Expo

André Marx Puente Arévalo

5/11/2020

Paridad Peso-Dolar

Primero cargamos las paqueterias que usaremos

```
library(descomponer)
library(tseries)
library(stats)
library(forecast)
library(aTSA)
library(FinTS)
library(fGarch)
library(astsa)
```

Cargamos el csv con los precios historicos diarios del 2015 al 2020 obtenidos de Yahoo Finanza y los hacemos una serie de tiempo.

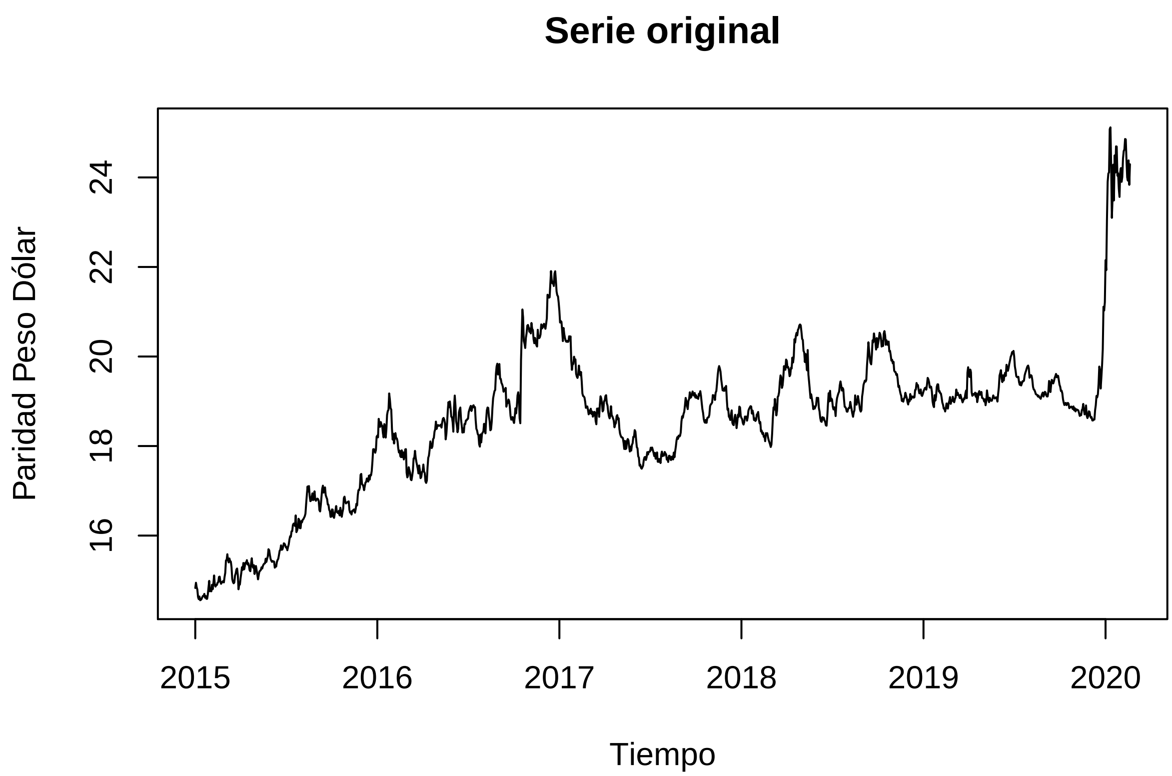
```
data<-read.csv("Historico.csv")
data1<-data$Niveles
tserie<-ts(data1, frequency = 261, start = 2015)
```

