# Exercicio de clase: series, dia 9 (13-11-2019) Descarga e preparacion de datos

- Enunciado: Descargar os datos da cotización do S&P500 entre o 01-01-2000 e o 07-10-2019, e analizade:
  - a.- Que modelo ARMA se pode axustar mellor a esa serie?
  - b.- Que modelo GARCH se pode axustar mellor a esa serie?

**PRELIMINAR**: 1º descargar datos e gardalos nun ficheiro:

```
library(quantmod)
getSymbols("^GSPC",from="2000-01-01",to="2019-10-07")
```

```
## [1] "^GSPC"
```

Extraer os datos axustados e gardalos, para seguir a análise nunha segunda folla de código:

```
gspc=Ad(GSPC)
save(gspc,file = "gspc.Rdata")
```

Limpar todos os datos que están en memoria, para comezar de cero co novo código:

```
rm(list = ls())
```

### Exercicio de clase: series, dia 9 (13-11-2019) Parte ARMA

- Enunciado: Descargar os datos da cotización do S&P500 entre o 01-01-2000 e o 07-10-2019, e analizade:
  - a.- Que modelo ARMA se pode axustar mellor a esa serie?
  - b.- Que modelo GARCH se pode axustar mellor a esa serie?

```
#cargar quantmod, para ver ben os datos
library(quantmod)
#Cargar os datos gardados:
load("gspc.Rdata")
#que hai aquí?
head(gspc)
```

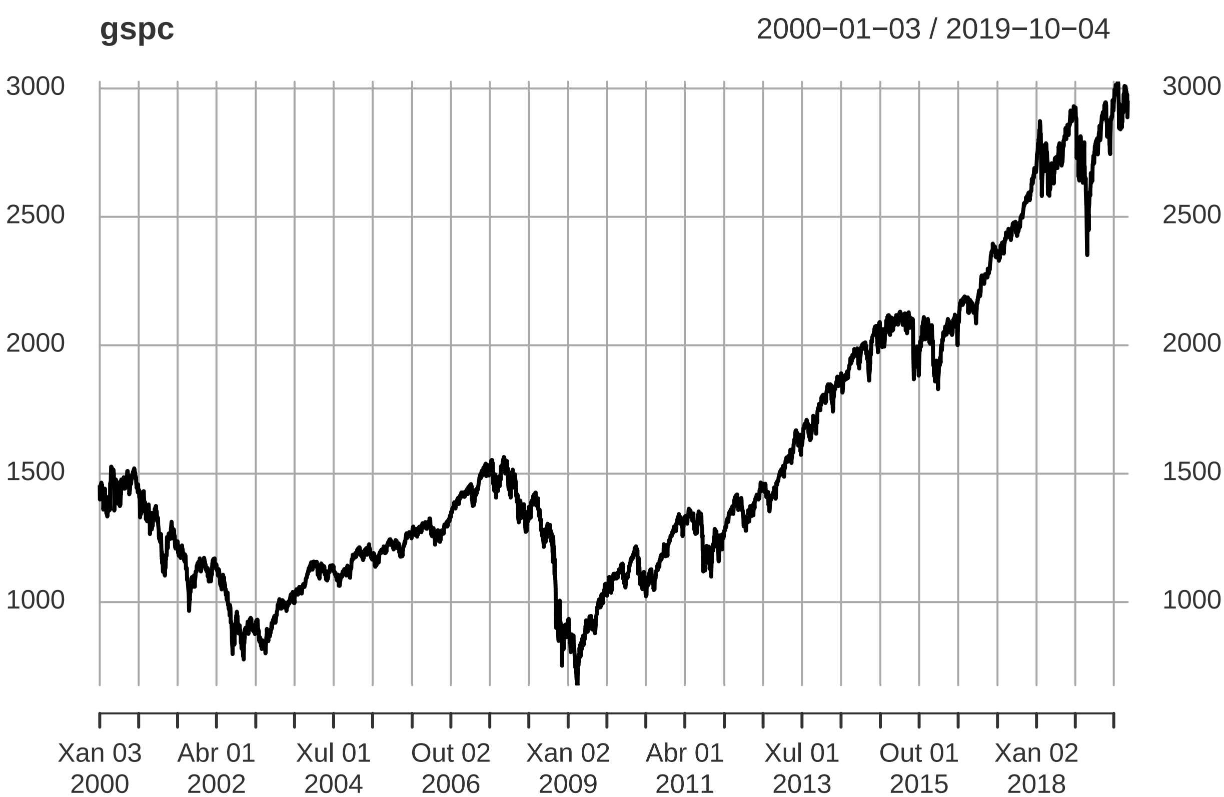
```
##            GSPC.Adjusted
## 2000-01-03       1455.22
## 2000-01-04       1399.42
## 2000-01-05       1402.11
## 2000-01-06       1403.45
## 2000-01-07       1441.47
## 2000-01-10       1457.60
```

**RESPOSTA a: Que modelo ARMA se pode axustar mellor a esa serie?**

**1a. Graficar os datos:**

**E en caso de necesidade *facer transformacións*** para ***eliminar partes integradas (I(d)) e tendencias**

```
#graficar os datos
plot(gspc)
```




O que se ve na gráfica é unha serie que **non é estacionaria**, dado que a simple vista nótase que a media non é constante en todo o rango da serie.

