BMW vs Volkswagen

Sergio Casares

27/11/2020

Contents

1. Importamos librerías

Importamos las librerías necesarias para nuestro trabajo

```r
library(quantmod)
library(forecast)
library(fGarch)

library(forecast)
library(xts)
library(ggplot2)
library(ggfortify)
library(corrplot)
library(zoo)
library(TSA)

library(vars)
```

2. Introducción al Trabajo

Utilizando datos diarios de los dos series desde el año 2014, debemos analizar la relación entre BMW y Volkswagen, contestando a las siguiente preguntas:

- ¿Estimar el modelo garch de BMW?
- Plantear un modelo VAR mensual entre los dos activos.

```r
getSymbols('BMW.DE', from='2014-01-02', to='2020-11-24')
```

```
## [1] "BMW.DE"
```

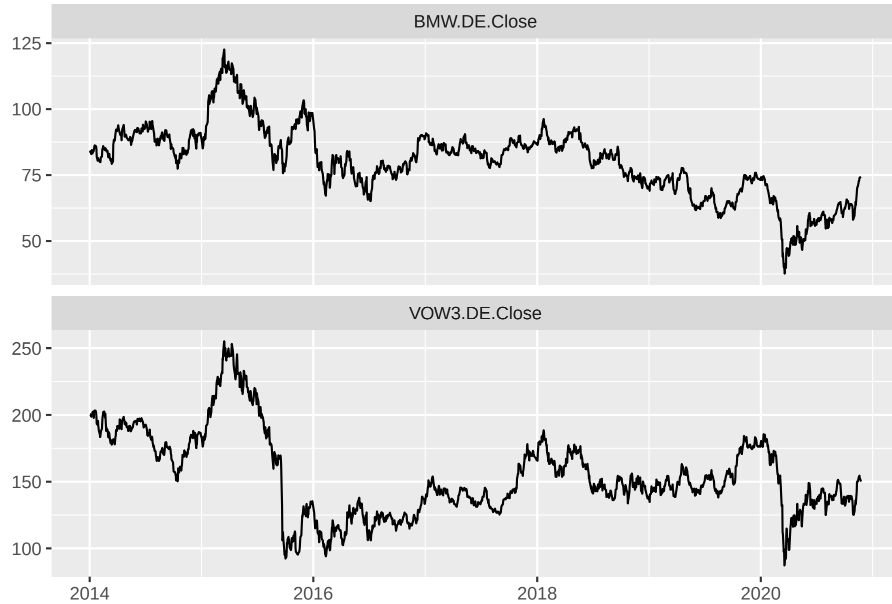
```r
getSymbols('VOW3.DE', from='2014-01-02', to='2020-11-24')
```

```
## [1] "VOW3.DE"
```

```r
BMW <- BMW.DE[,4]
VOW <- VOW3.DE[,4]

BMW <- na.omit(BMW)
VOW <- na.omit(VOW)

#View(BMW.DE)
#View(VOW)

data_union <- cbind(BMW, VOW)
```

Evolución del precio de cierre en bolsa de ambas empresas.

```r
autoplot(data_union)
```



3. Modelo Garch (para BMW)

El modelo GARCH es un modelo autorregresivo generalizado que captura las agrupaciones de volatilidad de las rentabilidades a través de la varianza condicional.

El modelo GARCH encuentra la volatilidad promedio a medio plazo mediante una autorregresión que depende de la suma de perturbaciones rezagadas y de la suma de varianzas rezagadas.

- Generalizado porque tiene en cuenta tanto las observaciones recientes como las históricas.
- Autorregresivo porque la variable dependiente se regresa en sí misma.
- Condicional porque la varianza futura depende de la varianza histórica.
- Heterocedástico porque la varianza varía en función de las observaciones.

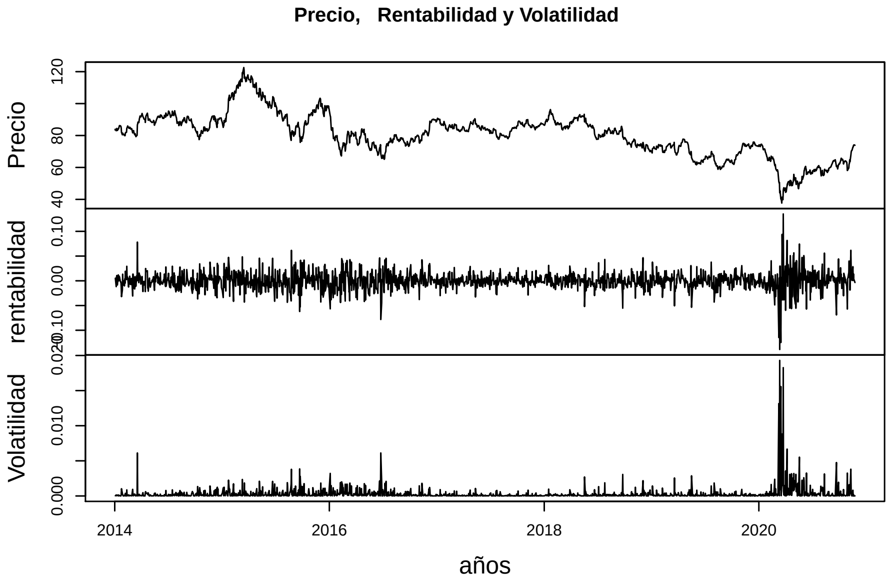
3.1. Creación de la función

```
#funciones
archTest <- function(rtn,m=10){
  # Perform Lagrange Multiplier Test for ARCH effect of a time series
  # rtn: time series
  # m: selected AR order
  # TSAY(2013)
  y=(rtn-mean(rtn))^2
  T=length(rtn)
  atsq=y[(m+1):T]
  x=matrix(0,(T-m),m)
  for (i in 1:m){
    x[,i]=y[(m+1-i):(T-i)]
  }
  md=lm(atsq~x)
  summary(md)
}
```

