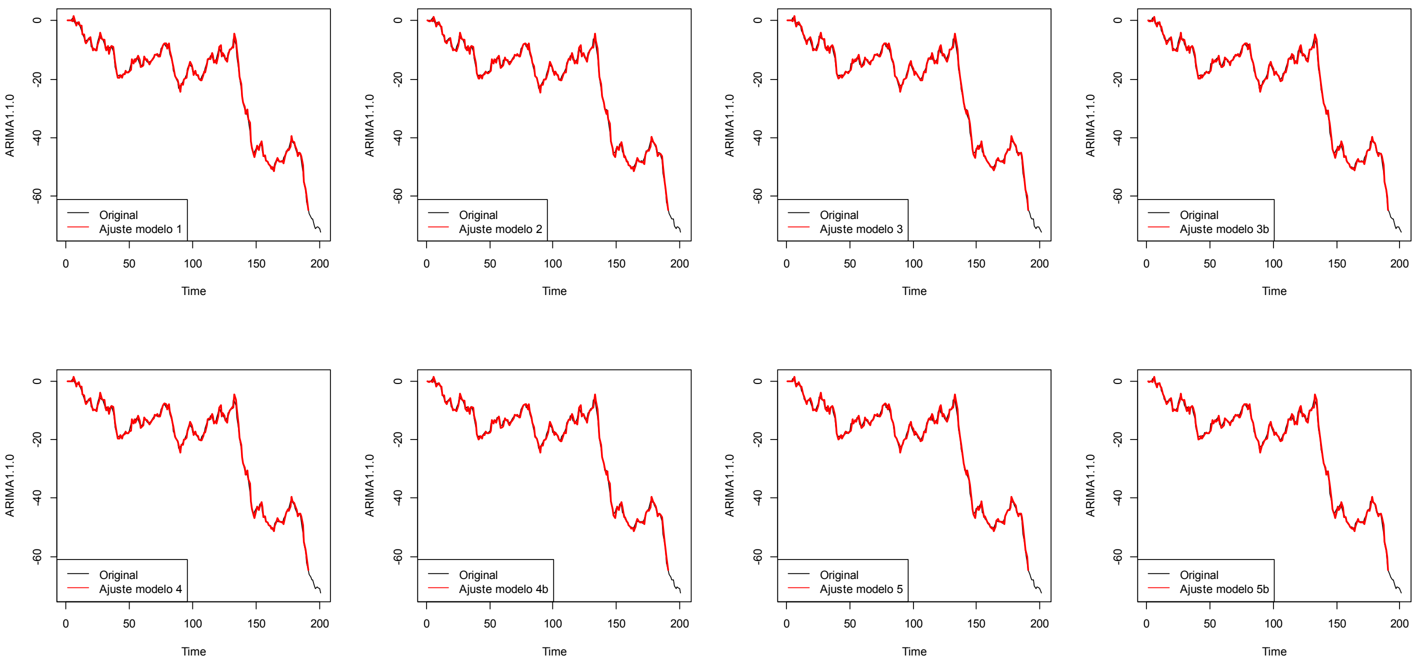
Modelo4: $(1-\phi_1 B)(1-B)Y_t = (1+\theta_1 B)E_t$, $E_t\ un\ RB \sim N(0,\sigma^2)$. Como $(1-\phi_1 B)(1-B) = 1-(1+\phi_1)B+\phi_1 B^2$, la ecuación final es:
$Y_t = (1+\phi_1)Y_{t-1} - \phi_1 Y_{t-2} + E_t + \theta_1 E_{t-1}$, $E_t\ un\ RB \sim N(0,\sigma^2)$.
Ecuación ajustada: $\hat{Y}_t = 1.7240Y_{t-1} - 0.7240Y_{t-2} - 0.1083\hat{E}_{t-1}$.

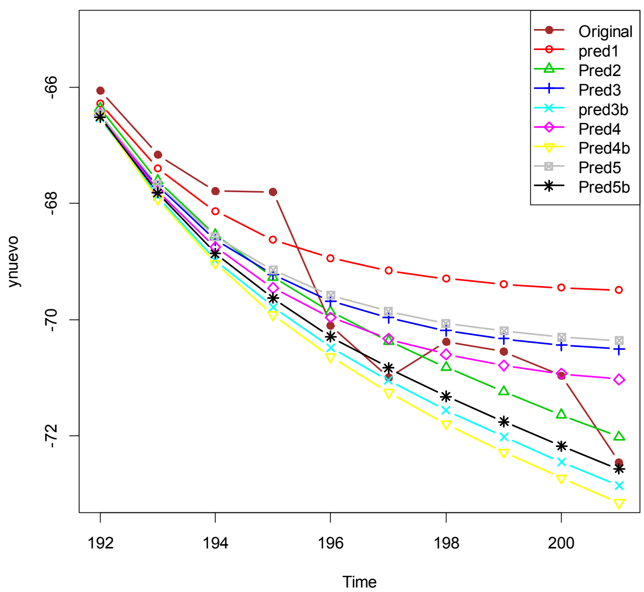
Modelo 4b: $(1-\phi_1 B)(1-B)Y_t = \delta + (1+\theta_1 B)E_t$, $E_t\ un\ RB \sim N(0,\sigma^2)$. La ecuación final es
$Y_t = \delta + (1+\phi_1)Y_{t-1} - \phi_1 Y_{t-2} + E_t + \theta_1 E_{t-1}$, $E_t\ un\ RB \sim N(0,\sigma^2)$.
Ecuación ajustada: $\hat{Y}_t = -0.361760 + 1.697849Y_{t-1} - 0.697849Y_{t-2} - 0.095204\hat{E}_{t-1}$.