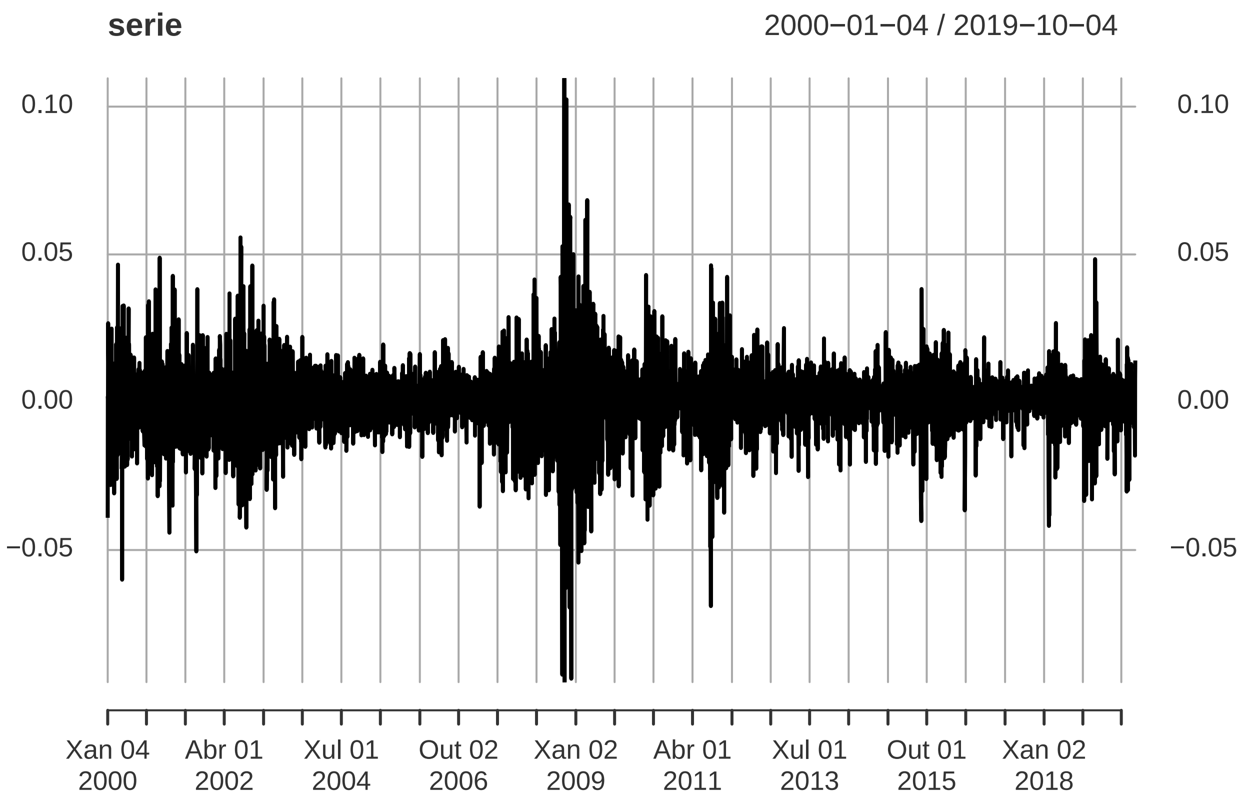
Ademais, ao ser unha serie financeira esperarase que teña heterocedasticidade, xa que un 1% de variación con 100 euros non é a mesma cantidade que con 500, e iso aféctalle á súa variabilidade (varianza).

Polo tanto aplícase logaritmos para homoxeneizar a varianza, e diferencias para intentar que se volva estacionaria:

```
# Pode facerse cun só comando de quantmod:
#calculo dos retornos dos logaritmos
serie=Delt(gspc,type="log")
#este comando sempre produce un NA na 1ª posición, que pode dar problemas mais tarde.
# mellor quitarllo:
serie=na.omit(serie)
```

Volvemos a representar graficamente a nova serie:

```
# graficar os datos
plot(serie)
```




***Contrastes de raices unitarias?***

Neste caso a gráfica parece bastante clara sobre a estacionariedade, pero en caso de dúbida poden facerse contrastes de raices unitarias para ter máis seguridade:

```
#augmented Dickey-Fuller, Phillips-Perron, kpss,
library(tseries)
adf.test(serie) # p-value < 0.05 indicates the TS is stationary
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  serie
```

```
## Dickey-Fuller = -17.206, Lag order = 17, p-value = 0.01
## alternative hypothesis: stationary
```

```
pp.test(serie)# p-value < 0.05 indicates the TS is stationary
```

```
##
##  Phillips-Perron Unit Root Test
##
## data:  serie
## Dickey-Fuller Z(alpha) = -4969.3, Truncation lag parameter = 10,
## p-value = 0.01
## alternative hypothesis: stationary
```

```
kpss.test(serie)# AO REVES: p-value < 0.05 indicates the TS isn't stationary
```

```
##
##  KPSS Test for Level Stationarity
##
## data:  serie
## KPSS Level = 0.31162, Truncation lag parameter = 10, p-value = 0.1
```

Vese que os tres están de acordo: confirma o que se vía na gráfica, é estacionaria.

Agora toca buscar o modelo.

**1b. Hai autorrelación? Como é esa autocorrelación?**

Probase o **test de Ljung-Box**, para comprobar a autocorrelación:

```
## Hai autocorrelacion?
#Ljung-Box
Box.test(serie,lag=10)
```

```
##
##  Box-Pierce test
##
## data:  serie
## X-squared = 59.061, df = 10, p-value = 5.455e-09
```

Sae ***p-valor<0.05***, si hai autocorrelación, pero, **como é a súa estructura?**