Obtenemos un resumen de los datos y su gráfica.

```
summary(tserie)
```

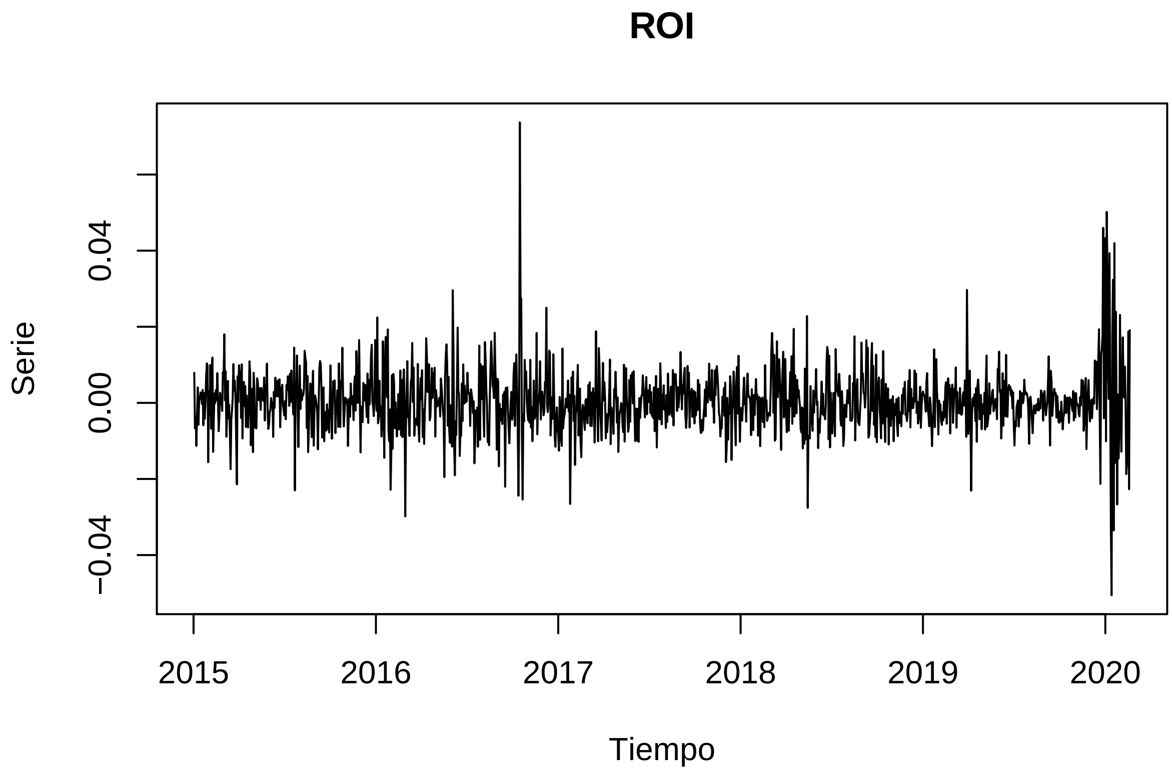
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   14.56   17.86   18.83   18.57   19.30   25.12
```

```
plot(tserie, main= "Serie original", ylab = "Paridad Peso Dólar", xlab="Tiempo")
```



A los datos les aplicamos logaritmo para reducir la varianza de la serie y luego aplicamos una diferencia para quitarle la tendencia a nuestra serie de tiempo.

```
LnTseries <- log(tserie)
seriedif<-diff(LnTseries)
plot(seriedif, main = "ROI", ylab = "Serie", xlab = "Tiempo")
```



Hasta aquí hemos obtenido el ROI de la serie.

Ahora le haremos la prueba de hipótesis de *Ljung-Box* a nuestros datos, la cual, tiene las siguientes hipótesis:

$H_0 : \rho_t = 0$ vs $H_1 : \rho_t \neq 0$

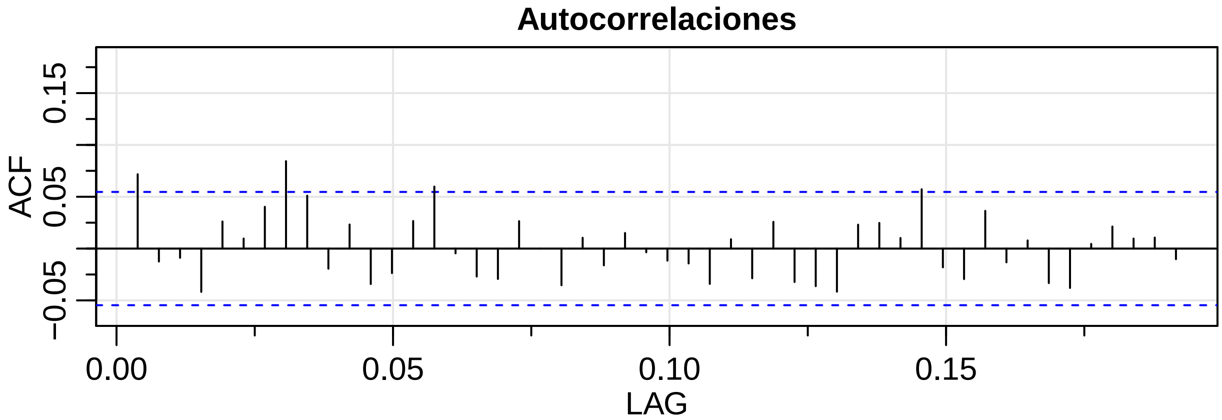
```r
Box.test(seriedif, type = "Ljung-Box")
```

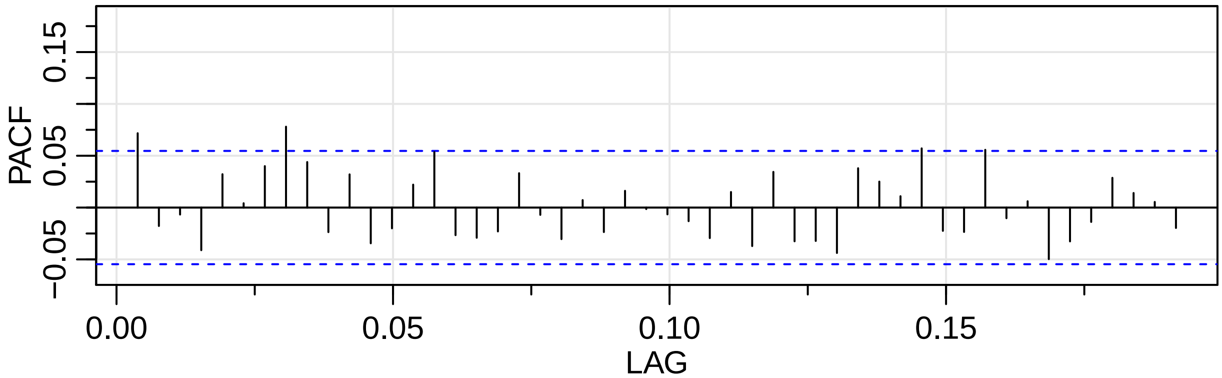
```
##
##  Box-Ljung test
##
## data:  seriedif
## X-squared = 6.9047, df = 1, p-value = 0.008597
```

Dado que el $p - value$ obtenido es menor a un nivel de significancia $\alpha = 0.05$ rechazamos la hipótesis nula (H_0), es decir, la muestra no se distribuye de forma independiente.

Ahora, realizamos la ACf y la PACF