Modelo 5: $(1-\phi_1 B-\phi_2 B^2)(1-B)Y_t = (1+\theta_1 B+\theta_2 B^2)E_t$, $E_t\ un\ RB \sim N(0,\sigma^2)$. Como
$(1-\phi_1 B-\phi_2 B^2)(1-B) = 1-(1+\phi_1)B-(\phi_2-\phi_1)B^2+\phi_2 B^3$, la ecuación final es:
$Y_t = (1+\phi_1)Y_{t-1} + (\phi_2-\phi_1)Y_{t-2} - \phi_2 Y_{t-3} + E_t + \theta_1 E_{t-1} + \theta_2 E_{t-2}$, $E_t\ un\ RB \sim N(0,\sigma^2)$.
Ecuación ajustada: $\hat{Y}_t = 0.919231Y_{t-1} + 0.616336Y_{t-2} - 0.535567Y_{t-3} + 0.696578\hat{E}_{t-1} + 0.013774\hat{E}_{t-2}$

Modelo5b: $(1-\phi_1 B-\phi_2 B^2)(1-B)Y_t = \delta + (1+\theta_1 B+\theta_2 B^2)E_t$, $E_t\ un\ RB \sim N(0,\sigma^2)$. La ecuación final es:
$Y_t = \delta + (1+\phi_1)Y_{t-1} + (\phi_2-\phi_1)Y_{t-2} - \phi_2 Y_{t-3} + E_t + \theta_1 E_{t-1} + \theta_2 E_{t-2}$, $E_t\ un\ RB \sim N(0,\sigma^2)$.
Ecuación ajustada: $\hat{Y}_t = -0.358301 + 0.905725Y_{t-1} + 0.59526Y_{t-2} - 0.500985Y_{t-3} + 0.696618\hat{E}_{t-1} + 0.031185\hat{E}_{t-2}$

Nota: En las ecuaciones ajustadas de los modelos que tienen parte MA, $\hat{E}_{t-j}$ es el residuo del ajuste para el tiempo $t-j$ y los residuos de ajuste para cualquier tiempo t corresponden a $\hat{E}_t = Y_t - \hat{Y}_t$ y con base en estos residuos se construyen las gráficas y tests para validar supuestos sobre el error de ajuste E_t: Proceso Ruido Blanco distribuido $N(0,\sigma^2)$





	AmplitudI.P	Cobertura	RMSE	MAE	MAPE
modelo1	19.71	100	1.38	1.13	1.60
modelo2	18.99	100	0.69	0.61	0.88
modelo3	19.98	100	0.93	0.75	1.08
modelo3b	19.27	100	1.10	0.93	1.35
modelo4	19.94	100	0.83	0.64	0.92
modelo4b	19.22	100	1.26	1.10	1.58
modelo5	20.02	100	0.97	0.81	1.16
modelo5b	19.29	100	0.95	0.79	1.14

Ecuaciones de pronóstico con origen en t=191: En las siguientes ecuaciones, $\hat{Y}_{191}(L)$ es el pronóstico L períodos después de t=191;

$$\hat{Y}_{191}(L-j)=\begin{cases}\text{Observación } Y_{191+L-j}, \text{ si } L-j\le 0\\ \text{Pronóstico } L-j \text{ períodos después de } t=191, \text{ si } L-j>0\end{cases};\qquad \hat{E}_{191}(L-i)=\begin{cases}\text{Residuo } \hat{E}_{191+L-i}, \text{ si } L-i\le 0\\ 0, \text{ si } L-i>0\end{cases}$$

Modelo 1:
$\hat{Y}_{191}(L)=1.6616\hat{Y}_{191}(L-1)-0.6616\hat{Y}_{191}(L-2)$.

Modelo 2:
$\hat{Y}_{191}(L)=-0.3566+1.6405\hat{Y}_{191}(L-1)-0.6405\hat{Y}_{191}(L-2)$.

Modelo 3:
$\hat{Y}_{191}(L)=0.9066\hat{Y}_{191}(L-1)+0.6473\hat{Y}_{191}(L-2)-0.5539\hat{Y}_{191}(L-3)+0.7063\hat{E}_{191}(L-1)$.

Modelo 3b:
$\hat{Y}_{191}(L)=-0.35972+0.87699\hat{Y}_{191}(L-1)+0.66571\hat{Y}_{191}(L-2)+-0.54270\hat{Y}_{191}(L-3)+0.71922\hat{E}_{191}(L-1)$.

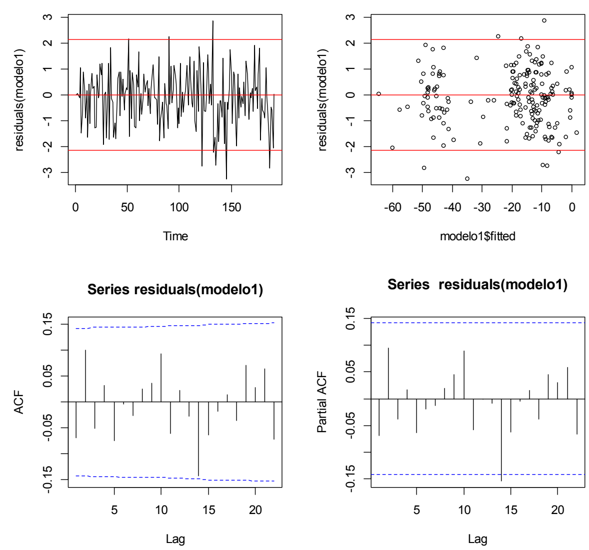
Modelo 4:
$\hat{Y}_{191}(L)=1.7240\hat{Y}_{191}(L-1)-0.7240\hat{Y}_{191}(L-2)-0.1083\hat{E}_{191}(L-1)$.