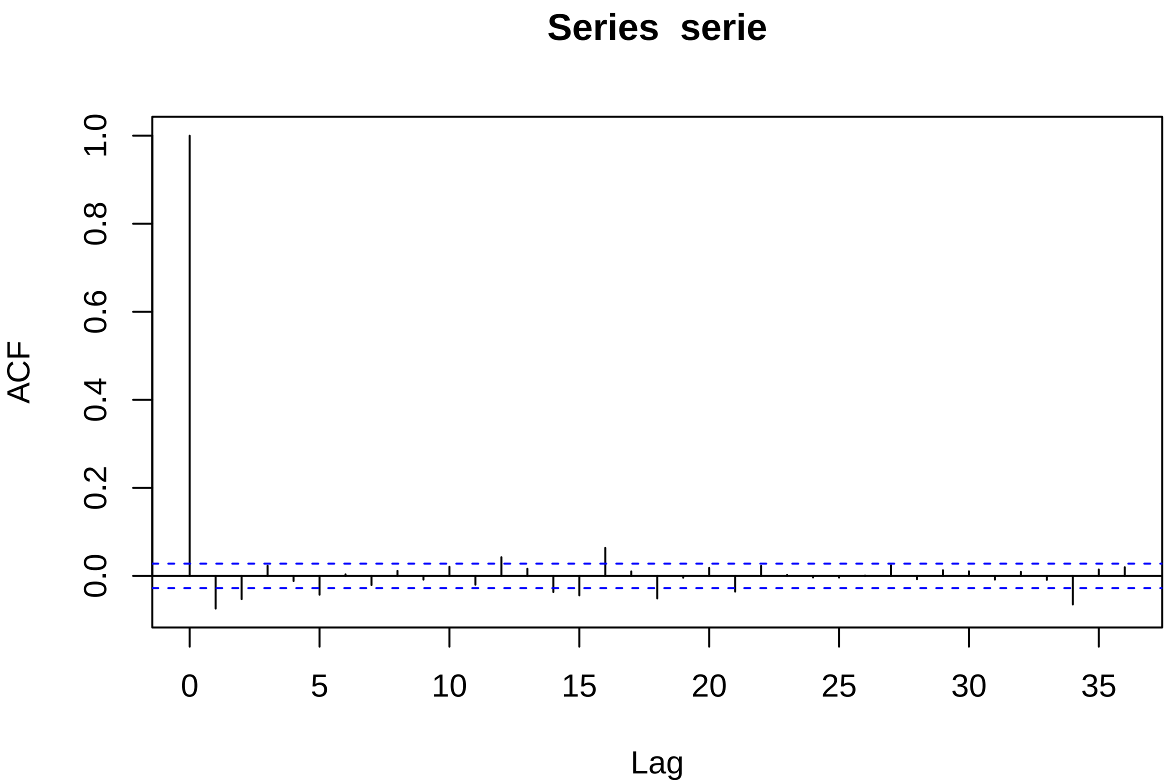
Funcións de autocorrelación e autocorrelación parcial:

Hai que observar o nº de retardos significativos nos dous correlogramas, e recordar a regra:

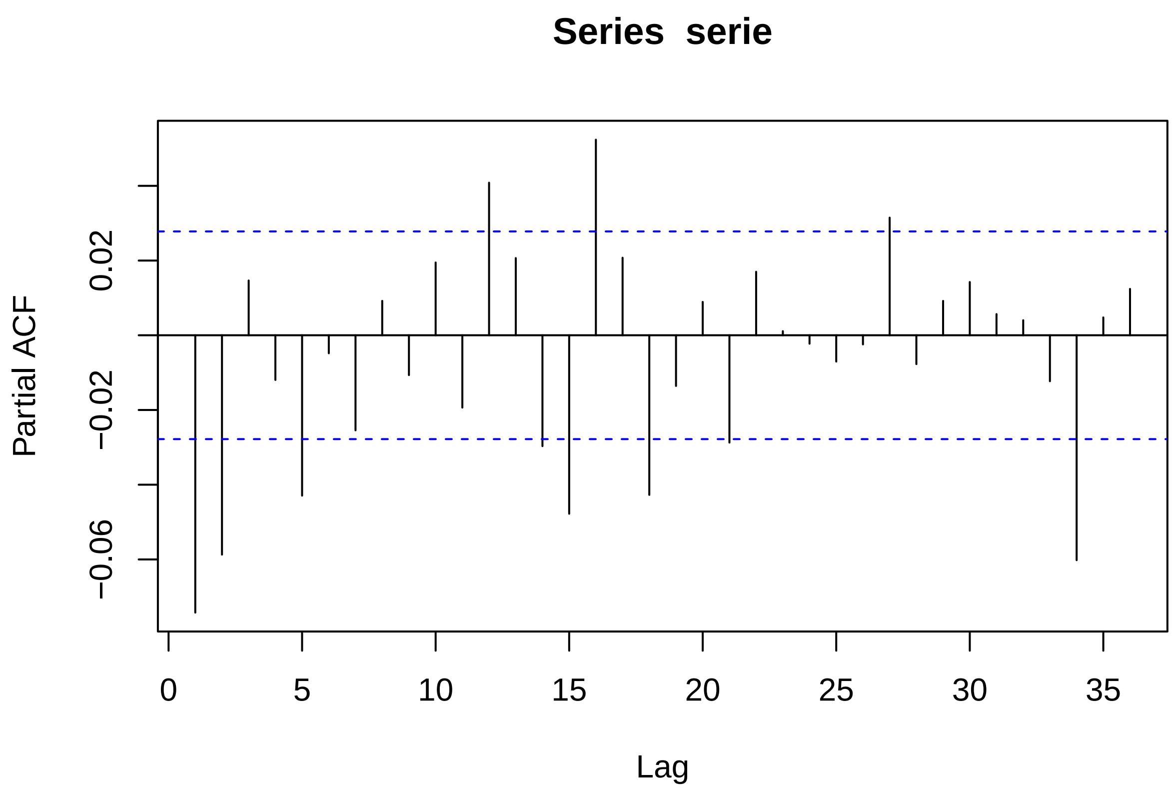
- AR(p):
  - ***acf***: espérase que sexan significativos, pero caendo o seu valor a medida que aumenta o retardo, ata que deixen de selo.
  - ***pacf***: espérase que haxa **p** retardos significativos.
- MA(q):
  - ***acf***: espérase que haxa **q** retardos significativos
  - ***pacf***: espérase que sexan significativos, pero caendo o seu valor a medida que aumenta o retardo, ata que deixen de selo.
- ARMA(p.q): máis complexos, combinan as dúas situacións anteriores

Vexamos para a nosa serie:

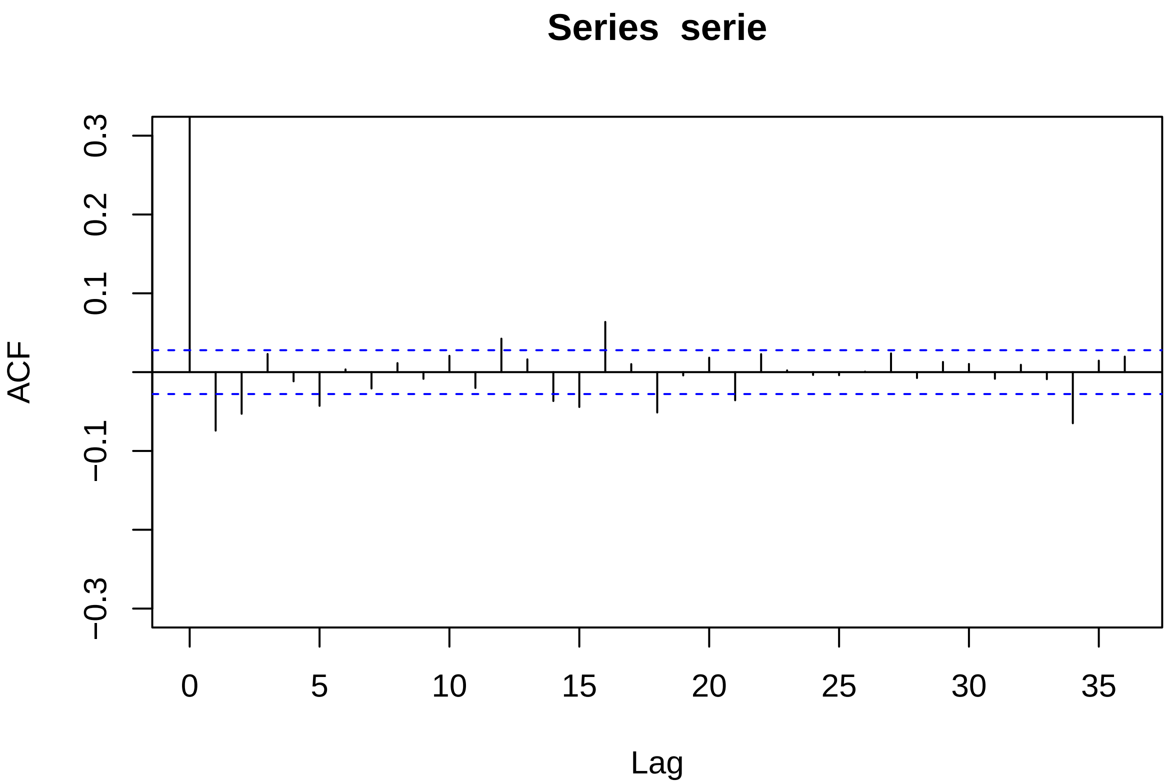
```
## Como e' esa autocorrelacion
### autocorrelacion: acf()
acf(serie)
```



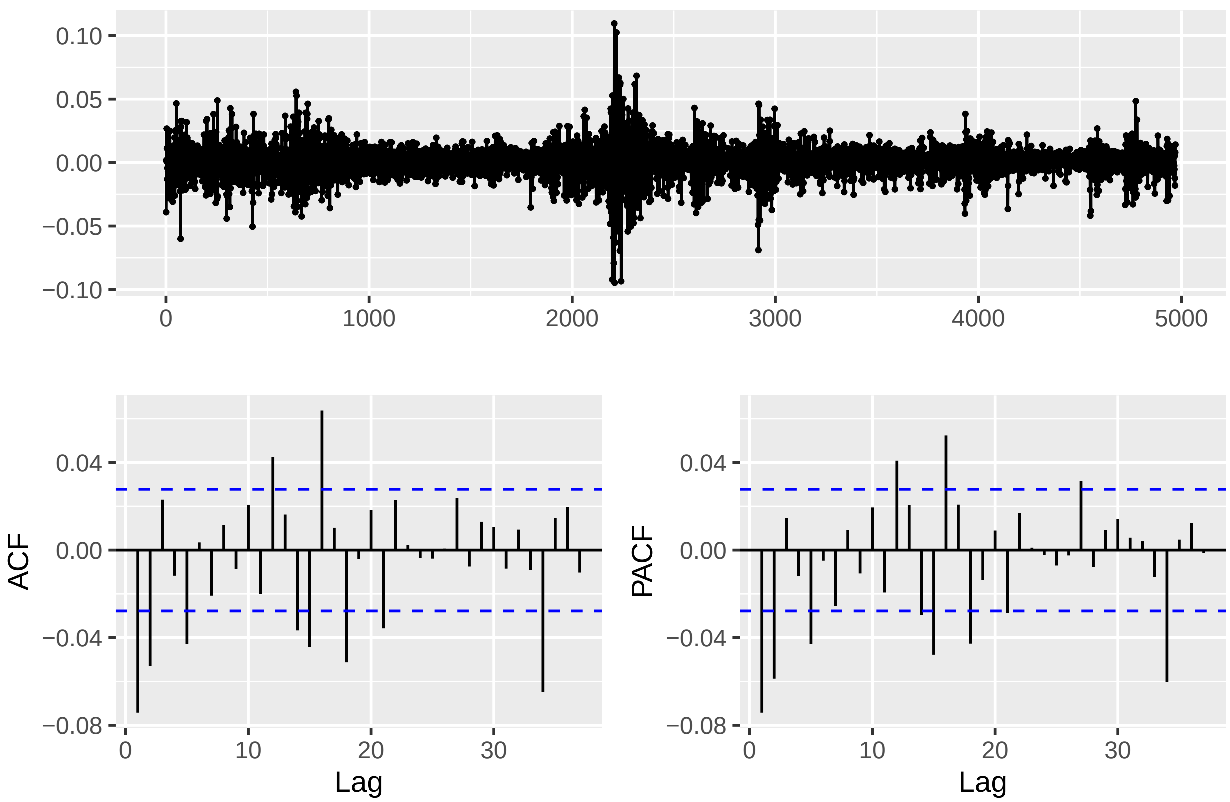

```
pacf(serie)
```




```
## Para que se vexa mellor o acf:
acf(serie,ylim=c(-0.3,0.3))
```




```
library(forecast)
ggtsdisplay(serie)
```




Vense **os dous primeiros retardos** significativos, tanto no **acf** coma no **pacf**, o que implica que non está nada clara a escolla. Pero podemos combinar modelos arredor de AR(2) e MA(2), e mirar cal deles ten os menores AIC e BIC.

Vou probar: **AR(2), AR(1), ARMA(1,1), ARMA(2,2), ARMA(2,1), ARMA(1,2), MA(2), MA(1)

Na practica do exercicio non é necesario que fagades todas estas probas, poderíades facer só: AR(2), MA(2), e ARMA(2,2), e escolleríades a mellor, así eu podería ver que tedes claro como facelo.