```r
acf2(seriedif, 50, main = "Autocorrelaciones")
```





Observando las autocorrelaciones parciales podemos proponer un modelo que en la parte Auto Regresiva tenga un parámetro y analizando las autocorrelaciones simple proponemos que el modelo tenga en la parte de Medias Móviles un parámetro.

Los modelos propuestos son los siguientes:

```
modelo1 <- arima(seriedif, order = c(1,0,0))
modelo2 <- arima(seriedif, order = c(0,0,1))
modelo3 <- arima(seriedif, order = c(1,0,1))
modelo4 <- arima(seriedif, order = c(2,0,0))
modelo5 <- arima(seriedif, order = c(2,0,1))
modelo6 <- arima(seriedif, order = c(1,0,2))
modelo7 <- arima(seriedif, order = c(2,0,2))
modelo8 <- arima(seriedif, order = c(0,0,2))

laquetuquieras <- data.frame("AIC"=c(modelo1$aic, modelo2$aic, modelo3$aic,
                                     modelo4$aic, modelo5$aic, modelo6$aic,
                                     modelo7$aic, modelo8$aic))
row.names(laquetuquieras)=c("ARIMA(1, 0, 0)", "ARIMA(0, 0, 1)", "ARIMA(1, 0, 1)",
                            "ARIMA(2, 0, 0)", "ARIMA(2, 0, 1)", "ARIMA(1, 0, 2)",
                            "ARIMA(2, 0, 2)", "ARIMA(0, 0, 2)")
laquetuquieras
```

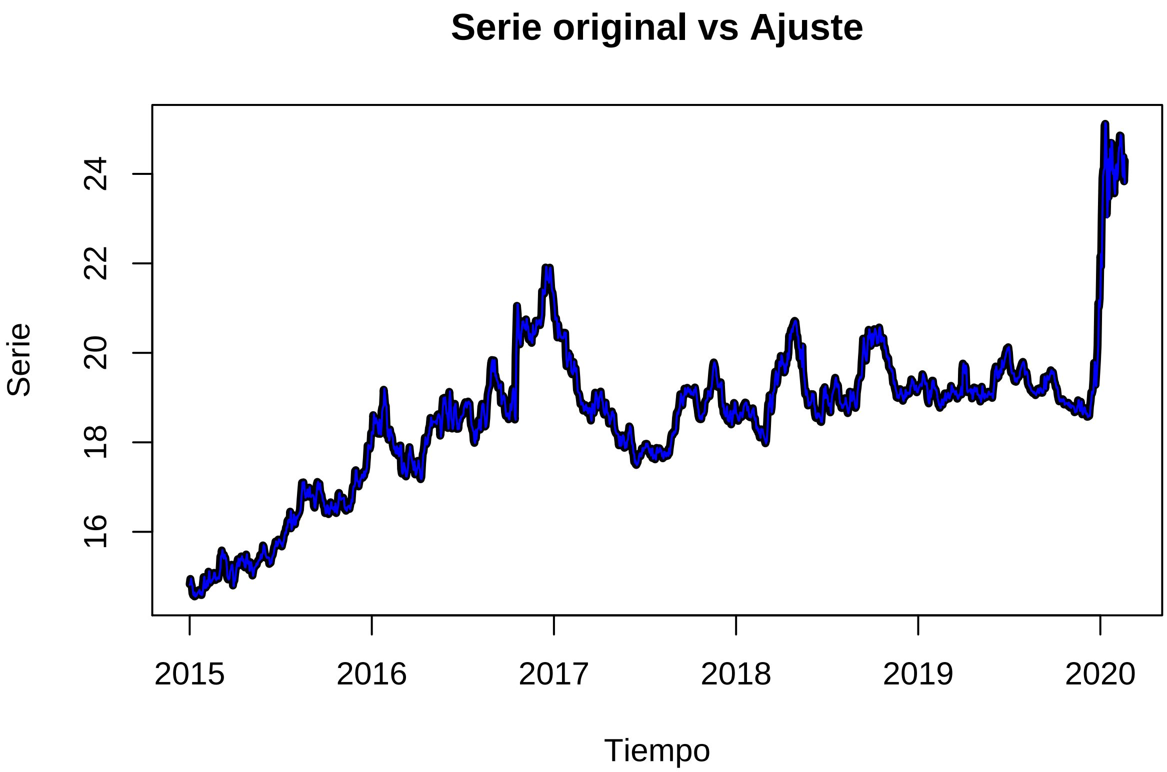
```
##                      AIC
## ARIMA(1, 0, 0) -9034.111
## ARIMA(0, 0, 1) -9034.330
## ARIMA(1, 0, 1) -9032.882
## ARIMA(2, 0, 0) -9032.558
```