3.2. Cálculo Retorno Diario y Volatilidad (más T-Test)

Se realiza el cálculo del retorno diario (con la función dailyReturn) y la volatilidad diaria (la cual es el rendimiento diario al cuadrado).

```
#Calculate Daily Arithmetic Return
dRentCont=dailyReturn(BMW,type='log',leading=FALSE)
#Exclude NA (First data)
dRentCont=na.exclude(dRentCont)

#Volatilidad GARCH
#Plot return squared
plot.zoo(cbind(BMW,dRentCont,dRentCont^2),main=paste("Precio,   Rentabilidad y Volatilidad"),xlab="años
```



El T Test contrasta medias independientes:

Hipótesis nula (H0): considera que no hay diferencia o cambio. En el caso de comparar dos medias independientes la hipótesis nula considera que media1 = media2.

Hipótesis alternativa (HA): considera que el valor real de la media poblacional es mayor, menor o distinto del valor que establece la Ho. En el caso de comparar dos medias independientes la hipótesis alternativa considera que media1 != media2.

Si el p-value es menor que el valor de alpha seleccionado, existen evidencias suficientes para rechazar H0 en favor de HA

```
#testing mean
t.test(dRentCont)
```

```
##
##  One Sample t-test
##
## data:  dRentCont
## t = -0.16539, df = 1744, p-value = 0.8687
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
##  -0.0009045131  0.0007638276
## sample estimates:
##     mean of x
## -7.034274e-05
```

Al ser el p-value de 0.86, se acepta Ho y afirmamos que es un modelo NO GARCH.

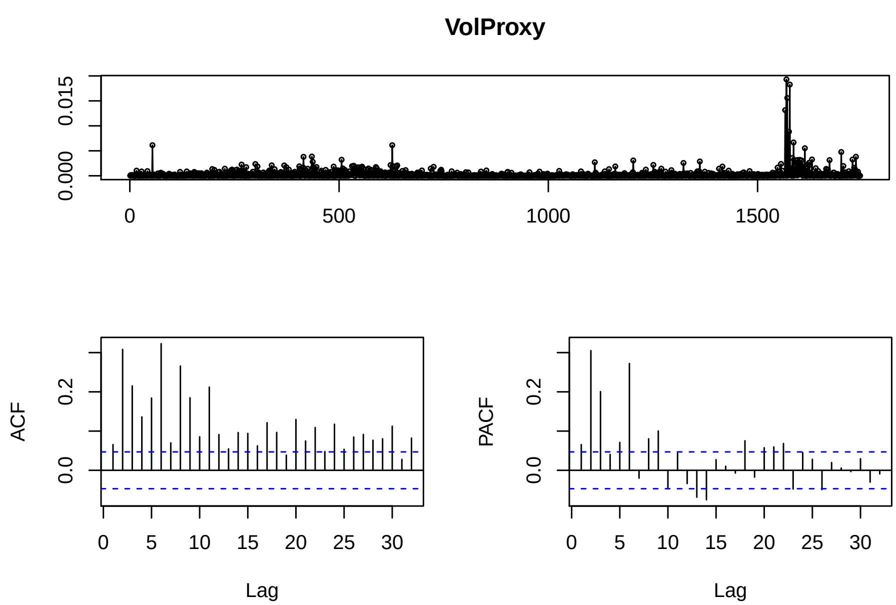
3.3. ACF Y PACF (más Ljung-Box Test y LM test)

ACF Y PACF

En la siguiente gráfica se puede observar como en la parte regular la correlación y el precio de hoy puede ser explicar por lo que ha ocurrido 30 días antes.

En la parte estacional, el valor disminuye hasta los 5-7 periodos anteriores.

```
#ACF & PACF
# VolProxy=abs(dRentCont) # absolute value
VolProxy=dRentCont^2 #squared


#ACF y PACF
tsdisplay(VolProxy)
```



La prueba de Ljung-Box se puede definir de la siguiente manera.

H0: Los datos se distribuyen de forma independiente (es decir, las correlaciones en la población de la que se toma la muestra son 0, de modo que cualquier correlación observada en los datos es el resultado de la aleatoriedad del proceso de muestreo). Ha: Los datos no se distribuyen de forma independiente.

Lagrange Multiplier test:

The Lagrange Multiplier test proposed by Engle (1982) fits a linear regression model for the squared residuals and examines whether the fitted model is significant.

So the null hypothesis is that the squared residuals are a sequence of white noise, namely, the residuals are homoscedastic.

Ljung-Box Test

```r
#Ljung-Box Test
Box.test(VolProxy,lag=10,  type="Lj")
```

```
##
##  Box-Ljung test
##
## data:  VolProxy
## X-squared = 734.86, df = 10, p-value < 2.2e-16
```

```r
Box.test(VolProxy,lag=20,  type="Lj")
```

```
##
##  Box-Ljung test
##
## data:  VolProxy
## X-squared = 947.41, df = 20, p-value < 2.2e-16
```

```r
Box.test(VolProxy,lag=40,  type="Lj")
```

```
##
##  Box-Ljung test
##
## data:  VolProxy
## X-squared = 1162.9, df = 40, p-value < 2.2e-16
```

En el test de Box Lung, rechazamos la hipótesis Nulo y afirmamos que los datos no se distribuyen de manera independiente.