**1c. Estimar as diferentes propostas**

```
estimacion10=arima(serie,order=c(1,0,0))#AR(1)
#intervalos de confianza
confint(estimacion10)
```

```
##                   2.5 %        97.5 %
## ar1       -0.1021547169 -0.0466388609
## intercept -0.0001670853  0.0004524355
```

O coeficiente $\phi$ do AR(1) é significativo, posto que o cero non se inclúe no intervalo. En principio podemos aceptar este modelo.

```
estimacion20=arima(serie,order=c(2,0,0))#AR(2)
#intervalos de confianza
confint(estimacion20)
```

```
##                   2.5 %        97.5 %
## ar1       -0.1065753714 -0.0510005920
## ar2       -0.0866370404 -0.0310668943
## intercept -0.0001493308  0.0004350439
```

Os dous coeficientes $\phi$ do AR(2) son significativos. En principio podemos aceptar este modelo.

```
estimacion21=arima(serie,order=c(2,0,1))#ARMA(2,1)
#intervalos de confianza
confint(estimacion21)
```

```
##                   2.5 %        97.5 %
## ar1       -0.6291644499  0.1044610825
## ar2       -0.1100131484 -0.0364897289
## ma1       -0.1831814489  0.5513850972
## intercept -0.0001470703  0.0004422167
```

Non son significativos os coeficientes $\phi_1$ e $\theta_1$. Este modelo descártoo.

```
estimacion22=arima(serie,order=c(2,0,2))#ARMA(2,2)
#intervalos de confianza
confint(estimacion22)
```

```
##                  2.5 %       97.5 %
## ar1       -0.690866337 0.0031758379
## ar2       -0.597318139 0.1033091077
## ma1       -0.084860339 0.6203656208
## ma2       -0.184616300 0.5300572135
## intercept -0.000166035 0.0004356607
```

Non sae nada significativo. Descartado.

```
estimacion11=arima(serie,order=c(1,0,1))#ARMA(1,1)
#intervalos de confianza
confint(estimacion11)
```

```
##                   2.5 %        97.5 %
```

```
## ar1        0.2159436426  0.7554703770
## ma1       -0.8210506359 -0.3102050117
## intercept -0.0001369866  0.0004249381
```

Os dous coeficientes significativos.

```
estimacion12=arima(serie,order=c(1,0,2))#ARMA(1,2)
#intervalos de confianza
confint(estimacion12)
```

```
##                   2.5 %        97.5 %
## ar1       -0.6082738006  0.1917147926
## ma1       -0.2691829244  0.5294555869
## ma2       -0.1087868340 -0.0302055334
## intercept -0.0001442156  0.0004393433
```

O coeficiente da parte AR non sae significativo, polo que descarto este modelo.

```
estimacion02=arima(serie,order=c(0,0,2))#MA(2)
#intervalos de confianza
confint(estimacion02)
```

```
##                  2.5 %        97.5 %
## ma1       -0.105161817 -0.0494281193
## ma2       -0.080891508 -0.0239808037
## intercept -0.000146975  0.0004316944
```

Os dous coeficientes significativos.

```
estimacion01=arima(serie,order=c(0,0,1))#MA(1)
#intervalos de confianza
confint(estimacion01)
```

```
##                   2.5 %        97.5 %
## ma1       -0.1124637317 -0.0541038941
## intercept -0.0001626789  0.0004473715
```

O coeficiente é significativo.

### 1d. Estimar e seleccionar o mellor modelo: *AIC BIC*

```
AIC(estimacion10,estimacion20,estimacion11,estimacion01,estimacion02)
```

```
##              df       AIC
## estimacion10  3 -29917.47
## estimacion20  4 -29932.68
## estimacion11  4 -29929.18
## estimacion01  3 -29920.83
## estimacion02  4 -29931.80
```

```
BIC(estimacion10,estimacion20,estimacion11,estimacion01,estimacion02)
```

```
##              df       BIC
## estimacion10  3 -29897.94
## estimacion20  4 -29906.63
## estimacion11  4 -29903.14
## estimacion01  3 -29901.30
## estimacion02  4 -29905.76
```

Nos dous casos os resultados son moi cercanos, pero coinciden no menor valor da *estimacion 20*, ou sexa, quedaríamonos cun AR(2) para a nosa serie.

#### 1e. Comprobación

Se o modelo é aceptable entón os residuos non deben ter autocorrelación. Podemos volver a usar Ljung-Box, acf e pacf para comprobalo.