```
## ARIMA(2, 0, 1) -9030.549
## ARIMA(1, 0, 2) -9030.567
## ARIMA(2, 0, 2) -9032.569
## ARIMA(0, 0, 2) -9032.564
```

Ajustaremos el modelo que más minimiza el AIC, es decir, nos quedamos con el ARIMA(0,1,1).

Se tiene que el siguiente grafico con la serie original y la ajustada:

```r
modeloMinimo <- arima(tserie, order = c(0,1,1))
ajuste1 <- fitted.values(modeloMinimo)

plot(tserie, lwd=4, main = "Serie original vs Ajuste", xlab = "Tiempo", ylab = "Serie")
#legend(x = 1, y=24, legend = c("Original", "Ajuste"), fill=c("black","blue"))
lines(ajuste1, col="blue", lwd = 1)
```



Ahora, obtenemos los datos del ajuste que realizamos

```r
summary(modeloMinimo)
```

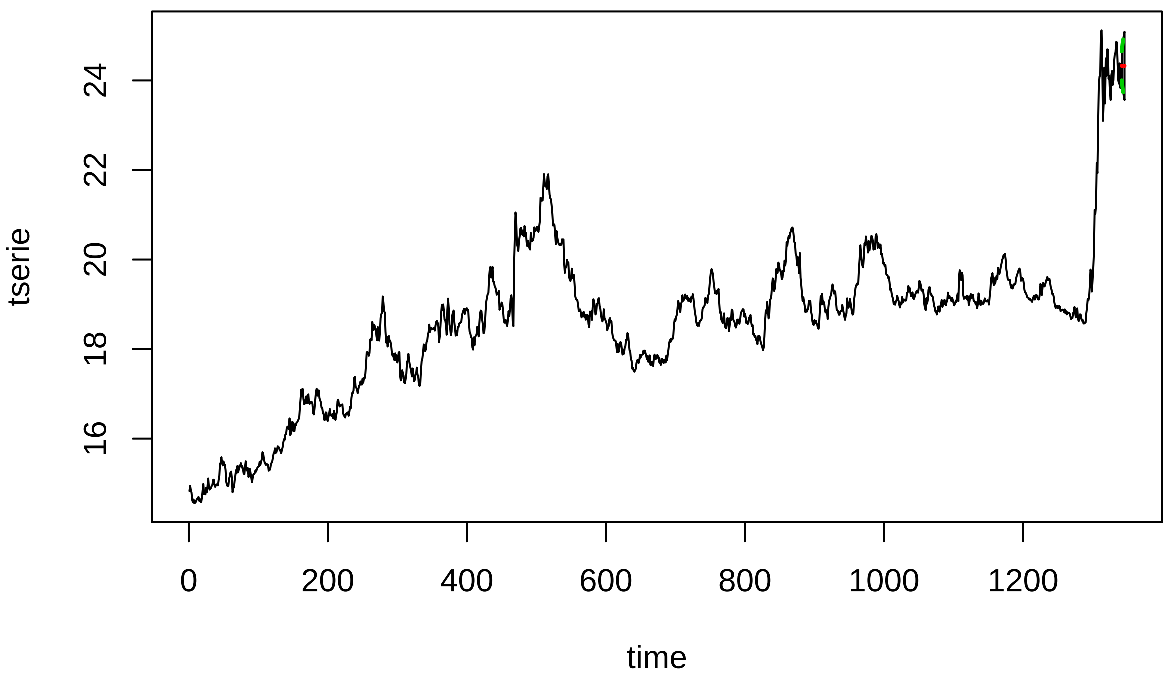
```
##
## Call:
## arima(x = tserie, order = c(0, 1, 1))
##
## Coefficients:
##          ma1
##       0.0642
## s.e.  0.0278
##
## sigma^2 estimated as 0.02738:  log likelihood = 509.36,  aic = -1014.73
##
## Training set error measures:
```