Lagrange Multiplier test:

The Lagrange Multiplier test proposed by Engle (1982) fits a linear regression model for the squared residuals and examines whether the fitted model is significant.

So the null hypothesis is that the squared residuals are a sequence of white noise, namely, the residuals are homoscedastic.

```r
#LM test
archTest(dRentCont,20)
```

```
##
## Call:
## lm(formula = atsq ~ x)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0048062 -0.0002269 -0.0001090  0.0000536  0.0180109
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  9.196e-05  2.588e-05   3.553 0.000391 ***
```

```
## x1          -4.213e-02  2.418e-02  -1.742 0.081666 .
## x2           2.354e-01  2.421e-02   9.724  < 2e-16 ***
## x3           9.659e-02  2.483e-02   3.891 0.000104 ***
## x4          -9.398e-03  2.494e-02  -0.377 0.706308
## x5           8.396e-02  2.493e-02   3.367 0.000776 ***
## x6           2.493e-01  2.502e-02   9.966  < 2e-16 ***
## x7          -2.243e-02  2.565e-02  -0.875 0.381853
## x8           1.193e-01  2.558e-02   4.662 3.38e-06 ***
## x9           7.800e-02  2.573e-02   3.032 0.002465 **
## x10         -3.928e-02  2.575e-02  -1.526 0.127251
## x11          6.696e-02  2.586e-02   2.589 0.009710 **
## x12         -4.711e-02  2.585e-02  -1.822 0.068589 .
## x13         -8.039e-02  2.573e-02  -3.124 0.001814 **
## x14         -8.871e-02  2.581e-02  -3.437 0.000601 ***
## x15          1.634e-02  2.516e-02   0.650 0.515985
## x16         -5.929e-03  2.508e-02  -0.236 0.813153
## x17         -6.134e-03  2.512e-02  -0.244 0.807110
## x18          6.136e-02  2.500e-02   2.454 0.014216 *
## x19         -1.441e-02  2.441e-02  -0.590 0.555026
## x20          5.858e-02  2.440e-02   2.401 0.016435 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0008791 on 1704 degrees of freedom
## Multiple R-squared:  0.2368, Adjusted R-squared:  0.2279
## F-statistic: 26.44 on 20 and 1704 DF,  p-value: < 2.2e-16
```

En el test de Lagrange Multiplier se rechaza H0, entonces, podemos afirmar qu elos residuos son heterocedásticos.

3.4. Modelo GARCH básico

Vamos a analizar las pertubaciones y varianzas retardadas.

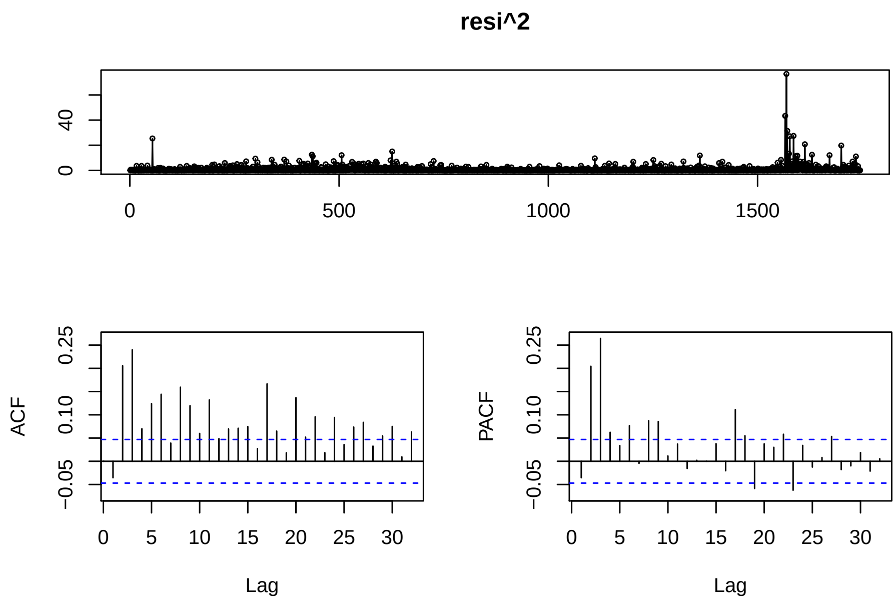
Creamos el modelo con p=1 y q=0

```r
#GARCH(1)
m1=garchFit(~1+garch(1,0),data=dRentCont,trace=F) # Fit an ARCH(1) model
summary(m1)
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~1 + garch(1, 0), data = dRentCont, trace = F)
##
## Mean and Variance Equation:
##  data ~ 1 + garch(1, 0)
## <environment: 0x00000000315b3a70>
##  [data = dRentCont]
##
## Conditional Distribution:
```

```
##  norm
##
## Coefficient(s):
##         mu       omega      alpha1
## 8.3412e-05  2.4058e-04  2.8689e-01
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu     8.341e-05   3.985e-04    0.209    0.834
## omega  2.406e-04   1.086e-05   22.155  < 2e-16 ***
## alpha1 2.869e-01   4.518e-02    6.350 2.15e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  4599.921    normalized:  2.636058
##
## Description:
##  Sat Nov 28 12:49:43 2020 by user: SergioC
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  3672.456  0
##  Shapiro-Wilk Test  R    W      0.9351779 0
##  Ljung-Box Test     R    Q(10)  22.10244  0.01459114
##  Ljung-Box Test     R    Q(15)  31.47007  0.007595859
##  Ljung-Box Test     R    Q(20)  37.40969  0.01044589
##  Ljung-Box Test     R^2  Q(10)  327.7234  0
##  Ljung-Box Test     R^2  Q(15)  389.8118  0
##  Ljung-Box Test     R^2  Q(20)  481.2582  0
##  LM Arch Test       R    TR^2   232.737   0
##
## Information Criterion Statistics:
##       AIC       BIC       SIC      HQIC
## -5.268678 -5.259283 -5.268684 -5.265205
```

```r
resi=residuals(m1,standardize=T) #residuals
resi=xts(resi,order.by=index(dRentCont)) #residuals as xts
tsdisplay(resi^2) #acf pacf residuals
```



De los análisis de contraste, obtenemos en todos que se debe rechazar H0, entonces podemos afirmar que NO nos encontramos ante un modelo Garch. Se puede observar también que las volatilidades no son ruido blanco.