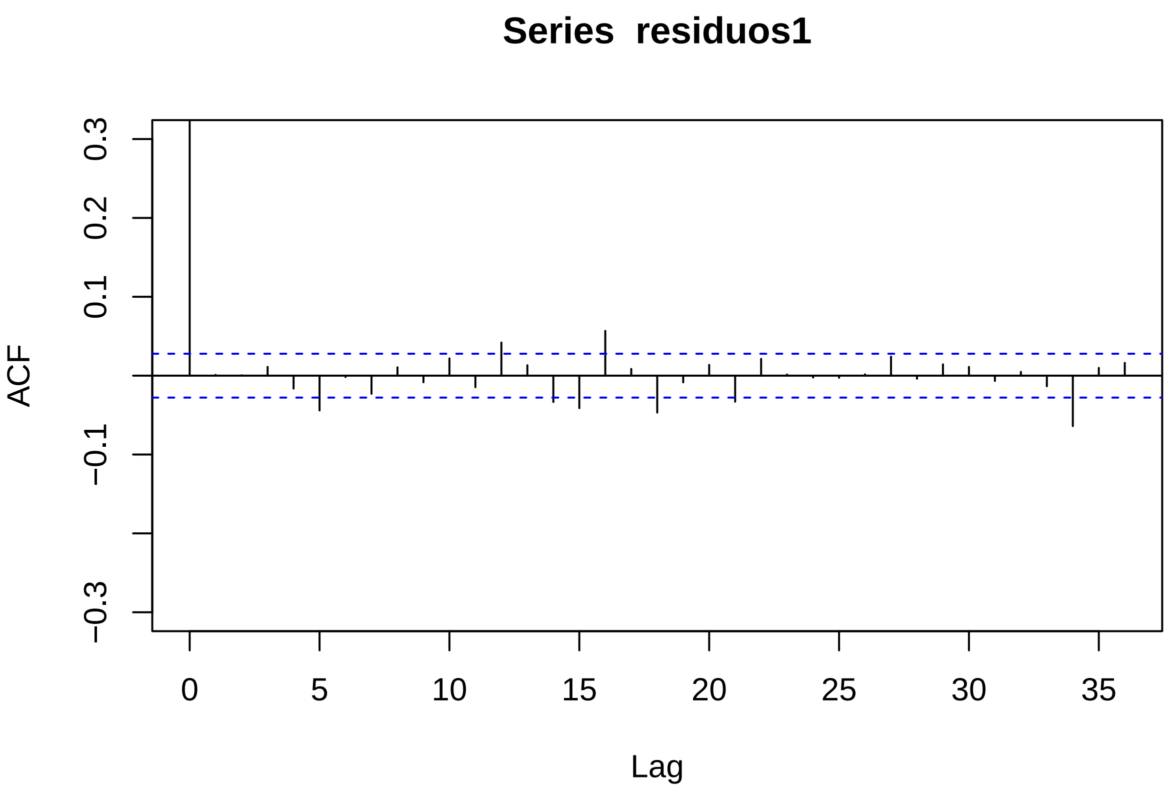
```
# 1º: obter os residuos
residuos1=estimacion20$residuals

Box.test(residuos1,lag=10)
```

```
##
##  Box-Pierce test
##
## data:  residuos1
## X-squared = 17.596, df = 10, p-value = 0.06218
```

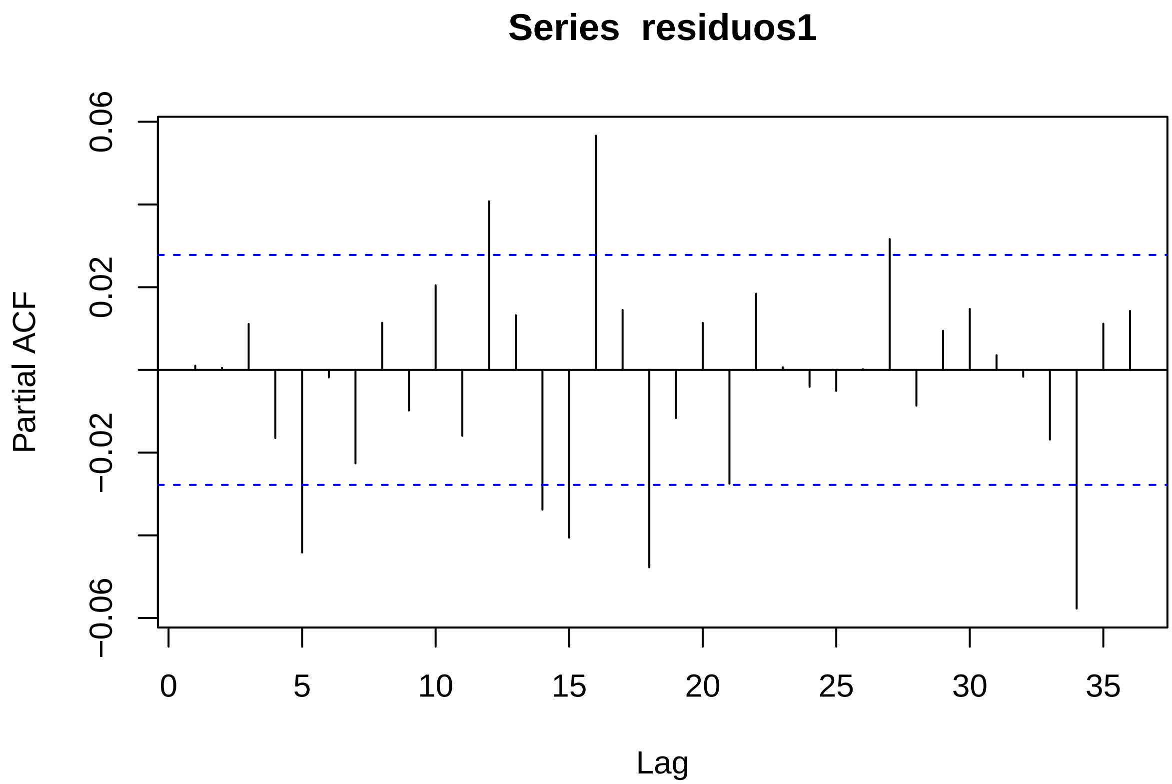
Non sae significativa a autocorrelación cun nivel de significación de 0.05; ou sexa rexeitámolo por pouco

Probamos cos correlogramas:

```
#correlogramas
acf(residuos1,ylim=c(-0.3,0.3))
```




```
pacf(residuos1)
```




Agora parece máis claro que se elimina a autocorrelación, polo menos dos primeiros retardos, posto que se consideraron os outros como efectos aleatorios.

Con esta parte rematada temos estimada a **ecuación da media** da nosa serie, queda a dúbida de se será suficiente ou será necesario estimar unha **ecuación da variabilidade**, debido a existencia de autocorrelación entre as varianzas.