```
##                         ME      RMSE       MAE        MPE      MAPE
## Training set 0.006665712 0.1653929 0.1090818 0.03154287 0.5747658
##                   MASE          ACF1
## Training set 0.9963204 -0.002384392
```

Se tiene el siguiente comportamiento de la serie pronosticando a cinco días.

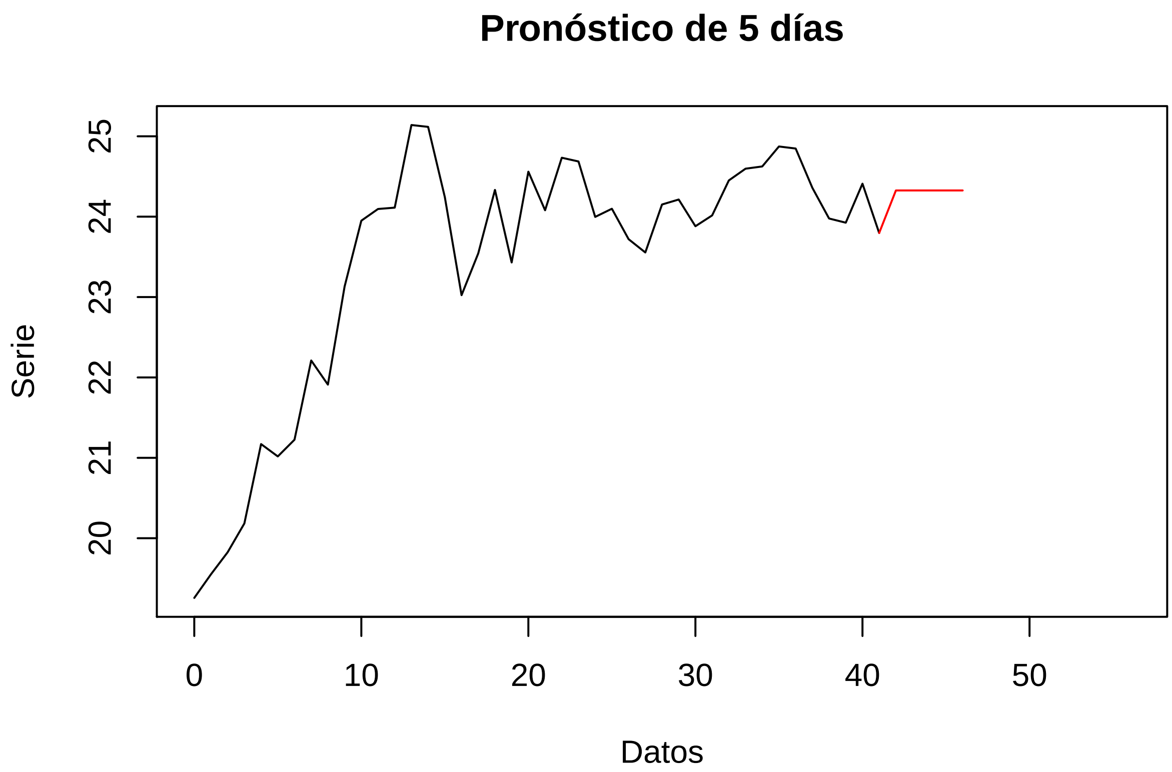
```
pronostico1 <- forecast(modeloMinimo, lead = 5)
```

```
## Forecast for univariate time series:
##      Lead Forecast   S.E Lower Upper
## 1342    1      24.3 0.165  24.0  24.7
## 1343    2      24.3 0.242  23.9  24.8
## 1344    3      24.3 0.299  23.7  24.9
## 1345    4      24.3 0.347  23.6  25.0
## 1346    5      24.3 0.389  23.6  25.1
## ------
## Note: confidence level = 95 %
```

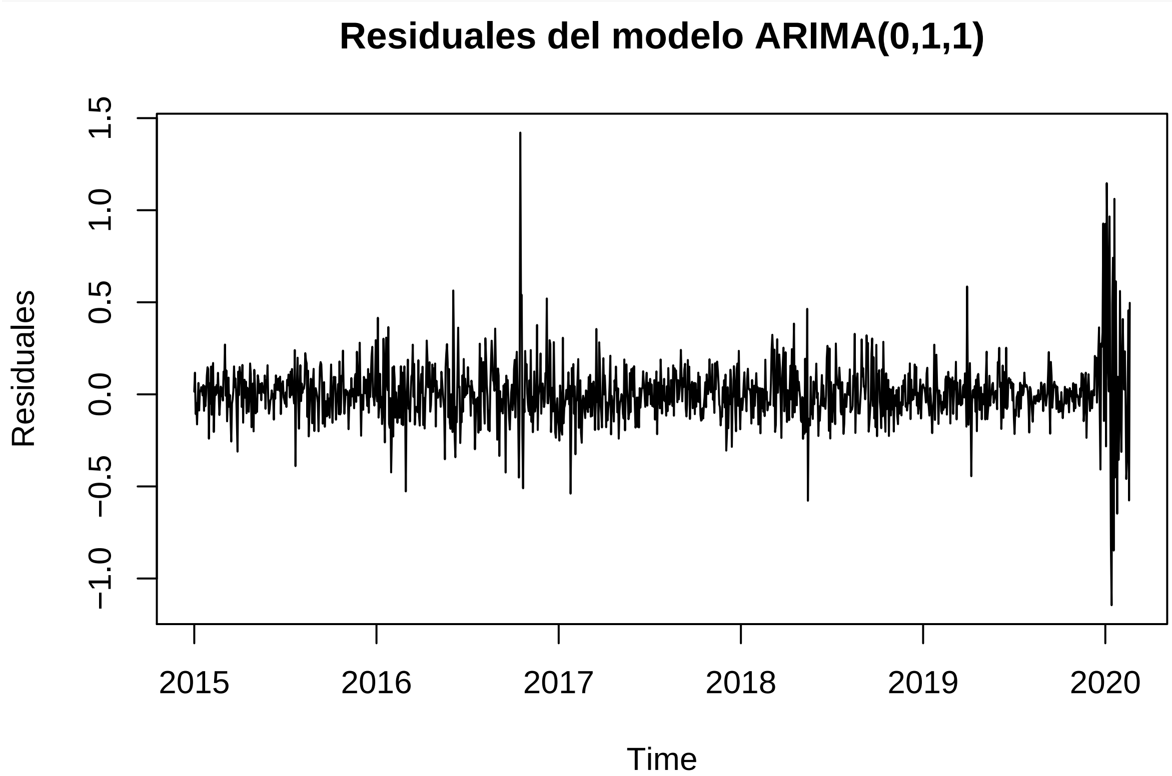


Haciendole un zoom al pronóstico se tiene lo siguiente:

```
vectorAjuste1 <- as.vector(ajuste1)

vectorAjuste1[length(vectorAjuste1)+1:dim(pronostico1)[1]] <- pronostico1[,2]

plot(x=c(0:41),y=vectorAjuste1[1300:1341],type="l",xlim=c(0,56), ylab = "Serie",
     xlab = "Datos", main = "Pronóstico de 5 días")
lines(x=c(41:56),y=vectorAjuste1[1341:1356],type="l",col="red")
```




```r
# Graficamos los residuales de los modelos obtenidos
plot(modeloMinimo$residuals, main = "Residuales del modelo ARIMA(0,1,1)",  ylab = "Residuales")
```



Tras analizar la grafica de los residuales de ambos modelos, nos damos cuenta que presentan picos muy extraños, lo que nos lleva a pensar que se podría modelar la serie respecto de su varianza, es decir, modelar con un GARCH.

Vamos a ver si podemos ajustar un modelo GARCH(p, q), para estos modelos, realizaremos la prueba de hipótesis de Engle, la cual, tiene las siguientes hipótesis:

H_0 : No pressenta efecto ARCH vs H_1 : Sí presenta efecto ARCH.

Aplicando la prueba a la serie que tiene aplicado logaritmo y una diferencia, obtenemos:

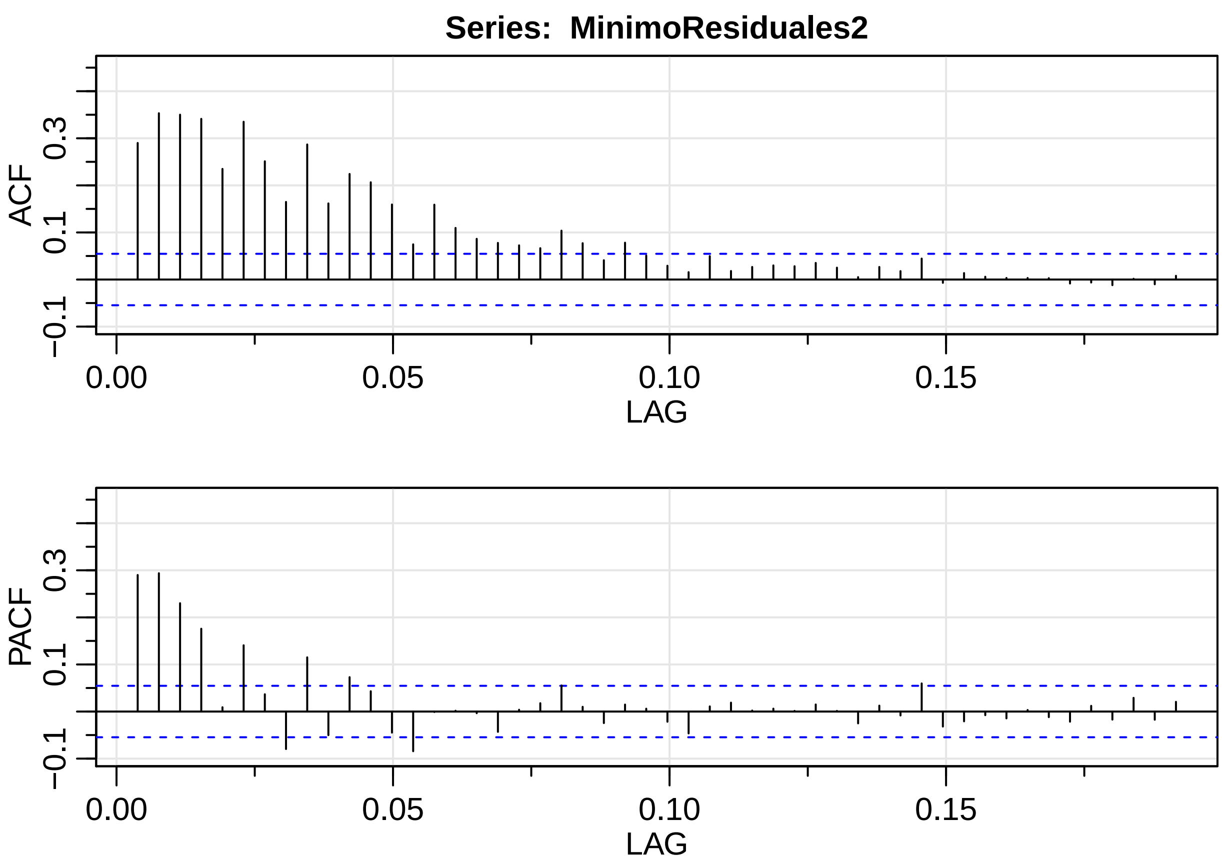
```
ArchTest(seriedif)
```