3.5. Averiguar orden correcto del modelo GARCH

Como el modelo Garch (0,1) ha salido que no lo es un modelo Garch (según los estadisticos), debemos averiguar cual es el verdadero orden del modelo por medio de un autoarima y agregarlo a nuestro modelo Garch

```
(fit_BMW <- auto.arima(dRentCont, seasonal = TRUE))
```

```
## Series: dRentCont
## ARIMA(3,0,1) with zero mean
##
## Coefficients:
##           ar1     ar2     ar3     ma1
##       -0.7384  0.0802  0.1123  0.7709
## s.e.   0.0639  0.0296  0.0241  0.0603
##
## sigma^2 estimated as 0.0003114:  log likelihood=4570.84
## AIC=-9131.68   AICc=-9131.65   BIC=-9104.36
```

Arma(3,1) y garch(1,0)

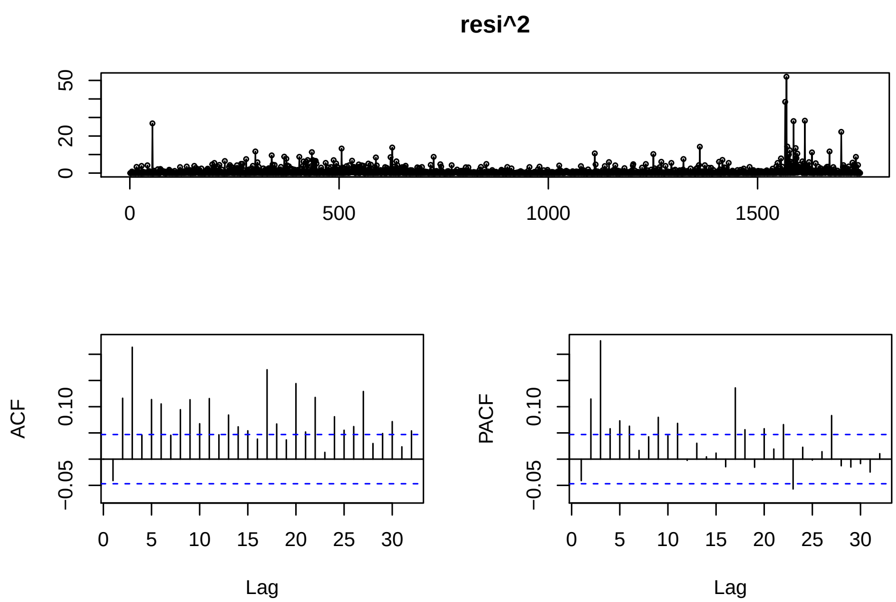
Con el autoarima obtenemos que nuestro modelo debe ser de orden (3,0,1), entonces añadiremos el modelo arma (3,0,1) a nuestro Garch (1,0).

```r
#ARCH(1)
m1=garchFit(~1+ arma(3,1) + garch(1,0),data=dRentCont,trace=F) # Fit an ARCH(1) model

#summary del modelo ARCH(1)
summary(m1) #observamos que no se cumple que sean ruido blanco todavia
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~1 + arma(3, 1) + garch(1, 0), data = dRentCont,
##     trace = F)
##
## Mean and Variance Equation:
##  data ~ 1 + arma(3, 1) + garch(1, 0)
## <environment: 0x000000002a47e220>
##  [data = dRentCont]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##          mu          ar1          ar2          ar3          ma1        omega
##  2.8735e-06   8.7944e-01   3.3352e-02  -1.3239e-01  -7.5431e-01   2.1556e-04
##      alpha1
##  3.8661e-01
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##          Estimate  Std. Error  t value Pr(>|t|)
## mu      2.873e-06   9.161e-05    0.031    0.975
## ar1     8.794e-01   1.063e-01    8.272 2.22e-16 ***
## ar2     3.335e-02   3.329e-02    1.002    0.316
## ar3    -1.324e-01   2.471e-02   -5.357 8.48e-08 ***
## ma1    -7.543e-01   1.043e-01   -7.229 4.85e-13 ***
## omega   2.156e-04   1.046e-05   20.616  < 2e-16 ***
## alpha1  3.866e-01   5.158e-02    7.496 6.59e-14 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  4625.684    normalized:  2.650822
##
## Description:
##  Sat Nov 28 12:49:44 2020 by user: SergioC
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  1571.603  0
```

```
##  Shapiro-Wilk Test  R    W      0.9526484 0
##  Ljung-Box Test     R    Q(10)  34.41166  0.0001572627
##  Ljung-Box Test     R    Q(15)  46.53583  4.368151e-05
##  Ljung-Box Test     R    Q(20)  53.71993  6.364473e-05
##  Ljung-Box Test     R^2  Q(10)  201.8306  0
##  Ljung-Box Test     R^2  Q(15)  253.4044  0
##  Ljung-Box Test     R^2  Q(20)  354.2905  0
##  LM Arch Test       R    TR^2   156.9019  0
##
## Information Criterion Statistics:
##       AIC       BIC       SIC      HQIC
## -5.293621 -5.271700 -5.293653 -5.285516
```

```r
resi=residuals(m1,standardize=T) #residuals
resi=xts(resi,order.by=index(dRentCont)) #residuals as xts
tsdisplay(resi^2) #acf pacf residuals
```



Analizando los estadisticos obtenemos que nuestro modelo no presenta las caracteríticas de un modelo Garch, por lo que debemos de bucar el orden correcto.