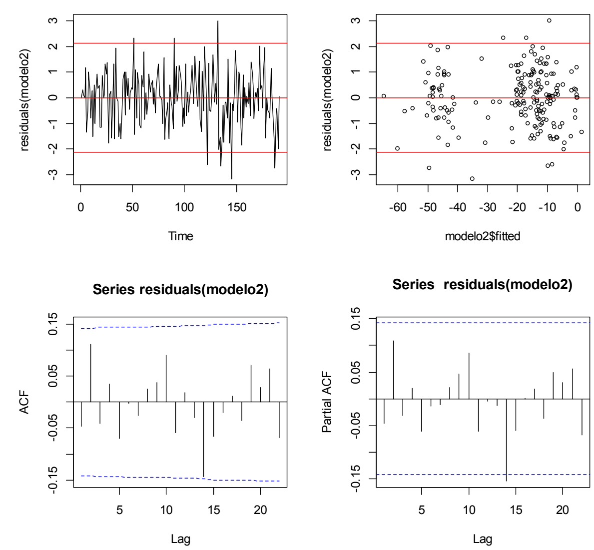
Modelo 4b:
$\hat{Y}_{191}(L)=-0.361760+1.697849\hat{Y}_{191}(L-1)-0.697849\hat{Y}_{191}(L-2)+-0.095204\hat{E}_{191}(L-1)$.

Modelo 5:
$\hat{Y}_{191}(L)=0.919231\hat{Y}_{191}(L-1)+0.616336\hat{Y}_{191}(L-2)-0.535567\hat{Y}_{191}(L-3)+0.696578\hat{E}_{191}(L-1)+0.013774\hat{E}_{191}(L-2)$.

Modelo 5b:
$\hat{Y}_{191}(L)=-0.358301+0.905725\hat{Y}_{191}(L-1)+0.59526\hat{Y}_{191}(L-2)+-0.500985\hat{Y}_{191}(L-3)+0.696618\hat{E}_{191}(L-1)+0.031185\hat{E}_{191}(L-2)$.

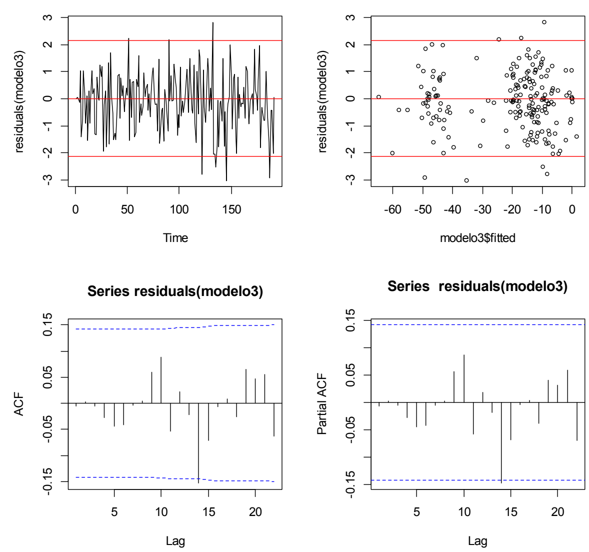
En la validación de supuestos recuerde que es sobre el error de ajuste, en este caso, E_t y por tanto, definimos en tests ACF y Ljung-Box, $\rho(k) = corr(E_t, E_{t+k})$ y en tests PACF definimos $\phi_{kk} = corr(E_t, E_{t+k} | E_{t+1}, \dots, E_{t+k-1})$, siendo los estimadores de estas funciones, respectivamente, $\hat{\rho}(k) = \widehat{corr}(E_t, E_{t+k}) = \frac{\sum_{t=1}^{191-k} \hat{E}_t \hat{E}_{t+k}}{\sum_{t=1}^{191} \hat{E}_t^2} \sim aprox.\ N\left(0, \frac{1}{191}\right)$ y $\hat{\phi}_{kk} = \widehat{corr}(E_t, E_{t+k} | E_{t+1}, \dots, E_{t+k-1}) \sim aprox.\ N\left(0, \frac{1}{191}\right)$. Para tests ACF y PACF se consideró $k = 1, 2, \dots, 22$, y para Ljung-Box $m = 6, 12, 18, 24, 30, 36$.
También test Shapiro es sobre el error de ajuste E_t.