```
##
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  seriedif
## Chi-squared = 232.24, df = 12, p-value < 2.2e-16
```

Ya que el $p - value$ obtenido es menor que un nivel de significancia $\alpha = 0.05$ podemos concluir que el modelo presenta el efecto ARCH, es decir, su varianza es heterosedástica.

Ahora, procedemos a realizar el ACF y el PACF con los residuales al cuadrado de nuestro ARIMA(2,1,1).

```
# Obtenemos los residuales
MinimoResiduales <- modeloMinimo$residuals
MinimoResiduales2 <- modeloMinimo$residuals^2
 acf2(MinimoResiduales2, 50)
```



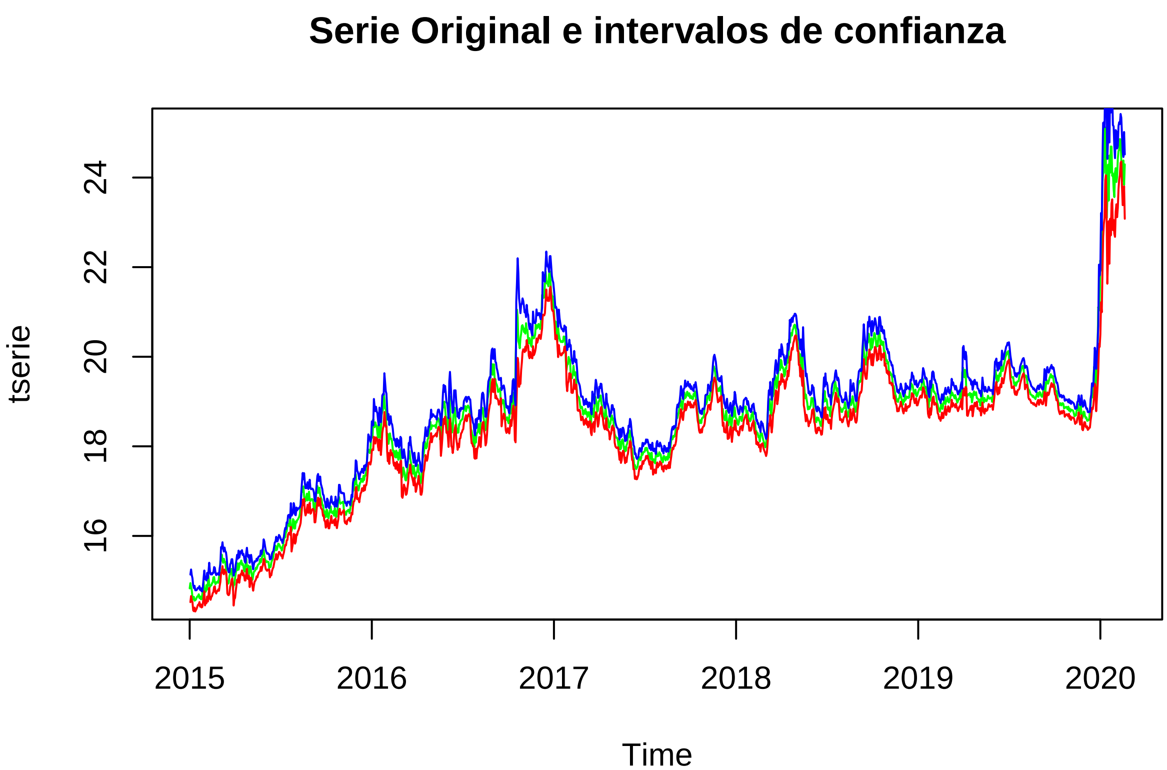

```
## Proponemos los siguientes modelos
arch01<-garch(MinimoResiduales,order=c(0,1),trace=F)
arch08<-garch(MinimoResiduales,order=c(0,8),trace=F)
arch11<-garch(MinimoResiduales,order=c(1,1),trace=F)
arch10<-garch(MinimoResiduales,order=c(1,0),trace=F)
arch02<-garch(MinimoResiduales,order=c(0,2),trace=F)
```