Arma(3,1) y garch(1,1)

Probamos ahora con el arma(3,1) y garch(1,1)

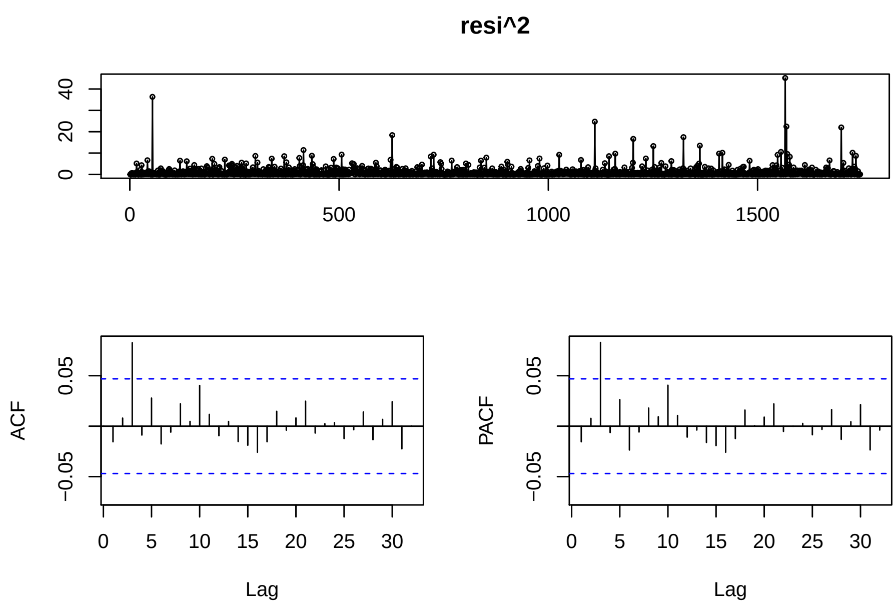
```r
#GARCH(1,1)
m2=garchFit(~1+arma(3,1) + garch(1,1),data=dRentCont,trace=F) # Fit an GARCH(1,1) model

summary(m2) #observamos aqui por los test de boxljung que para diferentes periodos el modelo tiene vari
```

```
##
```

```
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~1 + arma(3, 1) + garch(1, 1), data = dRentCont,
##     trace = F)
##
## Mean and Variance Equation:
##  data ~ 1 + arma(3, 1) + garch(1, 1)
## <environment: 0x0000000021f25110>
##  [data = dRentCont]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##          mu          ar1          ar2          ar3          ma1        omega
##  1.2904e-04  -2.6961e-01   5.1387e-02   5.3403e-02   3.4437e-01   3.0062e-06
##      alpha1        beta1
##  4.7160e-02   9.4334e-01
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##          Estimate  Std. Error  t value Pr(>|t|)
## mu      1.290e-04   4.602e-04    0.280   0.7791
## ar1    -2.696e-01   2.632e-01   -1.024   0.3057
## ar2     5.139e-02   3.296e-02    1.559   0.1190
## ar3     5.340e-02   2.569e-02    2.079   0.0376 *
## ma1     3.444e-01   2.625e-01    1.312   0.1896
## omega   3.006e-06   1.115e-06    2.697   0.0070 **
## alpha1  4.716e-02   7.796e-03    6.049 1.46e-09 ***
## beta1   9.433e-01   9.967e-03   94.648  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  4772.407    normalized:  2.734904
##
## Description:
##  Sat Nov 28 12:49:44 2020 by user: SergioC
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  930.3815  0
##  Shapiro-Wilk Test  R    W      0.9673055 0
##  Ljung-Box Test     R    Q(10)  6.793012  0.7448307
##  Ljung-Box Test     R    Q(15)  12.35557  0.65194
##  Ljung-Box Test     R    Q(20)  20.50383  0.4268355
##  Ljung-Box Test     R^2  Q(10)  18.34423  0.04942719
##  Ljung-Box Test     R^2  Q(15)  19.81295  0.1792261
##  Ljung-Box Test     R^2  Q(20)  21.93636  0.3439737
```

```
##  LM Arch Test       R    TR^2   18.51444  0.1009397
##
## Information Criterion Statistics:
##       AIC       BIC       SIC      HQIC
## -5.460639 -5.435587 -5.460681 -5.451377
```

```
#a su vez se ve en los test de shapiro que no se comporta como normal y que en jarque bera que tiene un

resi = residuals(m2,standardize=T) #residuals
resi = xts(resi,order.by=index(dRentCont)) #residuals as xts
tsdisplay(resi^2) #acf pacf residuals. Observamos lo que habiamos visto antes, es decir, que se convier
```




```
#plot(m2)
```

Observamos aqui por los test de box-Ljung que para diferentes periodos el modelo tiene varianza ruido blanco por el valor del p value. Esto mismo lo confirmamos en el grafico de los residuos al cuadrado.