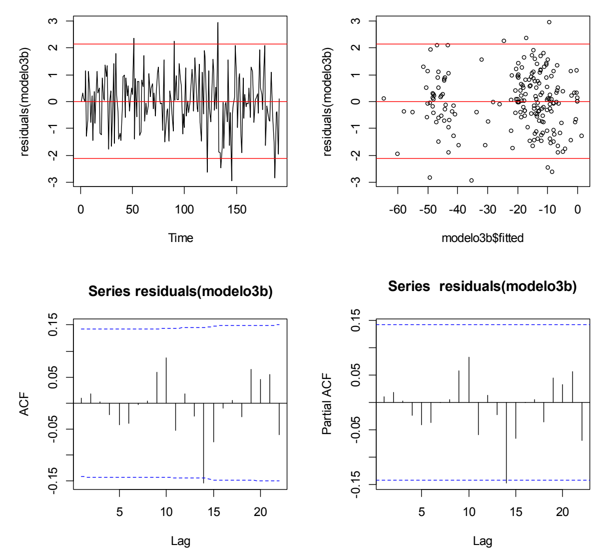




Test Ljung-Box Modelo 1			
m	Q_{LB}	df	$P(\chi^2_m > Q_{LB})$
6	4.680018	6	0.5854577
12	7.848533	12	0.7968522
18	13.463216	18	0.7633533
24	17.030447	24	0.8473606
30	29.202929	30	0.5069558
36	37.839494	36	0.3853181

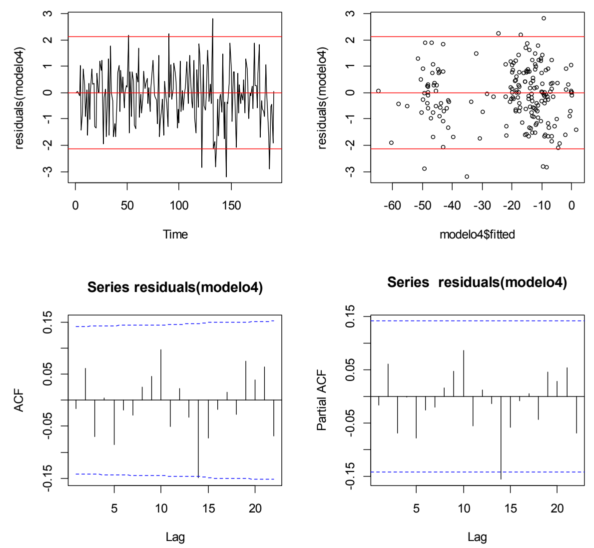
Test Ljung-Box Modelo 2			
m	Q_{LB}	df	$P(\chi^2_m > Q_{LB})$
6	4.341914	6	0.6305113
12	7.340454	12	0.8343196
18	13.158280	18	0.7820855
24	16.621000	24	0.8644378
30	28.807872	30	0.5277057
36	36.955505	36	0.4246057





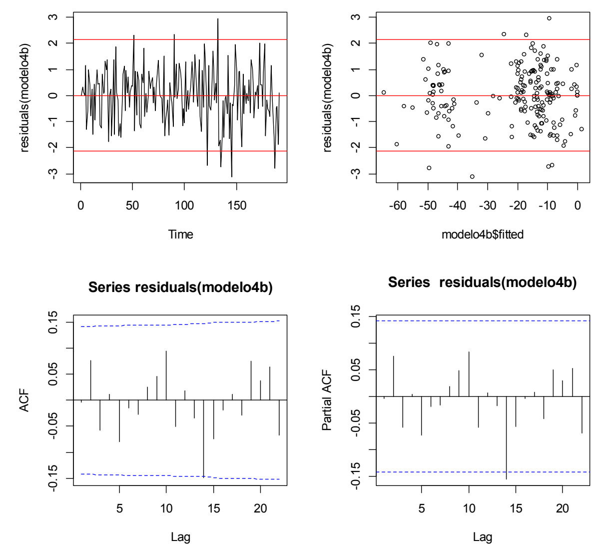
Test Ljung-Box Modelo 3			
m	Q_{LB}	df	$P(\chi^2_m > Q_{LB})$
6	0.899187	6	0.9891469
12	3.923219	12	0.9847823
18	10.176526	18	0.9259919
24	14.156606	24	0.9430339
30	23.673936	30	0.7865722
36	29.130220	36	0.7845763

Test Ljung-Box Modelo 3b			
m	Q_{LB}	df	$P(\chi^2_m > Q_{LB})$
6	0.827454	6	0.991317
12	3.698818	12	0.988279
18	10.089754	18	0.928937
24	13.915574	24	0.948534
30	23.637825	30	0.788158
36	28.941683	36	0.792000



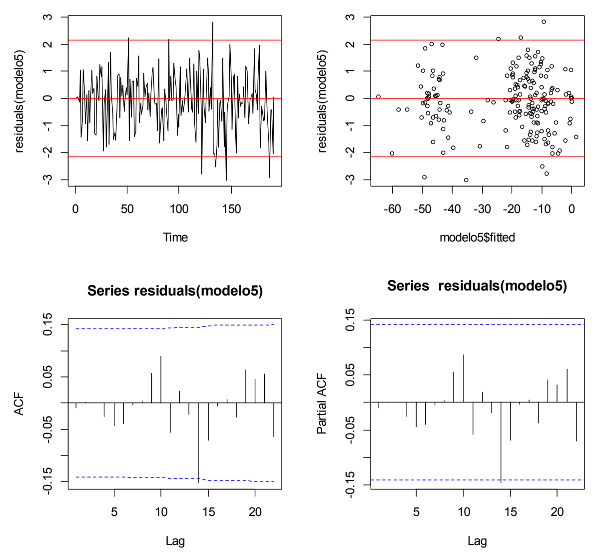
Test Ljung-Box Modelo 4

m	Q_{LB}	df	$P(\chi^2_m > Q_{LB})$
6	3.280698	6	0.7728806
12	6.553030	12	0.8856846
18	12.768151	18	0.8051582
24	16.641109	24	0.8635597
30	27.365525	30	0.6040198
36	34.494714	36	0.5402204