```r
arch22<-garch(MinimoResiduales,order=c(2,2),trace=F)
arch20<-garch(MinimoResiduales,order=c(2,0),trace=F)
arch04<-garch(MinimoResiduales,order=c(0,4),trace=F)
arch40<-garch(MinimoResiduales,order=c(4,0),trace=F)
arch44<-garch(MinimoResiduales,order=c(4,4),trace=F)
```

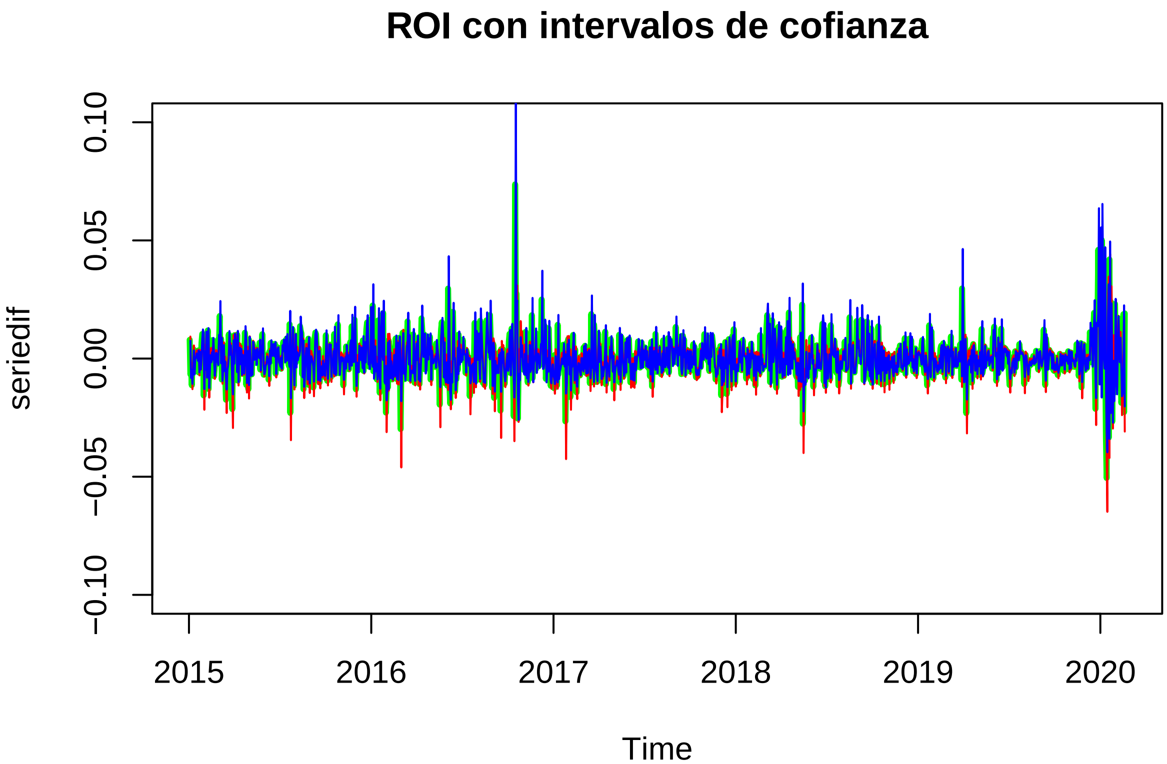
```r
aicarch01<-AIC(arch01)
aicarch08<-AIC(arch08)
aicarch11<-AIC(arch11)
aicarch10<-AIC(arch10)
aicarch02<-AIC(arch02)
aicarch22<-AIC(arch22)
aicarch20<-AIC(arch20)
aicarch04<-AIC(arch04)
aicarch40<-AIC(arch40)
aicarch44<-AIC(arch44)
```

```r
names<-c("aicarch01","aicarch08","aicarch11","aicarch10","aicarch02",
         "aicarch22","aicarch20","aicarch04","aicarch40","aicarch44")

aic2<-as.numeric(c(aicarch01,aicarch08,aicarch11,aicarch10,aicarch02,
                   aicarch22,aicarch20,aicarch04,aicarch40,aicarch44))

table2 <- data.frame(names,aic2)
table2

summary(arch11)

ht.garch11=arch11$fit[,1]^2
ht.garch11
```

```r
estimaciones_arima011=fitted.values(modeloMinimo)
estimaciones_arima011

inf_garch11= estimaciones_arima011-1.96*sqrt(ht.garch11)
sup_garch11= estimaciones_arima011+1.96*sqrt(ht.garch11)

plot(tserie, type="l", main = "Serie Original e intervalos de confianza", col="green")
lines(inf_garch11,col='red')
lines(sup_garch11,col='blue')
```




```r
dli<- diff(ts(log(inf_garch11), frequency = 261, start = 2015))
dls<- diff(ts(log(sup_garch11), frequency = 261, start = 2015))
plot(seriedif, col= "green", lwd = 3,
     main = "ROI con intervalos de cofianza",type="l", ylim = c(-0.1, 0.1))
lines(dli,col='red', type="l")
lines(dls,col='blue')
```


ROI con intervalos de cofianza
0.10
0.05
0.00
−0.05
−0.10
seriedif
2015
2016
2017
2018
2019
2020
Time