Predicción del arma(3,1) y garch(1,1)

```
#Error in a_vec[(i - 1):(i - u2)] : solamente 0's pueden ser mezclados con subscritos negativos
#predict(m2) #forecast volatility

#predict(m2, n.ahead = 10, plot=TRUE, crit_val=2) #plot with 2*standard error
#predict(m2,n.ahead=20,plot=TRUE,conf=.9,nx=100) # plot 100 data with 90% confidence
```

4. Modelo VAR mensual entre los dos activos.

Los modelos VAR (o modelos autorregresivos vectoriales) se utilizan para modelizar series temporales en contextos multivariantes donde hay dependencias dinámicas entre distintas series.

Los modelos VAR se utilizan cuando las series temporales a modelizar son estacionarias

Los modelos VAR constituyen una extensión directa de los modelos autorregresivos univariantes cuando tenemos más de una serie temporal y se quiere captar las dependencias dinámicas que puede haber entre estas series.

4.1. Cálculo de rendimientos mensuales

Se calculan los rendimientos mensuales para su posterior uso en el modelo VAR.

```
rbmw=monthlyReturn(BMW)
rvow=monthlyReturn(VOW)

#generar vector
vY=cbind(rbmw,rvow)
colnames(vY)=c("BMW","VOW")
vY=na.omit(vY)
```

4.2. Selección modelo VAR y estimación

Con la función VARselect(vY)

What the $selection object is telling you is the total lag order selected by minimizing each of the 4 criteria (Akaike, Hannan-Quinn, Schwarz, and Final Prediction Error).

En este caso, nos dice que el orden del lag es de 1 para los cuatro.

What the criteria object tells you is the value of each criteria at the given lag. This may be useful if there are a lot of lags that have similar criterion values, allowing you to choose a more parsimonious specification if the minimizer has p very high, but a much lower value of p gives you a similar criterion.

```
#Seleccionar modelo
VARselect(vY)
```

```
## $selection
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      1      1      1      1
##
## $criteria
##                    1             2             3             4             5
## AIC(n) -1.025473e+01 -1.022222e+01 -1.013554e+01 -1.004775e+01 -9.961761e+00
## HQ(n)  -1.017971e+01 -1.009718e+01 -9.960483e+00 -9.822683e+00 -9.686675e+00
## SC(n)  -1.006648e+01 -9.908460e+00 -9.696272e+00 -9.482983e+00 -9.271486e+00
## FPE(n)  3.519377e-05  3.636905e-05  3.969219e-05  4.339213e-05  4.738825e-05
##                    6             7             8             9            10
## AIC(n) -1.000658e+01 -9.957692e+00 -9.890478e+00 -9.827826e+00 -9.840276e+00
## HQ(n)  -9.681475e+00 -9.582574e+00 -9.465344e+00 -9.352677e+00 -9.315111e+00
## SC(n)  -9.190797e+00 -9.016407e+00 -8.823688e+00 -8.635532e+00 -8.522478e+00
## FPE(n)  4.544986e-05  4.792746e-05  5.154432e-05  5.526819e-05  5.507255e-05
```

```r
#estimar
model.var=VAR(vY)
summary(model.var)
```

```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: BMW, VOW
## Deterministic variables: const
## Sample size: 82
## Log Likelihood: 200.588
## Roots of the characteristic polynomial:
## 0.3231 0.3231
## Call:
## VAR(y = vY)
##
##
## Estimation results for equation BMW:
## ====================================
## BMW = BMW.l1 + VOW.l1 + const
##
##         Estimate Std. Error t value Pr(>|t|)
## BMW.l1  0.319479   0.161069   1.983   0.0508 .
## VOW.l1 -0.415765   0.124644  -3.336   0.0013 **
## const   0.002551   0.008603   0.296   0.7676
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.07788 on 79 degrees of freedom
## Multiple R-Squared: 0.1269,  Adjusted R-squared: 0.1048
## F-statistic: 5.739 on 2 and 79 DF,  p-value: 0.004709
##
##
## Estimation results for equation VOW:
## ====================================
## VOW = BMW.l1 + VOW.l1 + const
##
##         Estimate Std. Error t value Pr(>|t|)
## BMW.l1  0.554471   0.203371   2.726  0.00788 **
## VOW.l1 -0.394845   0.157380  -2.509  0.01416 *
## const   0.003503   0.010862   0.322  0.74796
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.09834 on 79 degrees of freedom
## Multiple R-Squared: 0.09257, Adjusted R-squared: 0.0696
## F-statistic:  4.03 on 2 and 79 DF,  p-value: 0.02155
##
##
##
## Covariance matrix of residuals:
```

```
##          BMW      VOW
## BMW 0.006066 0.005563
## VOW 0.005563 0.009671
##
## Correlation matrix of residuals:
##        BMW    VOW
## BMW 1.0000 0.7264
## VOW 0.7264 1.0000
```

```r
model.var1=VAR(vY,type="none")
summary(model.var1)
```

```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: BMW, VOW
## Deterministic variables: none
## Sample size: 82
## Log Likelihood: 200.53
## Roots of the characteristic polynomial:
## 0.3227 0.3227
## Call:
## VAR(y = vY, type = "none")
##
##
## Estimation results for equation BMW:
## ====================================
## BMW = BMW.l1 + VOW.l1
##
##        Estimate Std. Error t value Pr(>|t|)
## BMW.l1   0.3185     0.1601   1.989  0.05012 .
## VOW.l1  -0.4153     0.1239  -3.351  0.00123 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.07744 on 80 degrees of freedom
## Multiple R-Squared: 0.1265,  Adjusted R-squared: 0.1047
## F-statistic: 5.795 on 2 and 80 DF,  p-value: 0.004462
##
##
## Estimation results for equation VOW:
## ====================================
## VOW = BMW.l1 + VOW.l1
##
##        Estimate Std. Error t value Pr(>|t|)
## BMW.l1   0.5531     0.2022   2.736  0.00767 **
## VOW.l1  -0.3942     0.1565  -2.519  0.01376 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.09779 on 80 degrees of freedom
## Multiple R-Squared: 0.09212, Adjusted R-squared: 0.06943
```

```
## F-statistic: 4.059 on 2 and 80 DF,  p-value: 0.02094
##
##
##
## Covariance matrix of residuals:
##          BMW      VOW
## BMW 0.005990 0.005494
## VOW 0.005494 0.009550
##
## Correlation matrix of residuals:
##        BMW    VOW
## BMW 1.0000 0.7264
## VOW 0.7264 1.0000
```