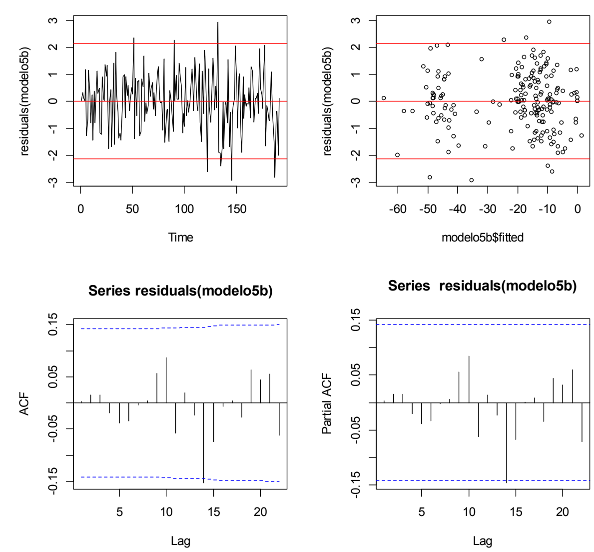
Test Ljung-Box Modelo 4b

m	Q_{LB}	df	$P(\chi^2_m > Q_{LB})$
6	3.125207	6	0.792976
12	6.222879	12	0.904432
18	12.528925	18	0.818764
24	16.257405	24	0.878562
30	27.231299	30	0.611101
36	34.128412	36	0.557841



Test Ljung-Box Modelo 5

m	Q_{LB}	df	$P(\chi^2_m > Q_{LB})$
6	0.834551	6	0.991115
12	3.895069	12	0.985256
18	10.103823	18	0.928465
24	14.089916	24	0.944593
30	23.737946	30	0.783749
36	29.226580	36	0.780736



Test Ljung-Box Modelo 5b

m	Q_{LB}	df	$P(\chi^2_m > Q_{LB})$
6	0.703133	6	0.994423
12	3.651127	12	0.988941
18	9.949504	18	0.933542
24	13.784760	24	0.951363
30	23.805434	30	0.780755
36	29.195847	36	0.781964

Código R 1.0: Cargando librerías y definiendo funciones de usuario

```
library(forecast); library(timsac); library(TSA); library(lmtest)

#Creando función usuario para obtener test Box-Pierce y Ljung-Box
BP.LB.test=function(serie,maxlag,type="Box"){
aux=floor(maxlag/6);
X.squared=c(rep(NA,aux))
df=c(rep(NA,aux))
p.value=c(rep(NA,aux))
for(i in 1:aux){
test=Box.test(serie,lag=(6*i),type=type)
X.squared[i]=test[[1]]
df[i]=test[[2]]
p.value[i]=test[[3]]
}
lag=6*c(1:aux)
teste=as.data.frame(cbind(X.squared,df,p.value))
rownames(teste)=lag
teste
}

#Creando función usuario crit.inf.resid() para calcular C_n^*(p)
crit.inf.resid=function(residuales,n.par,AIC="TRUE"){
if(AIC=="TRUE"){
#Calcula AIC
CI=log(mean(residuales^2))+2*n.par/length(residuales)
}
if(AIC=="FALSE"){
#Calcula BIC
CI=log(mean(residuales^2))+n.par*log(length(residuales))/length(residuales)
}
CI
}

#Función para calcular la amplitud de los I.P
amplitud=function(LIP,LSP){
a=LSP-LIP
am=mean(a)
am
}



#Función para calcular la cobertura de los I.P
cobertura=function(real,LIP,LSP){
I=ifelse(real>=LIP & real<=LSP,1,0)
p=mean(I)
p
}
```

Código R 2.0: Lectura datos y gráficos serie, primera diferencia y sus ACF's muestrales

```
ARIMA1.1.0=ts(scan(file.choose(),skip=1),freq=1)#leer ARIMA110.SIMUL.txt
layout(rbind(c(1,1,2,2),c(3,3,4,4)))
plot(ARIMA1.1.0)
acf(ARIMA1.1.0,ci.type="ma")
plot(diff(ARIMA1.1.0))
abline(h=mean(diff(ARIMA1.1.0)))
abline(h=c(-2*sd(diff(ARIMA1.1.0)),2*sd(diff(ARIMA1.1.0))),lty=2)
acf(diff(ARIMA1.1.0),ci.type="ma")
```

Código R 3.0: Identificación de modelos ARIMA usando sólo los primeros n=191 datos

```
n=length(ARIMA1.1.0)-10
t=1:n
yt=ts(ARIMA1.1.0[t],freq=1) #serie con sólo 191 datos

layout(rbind(c(1,1,2,2),c(3,3,4,4)))
plot(diff(yt))
abline(h=mean(diff(ARIMA1.1.0)))
abline(h=c(-2*sd(diff(ARIMA1.1.0)),2*sd(diff(ARIMA1.1.0))),lty=2)
plot(time(diff(yt)),as.numeric(diff(yt)),xlab="Time")
abline(h=mean(diff(ARIMA1.1.0)))
abline(h=c(-2*sd(diff(ARIMA1.1.0)),2*sd(diff(ARIMA1.1.0))),lty=2)
acf(diff(yt),ci.type="ma")
pacf(diff(yt))
eacf(diff(yt))

win.graph()
plot(armasubsets(diff(yt),nar=12,nma=12,y.name='test',ar.method='ml'))

autoarmafit(diff(yt)) #identificando ARMA's sobre serie diferencia
auto.arima(diff(yt)) #auto.arima sobre serie diferencia identifica ARIMA(p,0,q)
auto.arima(yt) #auto.arima sobre serie sin diferenciar identifica ARIMA(p,d,q)

mean(diff(yt)) #verificando media
```