Segue na seguinte parte, pero agora gardo os resultados para abrir no seguinte pdf:

```
save(serie,residuos1,estimacion20,file="resultados_apartado1a.Rdata")
```

## Exercicio de clase: series, dia 9 (13-11-2019) parte GARCH

- Enunciado: Descargar os datos da cotización do S&P500 entre o 01-01-2000 e o 07-10-2019, e analizade:
  - a.- Que modelo ARMA se pode axustar mellor a esa serie?
  - b.- Que modelo GARCH se pode axustar mellor a esa serie?

```
#cargar quantmod, para ver ben os datos
library(quantmod)
#Cargar os datos e os resultados gardados:
load("resultados_apartado1a.Rdata")
```

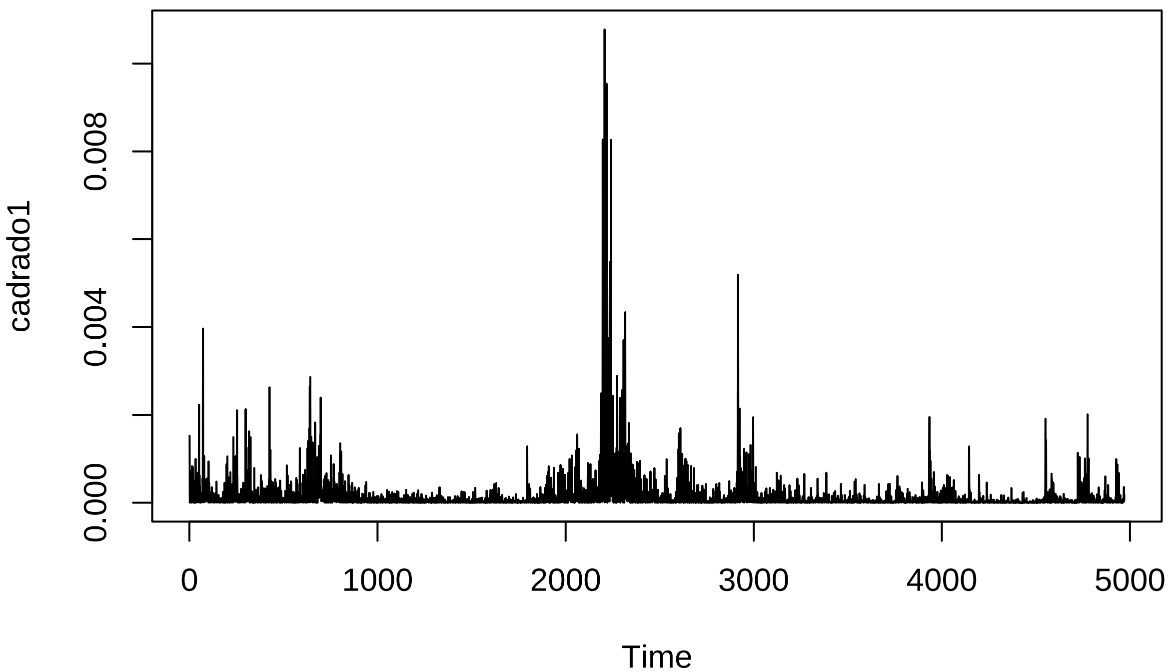
### RESPOSTA b: Que modelo GARCH se pode axustar mellor a esa serie?

Os modelos GARCH modelizan a autocorrelación da variabilidade, polo que se comeza a estudiar (se existe ou non) estudando **a autocorrelación do cadrado dos residuos**, do mesmo xeito que se fixo anteriormente para os valores da serie para o modelo ARMA.

#### 1a. Graficar o cadrado dos residuos:

```
#Primeiro calculalos
cadrado1=residuos1^2
```

```
#graficar os cadrados dos residuoss
plot(cadrado1)
```




Obsérvanse *clusteres de volatilidade*, o que pode implicar a utilidade dalgun tipo de GARCH para completar o modelo.

#### 2a. Hai autorrelación? Como é esa autocorrelación?

Probase o **test de Ljung-Box**, para comprobar a autocorrelación:

```
## Hai autocorrelacion?
#Ljung-Box
Box.test(cadrado1,lag=10)
```

```
##
```

```
##  Box-Pierce test
##
## data:  cadrado1
## X-squared = 4227.3, df = 10, p-value < 2.2e-16
```

Sae ***p-valor<0.05***, si hai autocorrelación, pero, **como é a súa estructura?**

O seguinte paso sería obter os correlogramas do cadrado dos residuos e a partir de aí deducir que modelo ARMA lles vería mellor. Deduciriamos un ARMA(p,q), para eses cadrados, o que implicaría un GARCH(q,p) para a ecuación da varianza.

Neste curso non imos seguir este esquema, se non que aplicaremos directamente un GARCH(1,1), xa que é un modelo que se aplica moi frecuentemente nas series financeiras, e despois comprobarase se é un modelo aceptable.

### 2b. Estimacion

Estimarase unha combinación de 2 ecuacións. A 1ª a ecuación da media (Un AR(2) neste caso) e a 2ª a ecuación da variabilidade (probamos un GARCH(1,1)). En total é unmodelo AR(2)+GARCH(1,1)

```
library(fGarch)
(modeloA <- garchFit(data=serie,formula=serie~arma(2,0)+garch(1,1),trace=F,include.mean=F))
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = serie ~ arma(2, 0) + garch(1, 1), data = serie,
##     include.mean = F, trace = F)
##
## Mean and Variance Equation:
##  data ~ arma(2, 0) + garch(1, 1)
## <environment: 0x560a077f0f28>
##  [data = serie]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##         ar1          ar2        omega       alpha1        beta1
## -5.3285e-02  -2.3949e-02   1.9825e-06   1.0748e-01   8.7688e-01
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##          Estimate  Std. Error  t value Pr(>|t|)
## ar1    -5.329e-02   1.531e-02   -3.481 0.000499 ***
## ar2    -2.395e-02   1.515e-02   -1.581 0.113843
## omega   1.983e-06   2.846e-07    6.965 3.28e-12 ***
## alpha1  1.075e-01   9.138e-03   11.761  < 2e-16 ***
## beta1   8.769e-01   9.695e-03   90.444  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
```

```
##  16111.01    normalized:  3.241653
##
## Description:
##  Tue Nov  1 20:00:32 2022 by user:
```

```
#trace=F implica que as saidas serán máis simples
#include.mean=F implica que non consideramos  no modelo o término independente da parte ARMA
# summary(modelo)
```

As ecuacións serían:

$X_t = 0.053X_{t-1} - 0.027Xt - 2 + \eta_t$

Sendo $\eta_t = \sigma_t \epsilon_t$, con $\sigma_t^2 = 0.012 + 0.161\eta_{t-1}^2 + 0.795\sigma_{t-1}^2$