4.3. Causalidad de Granger y respuesta al impulso

Es un test consistente en comprobar si los resultados de una variable sirven para predecir a otra variable, si tiene carácter unidireccional o bidireccional. Para ello se tiene que comparar y deducir si el comportamiento actual y el pasado de una serie temporal A predice la conducta de una serie temporal B. Si ocurre el hecho, se dice que "el resultado A" causa en el sentido de Wiener-Granger "el resultado B"; el comportamiento es unidireccional. Si sucede lo explicado e igualmente "el resultado B" predice "el resultado A", el comportamiento es bidireccional, entonces "el resultado A" causa "el resultado B", y "el resultado B" causa "el resultado A".

Con unos valores cercanos a 0 del p-valor (se rechaza H0) y se puede afirmar que existe causalidad instantánea entre ambos activos como se puede observar en las siguientes ilustraciones.

```r
#causalidad de granger
causality(model.var1)
```

```
## $Granger
##
##  Granger causality H0: BMW do not Granger-cause VOW
##
## data:  VAR object model.var1
## F-Test = 7.4831, df1 = 1, df2 = 160, p-value = 0.006931
##
##
## $Instant
##
##  H0: No instantaneous causality between: BMW and VOW
##
## data:  VAR object model.var1
## Chi-squared = 28.337, df = 1, p-value = 1.019e-07
```

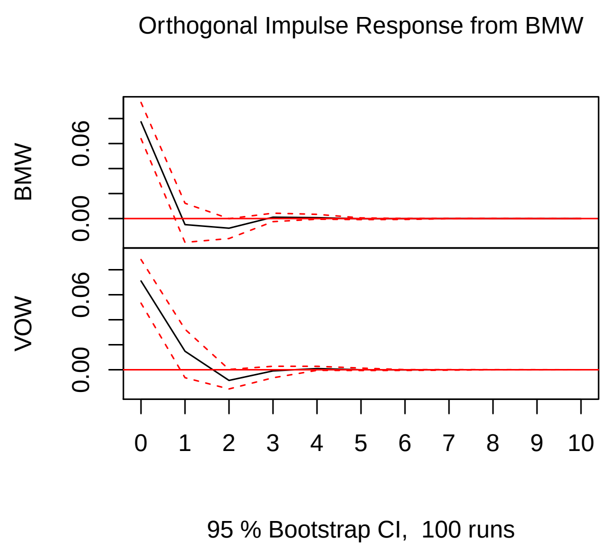
```r
#respuesta al impulso
model.ri=irf(model.var1)
model.ri
```