Código R 4.0: Ajustes modelos ARIMA usando sólo los primeros n=191 datos (Valores P para significancia de parámetros calculados bajo N(0,1))

4.1 Modelo 1: ARIMA(1,1,0) sin deriva

```
modelo1=Arima(yt,order=c(1,1,0),method="ML")
coeftest(modelo1)

layout(rbind(c(1,2),c(3,4)))
plot.ts(residuals(modelo1))
abline(h=c(-2*sqrt(modelo1$sigma2),0,2*sqrt(modelo1$sigma2)),col=2)
plot(modelo1$fitted,residuals(modelo1))
abline(h=c(-2*sqrt(modelo1$sigma2),0,2*sqrt(modelo1$sigma2)),col=2)
acf(residuals(modelo1),ci.type="ma")
pacf(residuals(modelo1))

BP.LB.test(residuals(modelo1),maxlag=36,type="Ljung")
shapiro.test(residuals(modelo1))
```

4.2. Modelo 2: ARIMA(1,1,0) con deriva

```
modelo2=Arima(yt,order=c(1,1,0),include.drift=TRUE,method="ML")
coeftest(modelo2)

layout(rbind(c(1,2),c(3,4)))
plot.ts(residuals(modelo2))
abline(h=c(-2*sqrt(modelo2$sigma2),0,2*sqrt(modelo2$sigma2)),col=2)
plot(modelo2$fitted,residuals(modelo2))
abline(h=c(-2*sqrt(modelo2$sigma2),0,2*sqrt(modelo2$sigma2)),col=2)
acf(residuals(modelo2),ci.type="ma")
pacf(residuals(modelo2))

BP.LB.test(residuals(modelo2),maxlag=36,type="Ljung")
shapiro.test(residuals(modelo2))
```

4.3. Modelo3: ARIMA(2,1,1) sin deriva

```
modelo3=Arima(yt,order=c(2,1,1),method="ML")
coeftest(modelo3)

layout(rbind(c(1,2),c(3,4)))
plot.ts(residuals(modelo3))
abline(h=c(-2*sqrt(modelo3$sigma2),0,2*sqrt(modelo3$sigma2)),col=2)
plot(modelo3$fitted,residuals(modelo3))
abline(h=c(-2*sqrt(modelo3$sigma2),0,2*sqrt(modelo3$sigma2)),col=2)
acf(residuals(modelo3),ci.type="ma")
pacf(residuals(modelo3))

BP.LB.test(residuals(modelo3),maxlag=36,type="Ljung")
shapiro.test(residuals(modelo3))
```

4.4. Modelo3b: ARIMA(2,1,1) con deriva

```
modelo3b=Arima(yt,order=c(2,1,1),include.drift=TRUE,method="ML")
coeftest(modelo3b)

layout(rbind(c(1,2),c(3,4)))
plot.ts(residuals(modelo3b))
abline(h=c(-2*sqrt(modelo3b$sigma2),0,2*sqrt(modelo3b$sigma2)),col=2)
plot(modelo3b$fitted,residuals(modelo3b))
abline(h=c(-2*sqrt(modelo3b$sigma2),0,2*sqrt(modelo3b$sigma2)),col=2)
acf(residuals(modelo3b),ci.type="ma")
pacf(residuals(modelo3b))

BP.LB.test(residuals(modelo3b),maxlag=36,type="Ljung")
shapiro.test(residuals(modelo3b))
```

4.5. Modelo4: ARIMA(1,1,1) sin deriva

```
modelo4=Arima(yt,order=c(1,1,1),method="ML")
coeftest(modelo4)

layout(rbind(c(1,2),c(3,4)))
plot.ts(residuals(modelo4))
abline(h=c(-2*sqrt(modelo4$sigma2),0,2*sqrt(modelo4$sigma2)),col=2)
plot(modelo4$fitted,residuals(modelo4))
abline(h=c(-2*sqrt(modelo4$sigma2),0,2*sqrt(modelo4$sigma2)),col=2)
acf(residuals(modelo4),ci.type="ma")
pacf(residuals(modelo4))

BP.LB.test(residuals(modelo4),maxlag=36,type="Ljung")
shapiro.test(residuals(modelo4))
```

4.6. Modelo4b: ARIMA(1,1,1) con deriva

```
modelo4b=Arima(yt,order=c(1,1,1),include.drift=TRUE,method="ML")
coeftest(modelo4b)

layout(rbind(c(1,2),c(3,4)))
plot.ts(residuals(modelo4b))
abline(h=c(-2*sqrt(modelo4b$sigma2),0,2*sqrt(modelo4b$sigma2)),col=2)
plot(modelo4b$fitted,residuals(modelo4b))
abline(h=c(-2*sqrt(modelo4b$sigma2),0,2*sqrt(modelo4b$sigma2)),col=2)
acf(residuals(modelo4b),ci.type="ma")
pacf(residuals(modelo4b))
```

```
BP.LB.test(residuals(modelo4b),maxlag=36,type="Ljung")
shapiro.test(residuals(modelo4b))
```

4.7. Modelo5: ARIMA(2,1,2) sin deriva

```
modelo5=Arima(yt,order=c(2,1,2),method="ML")
coeftest(modelo5)

layout(rbind(c(1,2),c(3,4)))
plot.ts(residuals(modelo5))
abline(h=c(-2*sqrt(modelo5$sigma2),0,2*sqrt(modelo5$sigma2)),col=2)
plot(modelo5$fitted,residuals(modelo5))
abline(h=c(-2*sqrt(modelo5$sigma2),0,2*sqrt(modelo5$sigma2)),col=2)
acf(residuals(modelo5),ci.type="ma")
pacf(residuals(modelo5))

BP.LB.test(residuals(modelo5),maxlag=36,type="Ljung")
shapiro.test(residuals(modelo5))
```