Vemos tamén que o coeficiente $\phi_2$ non é significativo, polo que podemos probar tamén cun modelo máis simple: AR(1)+GARCH(1,1)

```
(modeloB <- garchFit(data=serie,formula=serie~arma(1,0)+garch(1,1),trace=F,include.mean=F))
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = serie ~ arma(1, 0) + garch(1, 1), data = serie,
##     include.mean = F, trace = F)
##
## Mean and Variance Equation:
##  data ~ arma(1, 0) + garch(1, 1)
## <environment: 0x560a01038e58>
##  [data = serie]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##         ar1        omega       alpha1        beta1
## -5.2361e-02   1.9883e-06   1.0798e-01   8.7638e-01
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##          Estimate  Std. Error  t value Pr(>|t|)
## ar1    -5.236e-02   1.527e-02   -3.430 0.000604 ***
## omega   1.988e-06   2.850e-07    6.976 3.03e-12 ***
## alpha1  1.080e-01   9.173e-03   11.771  < 2e-16 ***
## beta1   8.764e-01   9.717e-03   90.189  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  16109.76    normalized:  3.241401
##
## Description:
##  Tue Nov  1 20:00:32 2022 by user:
```

As ecuacións serían agora:

$X_t = 0.052 X_{t-1} + \eta_t$

Sendo $\eta_t = \sigma_t \epsilon_t$, con $\sigma_t^2 = 0.011 + 0.158\eta_{t-1}^2 + 0.799\sigma_{t-1}^2$

Vemos que son todos os coeficientes significativos.

### 1e. Comprobación

O paso final será comprobar os nosos modelos, ou sexa, comprobar se os residuos seguen a ter autocorrelación ou xa foi eliminada. Ademais obterase o AIC para comprobar cal sería o mellor dos dous modelos que estimamos.

```
# comprobación do modelo A
summary(modeloA)
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = serie ~ arma(2, 0) + garch(1, 1), data = serie,
##     include.mean = F, trace = F)
##
## Mean and Variance Equation:
##  data ~ arma(2, 0) + garch(1, 1)
## <environment: 0x560a077f0f28>
##  [data = serie]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##         ar1          ar2        omega       alpha1        beta1
## -5.3285e-02  -2.3949e-02   1.9825e-06   1.0748e-01   8.7688e-01
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##          Estimate  Std. Error  t value Pr(>|t|)
## ar1    -5.329e-02   1.531e-02   -3.481 0.000499 ***
## ar2    -2.395e-02   1.515e-02   -1.581 0.113843
## omega   1.983e-06   2.846e-07    6.965 3.28e-12 ***
## alpha1  1.075e-01   9.138e-03   11.761  < 2e-16 ***
## beta1   8.769e-01   9.695e-03   90.444  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  16111.01    normalized:  3.241653
##
## Description:
##  Tue Nov  1 20:00:32 2022 by user:
##
##
```

```
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  838.4591  0
##  Shapiro-Wilk Test  R    W      0.9808119 0
##  Ljung-Box Test     R    Q(10)  11.56018  0.3155696
##  Ljung-Box Test     R    Q(15)  22.28218  0.1006157
##  Ljung-Box Test     R    Q(20)  30.41121  0.06345892
##  Ljung-Box Test     R^2  Q(10)  16.92849  0.07596143
##  Ljung-Box Test     R^2  Q(15)  23.60234  0.07216155
##  Ljung-Box Test     R^2  Q(20)  26.36676  0.1540378
##  LM Arch Test       R    TR^2   17.48395  0.1322816
##
## Information Criterion Statistics:
##       AIC       BIC       SIC      HQIC
## -6.481293 -6.474743 -6.481295 -6.478997
```

Con este comando obtense entre outras cousas uns test para os residuos estandarizados, e para os seus cadrados: Ljung-Box para 10,15 e 20 retardos. Observase en todos os casos que non sae significavo. Polo tanto non ten autocorrelación nin nos residuos (ecuación da media, un AR(2)) nin nos seus cadrados (ecuación da variabilidade, un GARCH(1,1)).

Ademais proporcionanos un AIC=1.123824

Imos agora co modelo B:

```r
#comprobación:
summary(modeloB)
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = serie ~ arma(1, 0) + garch(1, 1), data = serie,
##     include.mean = F, trace = F)
##
## Mean and Variance Equation:
##  data ~ arma(1, 0) + garch(1, 1)
## <environment: 0x560a01038e58>
##  [data = serie]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##         ar1        omega       alpha1        beta1
## -5.2361e-02   1.9883e-06   1.0798e-01   8.7638e-01
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##          Estimate  Std. Error  t value Pr(>|t|)
## ar1    -5.236e-02   1.527e-02   -3.430 0.000604 ***
## omega   1.988e-06   2.850e-07    6.976 3.03e-12 ***
## alpha1  1.080e-01   9.173e-03   11.771  < 2e-16 ***
```

```
## beta1   8.764e-01   9.717e-03   90.189  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  16109.76    normalized:  3.241401
##
## Description:
##  Tue Nov  1 20:00:32 2022 by user:
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  816.8501  0
##  Shapiro-Wilk Test  R    W      0.9813135 0
##  Ljung-Box Test     R    Q(10)  13.77006  0.1837405
##  Ljung-Box Test     R    Q(15)  25.00783  0.04983839
##  Ljung-Box Test     R    Q(20)  33.65036  0.02858944
##  Ljung-Box Test     R^2  Q(10)  16.72982  0.08056021
##  Ljung-Box Test     R^2  Q(15)  23.26309  0.07871154
##  Ljung-Box Test     R^2  Q(20)  26.04914  0.1641949
##  LM Arch Test       R    TR^2   17.34764  0.1369815
##
## Information Criterion Statistics:
##       AIC       BIC       SIC      HQIC
## -6.481193 -6.475953 -6.481194 -6.479356
```

Séguese cumprindo que non hai ningún test de autocorrelación significativo, polo que tamén é un modelo valido.

O seu AIC= 1.123392 menor ca o valor do modelo A, polo tanto **aceptaríamos un AR(1)+GARCH(1,1)** como modelo da nosa serie.

### Modelo estimado:

$X_t = 0.052X_{t-1} + \eta_t$

Sendo $\eta_t = \sigma_t \epsilon_t$, con $\sigma_t^2 = 0.011 + 0.158\eta_{t-1}^2 + 0.799\sigma_{t-1}^2$