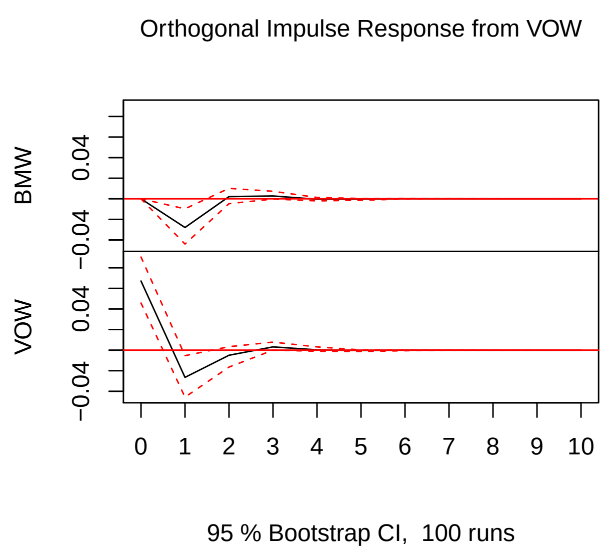
```
##
## Impulse response coefficients
## $BMW
##                 BMW           VOW
```

```
##  [1,]  7.743967e-02  7.106050e-02
##  [2,] -4.848076e-03  1.481977e-02
##  [3,] -7.698254e-03 -8.522954e-03
##  [4,]  1.087774e-03 -8.981852e-04
##  [5,]  7.194129e-04  9.556654e-04
##  [6,] -1.677592e-04  2.119053e-05
##  [7,] -6.222522e-05 -1.011365e-04
##  [8,]  2.218323e-05  5.450240e-06
##  [9,]  4.801201e-06  1.012068e-05
## [10,] -2.673882e-06 -1.333895e-06
## [11,] -2.975989e-07 -9.530737e-07
##
## $VOW
##                BMW           VOW
##  [1,]  0.000000e+00  6.717772e-02
##  [2,] -2.789740e-02 -2.647983e-02
##  [3,]  2.112161e-03 -4.991698e-03
##  [4,]  2.745587e-03  3.135794e-03
##  [5,] -4.278539e-04  2.824669e-04
##  [6,] -2.535583e-04 -3.479778e-04
##  [7,]  6.375816e-05 -3.072809e-06
##  [8,]  2.158075e-05  3.647440e-05
##  [9,] -8.274303e-06 -2.441509e-06
## [10,] -1.621165e-06 -3.613943e-06
## [11,]  9.845066e-07  5.279001e-07
##
##
## Lower Band, CI= 0.95
## $BMW
##                BMW           VOW
##  [1,]  6.395285e-02  5.331745e-02
##  [2,] -1.906564e-02 -6.341279e-03
##  [3,] -1.596995e-02 -1.534592e-02
##  [4,] -2.450606e-03 -6.419791e-03
##  [5,] -5.166666e-04 -4.662772e-04
##  [6,] -8.253522e-04 -5.873105e-04
##  [7,] -6.975769e-04 -5.166944e-04
##  [8,] -1.564131e-04 -3.025801e-04
##  [9,] -3.569406e-05 -2.302746e-05
## [10,] -2.227192e-05 -2.456223e-05
## [11,] -2.509009e-05 -1.634483e-05
##
## $VOW
##                BMW           VOW
##  [1,]  0.000000e+00  4.577498e-02
##  [2,] -4.396387e-02 -4.570481e-02
##  [3,] -4.721769e-03 -1.648259e-02
##  [4,] -2.520294e-04 -2.945493e-06
##  [5,] -2.056254e-03 -1.093441e-03
##  [6,] -1.384131e-03 -1.318983e-03
##  [7,] -2.792953e-04 -5.767651e-04
##  [8,] -3.326495e-05 -5.229544e-05
##  [9,] -7.579507e-05 -5.850684e-05
## [10,] -5.293010e-05 -3.302863e-05
```

```
## [11,] -1.737473e-05 -2.761711e-05
##
##
## Upper Band, CI= 0.95
## $BMW
##                BMW          VOW
##  [1,]  9.293198e-02 8.817605e-02
##  [2,]  1.226427e-02 3.241375e-02
##  [3,] -5.490478e-05 2.827516e-04
##  [4,]  4.307765e-03 2.810823e-03
##  [5,]  3.378678e-03 2.761730e-03
##  [6,]  5.526472e-04 1.357214e-03
##  [7,]  9.389575e-05 1.236433e-04
##  [8,]  1.438837e-04 1.185968e-04
##  [9,]  1.401456e-04 9.298918e-05
## [10,]  4.222165e-05 6.721338e-05
## [11,]  6.664771e-06 9.001507e-06
##
## $VOW
##                BMW           VOW
##  [1,]  0.000000e+00  9.047504e-02
##  [2,] -9.745164e-03 -5.386680e-03
##  [3,]  1.014112e-02  3.332282e-03
##  [4,]  7.281241e-03  7.794284e-03
##  [5,]  1.302553e-03  3.140361e-03
##  [6,]  2.849952e-04  2.016212e-04
##  [7,]  3.960997e-04  2.941956e-04
##  [8,]  2.794963e-04  2.022066e-04
##  [9,]  6.921353e-05  1.264110e-04
## [10,]  1.600876e-05  2.583313e-05
## [11,]  1.105585e-05  1.274669e-05
```

```r
plot(model.ri)
```





4.4. Se realiza la predición de los valores de BMW Y Volks

```
##prediccion
predict(model.var1, n.ahead = 8, ci = 0.95)
```

```
## $BMW
##                fcst      lower     upper        CI
## [1,] -4.424716e-04 -0.1522214 0.1513365 0.1517790
## [2,] -2.696233e-02 -0.1885693 0.1346446 0.1616070
```

```
## [3,]  2.087449e-03 -0.1602751 0.1644500 0.1623626
## [4,]  2.650072e-03 -0.1598156 0.1651158 0.1624657
## [5,] -4.180485e-04 -0.1628921 0.1620560 0.1624740
## [6,] -2.443527e-04 -0.1627194 0.1622307 0.1624751
## [7,]  6.203996e-05 -0.1624131 0.1625372 0.1624752
## [8,]  2.075209e-05 -0.1624544 0.1624960 0.1624752
##
## $VOW
##               fcst      lower     upper        CI
## [1,]  6.458674e-02 -0.1270737 0.2562472 0.1916604
## [2,] -2.570325e-02 -0.2263795 0.1749730 0.2006763
## [3,] -4.780645e-03 -0.2063885 0.1968272 0.2016079
## [4,]  3.038935e-03 -0.1986703 0.2047481 0.2017092
## [5,]  2.678192e-04 -0.2014508 0.2019865 0.2017187
## [6,] -3.367809e-04 -0.2020566 0.2013830 0.2017198
## [7,] -2.394996e-06 -0.2017223 0.2017175 0.2017199
## [8,]  3.525693e-05 -0.2016847 0.2017552 0.2017199
```

5. Bibliografía

Modelo GARCH:

- https://economipedia.com/definiciones/modelo-garch.html

Modelo VAR:

- https://guiasjuridicas.wolterskluwer.es/Content/Documento.aspx?params=H4sIAAAAAAAEAMtMSbF1jTAAASMTc
DxbIwMDS0NDQ3OQQGZapUt-ckhlQaptWmJOcSoAFtLjADUAAAA=WKE

T - Test:

- https://www.cienciadedatos.net/documentos/12_t-test

Causalidad de Granger:

- https://es.wikipedia.org/wiki/Causalidad_de_Granger

VarSelect:

- https://stackoverflow.com/questions/14131676/selecting-an-appropriate-lag-for-a-regression-equation-and-how-to-interpret-the