4.8. Modelo5b: ARIMA(2,1,2) con deriva

```
modelo5b=Arima(yt,order=c(2,1,2),include.drift=T,method="ML")
coeftest(modelo5b)

layout(rbind(c(1,2),c(3,4)))
plot.ts(residuals(modelo5b))
abline(h=c(-2*sqrt(modelo5b$sigma2),0,2*sqrt(modelo5b$sigma2)),col=2)
plot(modelo5b$fitted,residuals(modelo5b))
abline(h=c(-2*sqrt(modelo5b$sigma2),0,2*sqrt(modelo5b$sigma2)),col=2)
acf(residuals(modelo5b),ci.type="ma")
pacf(residuals(modelo5b))

BP.LB.test(residuals(modelo5b),maxlag=36,type="Ljung")
shapiro.test(residuals(modelo5b))
```

4.9 Gráficas de los ajustes y medidas de bondad de ajuste

```
win.graph(width=16,height=8)
layout(rbind(c(1,2,3,4),c(5,6,7,8)))plot(ARIMA1.1.0)
lines(modelo1$fitted,col=2,lwd=2)
legend("bottomleft",legend=c("Original","Ajuste modelo 1"),col=1:2,lty=1)
plot(ARIMA1.1.0)
lines(modelo2$fitted,col=2,lwd=2)
legend("bottomleft",legend=c("Original","Ajuste modelo 2"),col=1:2,lty=1)
plot(ARIMA1.1.0)
lines(modelo3$fitted,col=2,lwd=2)
legend("bottomleft",legend=c("Original","Ajuste modelo 3"),col=1:2,lty=1)
plot(ARIMA1.1.0)
lines(modelo3b$fitted,col=2,lwd=2)
legend("bottomleft",legend=c("Original","Ajuste modelo 3b"),col=1:2,lty=1)
plot(ARIMA1.1.0)
lines(modelo4$fitted,col=2,lwd=2)
legend("bottomleft",legend=c("Original","Ajuste modelo 4"),col=1:2,lty=1)
plot(ARIMA1.1.0)
lines(modelo4b$fitted,col=2,lwd=2)
legend("bottomleft",legend=c("Original","Ajuste modelo 4b"),col=1:2,lty=1)
plot(ARIMA1.1.0)
lines(modelo5$fitted,col=2,lwd=2)
legend("bottomleft",legend=c("Original","Ajuste modelo 5"),col=1:2,lty=1)
plot(ARIMA1.1.0)
lines(modelo5b$fitted,col=2,lwd=2)
legend("bottomleft",legend=c("Original","Ajuste modelo 5b"),col=1:2,lty=1)

aic1=exp(crit.inf.resid(residuals(modelo1),n.par=1))
aic2=exp(crit.inf.resid(residuals(modelo2),n.par=2))
aic3=exp(crit.inf.resid(residuals(modelo3),n.par=3))
aic3b=exp(crit.inf.resid(residuals(modelo3b),n.par=4))
aic4=exp(crit.inf.resid(residuals(modelo4),n.par=2))
aic4b=exp(crit.inf.resid(residuals(modelo4b),n.par=3))
aic5=exp(crit.inf.resid(residuals(modelo5),n.par=4))
aic5b=exp(crit.inf.resid(residuals(modelo5b),n.par=5))

bic1=exp(crit.inf.resid(residuals(modelo1),n.par=1,AIC="FALSE"))
bic2=exp(crit.inf.resid(residuals(modelo2),n.par=2,AIC="FALSE"))
bic3=exp(crit.inf.resid(residuals(modelo3),n.par=3,AIC="FALSE"))
bic3b=exp(crit.inf.resid(residuals(modelo3b),n.par=4,AIC="FALSE"))
bic4=exp(crit.inf.resid(residuals(modelo4),n.par=2,AIC="FALSE"))
bic4b=exp(crit.inf.resid(residuals(modelo4b),n.par=3,AIC="FALSE"))
bic5=exp(crit.inf.resid(residuals(modelo5),n.par=4,AIC="FALSE"))
bic5b=exp(crit.inf.resid(residuals(modelo5b),n.par=5,AIC="FALSE"))

criterios=data.frame(p=c(1,2,3,4,2,3,4,5),AIC=c(aic1,aic2,aic3,aic3b,aic4,aic4b,aic5,aic5b),BIC=c(bic1,bic2,bic3,bic3b,bic4,bic4
b,bic5,bic5b),row.names=c("modelo1","modelo2","modelo3","modelo3b","modelo4","modelo4b","modelo5","modelo5b"))
criterios
```

4.10 Predicciones de las últimas 10 observaciones con I.P del 95% y medidas de precisión pronósticos

```
tnuevo=(n+1):length(ARIMA1.1.0) #valor índice de tiempo en los pronósticos
ynuevo=ts(ARIMA1.1.0[tnuevo],freq=1,start=192) #valor serie en las últimos 10 tiempo

pred1=forecast(modelo1,h=10,level=95)
pred1=cbind(Predic=pred1$mean,LIP95=pred1$lower,LSP95=pred1$upper)
pred2=forecast(modelo2,h=10,level=95)
pred2=cbind(Predic=pred2$mean,LIP95=pred2$lower,LSP95=pred2$upper)
pred3=forecast(modelo3,h=10,level=95)
pred3=cbind(Predic=pred3$mean,LIP95=pred3$lower,LSP95=pred3$upper)
pred3b=forecast(modelo3b,h=10,level=95)
pred3b=cbind(Predic=pred3b$mean,LIP95=pred3b$lower,LSP95=pred3b$upper)
pred4=forecast(modelo4,h=10,level=95)
pred4=cbind(Predic=pred4$mean,LIP95=pred4$lower,LSP95=pred4$upper)
pred4b=forecast(modelo4b,h=10,level=95)
pred4b=cbind(Predic=pred4b$mean,LIP95=pred4b$lower,LSP95=pred4b$upper)
pred5=forecast(modelo5,h=10,level=95)
pred5=cbind(Predic=pred5$mean,LIP95=pred5$lower,LSP95=pred5$upper)
pred5b=forecast(modelo5b,h=10,level=95)
pred5b=cbind(Predic=pred5b$mean,LIP95=pred5b$lower,LSP95=pred5b$upper)

#Gráfico comparativo de predicciones puntuales vs. valores reales
plot(ynuevo,col="brown",lwd=2,type="b",pch=19,ylim=c(-73,-65))
lines(pred1[,1],col=2,lwd=2,type="b",pch=1)
lines(pred2[,1],col=3,lwd=2,type="b",pch=2)
lines(pred3[,1],col=4,lwd=2,type="b",pch=3)
lines(pred3b[,1],col=5,lwd=2,type="b",pch=4)
lines(pred4[,1],col=6,lwd=2,type="b",pch=5)
lines(pred4b[,1],col=7,lwd=2,type="b",pch=6)
lines(pred5[,1],col=8,lwd=2,type="b",pch=7)
lines(pred5b[,1],col=9,lwd=2,type="b",pch=8)
legend("topright",legend=c("Original","pred1","Pred2","Pred3","pred3b","Pred4","Pred4b","Pred5","Pred5b"),col=c("brown",2:9),
pch=c(19,1:8),lty=1,lwd=2)

#Amplitud I.P
amp1=amplitud(LIP=pred1[,2],LSP=pred1[,3])
amp2=amplitud(LIP=pred2[,2],LSP=pred2[,3])
amp3=amplitud(LIP=pred3[,2],LSP=pred3[,3])
amp3b=amplitud(LIP=pred3b[,2],LSP=pred3b[,3])
amp4=amplitud(LIP=pred4[,2],LSP=pred4[,3])
amp4b=amplitud(LIP=pred4b[,2],LSP=pred4b[,3])
amp5=amplitud(LIP=pred5[,2],LSP=pred5[,3])
amp5b=amplitud(LIP=pred5b[,2],LSP=pred5b[,3])

#Cobertura I.P
cob1=cobertura(real=ynuevo,LIP=pred1[,2],LSP=pred1[,3])
cob2=cobertura(real=ynuevo,LIP=pred2[,2],LSP=pred2[,3])
cob3=cobertura(real=ynuevo,LIP=pred3[,2],LSP=pred3[,3])
cob3b=cobertura(real=ynuevo,LIP=pred3b[,2],LSP=pred3b[,3])
cob4=cobertura(real=ynuevo,LIP=pred4[,2],LSP=pred4[,3])
cob4b=cobertura(real=ynuevo,LIP=pred4b[,2],LSP=pred4b[,3])
cob5=cobertura(real=ynuevo,LIP=pred5[,2],LSP=pred5[,3])
cob5b=cobertura(real=ynuevo,LIP=pred5b[,2],LSP=pred5b[,3])

#Tabla con todas las medidas de precisión de pronósticos
tablaprec=data.frame(AmplitudI.P=c(amp1,amp2,amp3,amp3b,amp4,amp4b,amp5,amp5b),Cobertura=c(cob1,cob2,cob3,cob3b,cob4,cob4b,cob5,
cob5b)*100,
RMSE=c(accuracy(pred1[,1],ynuevo)[2],accuracy(pred2[,1],ynuevo)[2],accuracy(pred3[,1],ynuevo)[2],accuracy(pred3b[,1],ynuevo)[2],
accuracy(pred4[,1],ynuevo)[2],accuracy(pred4b[,1],ynuevo)[2],accuracy(pred5[,1],ynuevo)[2],accuracy(pred5b[,1],ynuevo)[2]),
MAE=c(accuracy(pred1[,1],ynuevo)[3],accuracy(pred2[,1],ynuevo)[3],accuracy(pred3[,1],ynuevo)[3],accuracy(pred3b[,1],ynuevo)[3],a
ccuracy(pred4[,1],ynuevo)[3],accuracy(pred4b[,1],ynuevo)[3],accuracy(pred5[,1],ynuevo)[3],accuracy(pred5b[,1],ynuevo)[3]),
MAPE=c(accuracy(pred1[,1],ynuevo)[5],accuracy(pred2[,1],ynuevo)[5],accuracy(pred3[,1],ynuevo)[5],accuracy(pred3b[,1],ynuevo)[5],
accuracy(pred4[,1],ynuevo)[5],accuracy(pred4b[,1],ynuevo)[5],accuracy(pred5[,1],ynuevo)[5],accuracy(pred5b[,1],ynuevo)[5]),
row.names=c("modelo1","modelo2","modelo3","modelo3b","modelo4","modelo4b","modelo5","modelo5b"))
round(tablaprec,digits=2)
```