

Homework 4

Daniel de Souza Sobrinho Macedo RA:813524

Inicialização das bibliotecas

```
#install.packages("quantmod")
#install.packages("rugarch")

library(quantmod)
library(BatchGetSymbols)
library(tidyverse)
library(ggthemes)
library(rugarch)
library(FinTS)
```

Exercício 1

(a) Por que os parâmetros ϖ e α são restritos a serem positivos ou não negativos?

Os parâmetros ϖ e α determinam a variância condicional σ_t^2 do modelo. Como a variância precisa ser sempre positiva:

- $\varpi > 0$: garante que mesmo quando o termo $\epsilon_{t-1}^2 = 0$, a variância condicional σ_t^2 continua sendo positiva.
 - $\alpha \geq 0$: impede que a variância seja reduzida por valores negativos, o que poderia gerar uma variância negativa, algo não admissível.
-

(b) Como o modelo permite a formação de clusters de volatilidade?

O modelo ARCH(1) define a variância condicional σ_t^2 como dependente de ϵ_{t-1}^2 , o erro ao quadrado do período anterior. Assim:

- Quando ocorrem choques grandes (valores altos de ϵ_{t-1}^2), a variância condicional aumenta.
 - Isso gera períodos contínuos de alta ou baixa volatilidade, um comportamento conhecido como **clusters de volatilidade**, que é comumente observado em séries financeiras.
-

(c) Dois fatos estilizados que não são capturados pelo modelo ARCH(1):

1. **Assimetria (efeito leverage):** choques negativos costumam aumentar mais a volatilidade do que choques positivos de mesma magnitude. O modelo ARCH, por depender apenas de ϵ_t^2 , não diferencia entre choques positivos e negativos.
 2. **Persistência de longo prazo:** o ARCH(1) considera apenas um termo defasado. Séries reais frequentemente apresentam persistência por vários períodos, exigindo modelos mais complexos como GARCH ou FIGARCH para capturar esse comportamento.
-

(d) Diferença entre variância condicional e incondicional:

- **Variância condicional** é a variância de r_t dado o passado da série. No modelo ARCH(1), é dada por:

$$\text{Var}(r_t | \mathcal{I}_{t-1}) = \sigma_t^2 = \varpi + \alpha \epsilon_{t-1}^2$$

Ela varia com o tempo conforme os choques passados.

- **Variância incondicional** é a variância de r_t ao longo do tempo, sem considerar a informação passada. Se $\alpha < 1$, ela pode ser calculada como:

$$\text{Var}(r_t) = \frac{\varpi}{1 - \alpha}$$

Exercício 2

Utilização das FAC e FACP em modelos ARCH

- **FAC (Função de Autocorrelação):** mede a correlação linear entre os valores da série original em diferentes defasagens. Em séries financeiras, a FAC dos retornos costuma indicar **baixa autocorrelação**, mas isso não significa ausência de dependência na variância.
 - **FACP (Função de Autocorrelação dos Quadrados dos Resíduos):** avalia a autocorrelação dos **retornos ao quadrado**, ou seja, da variância. É especialmente útil porque séries ARCH apresentam **dependência na variância**, mesmo quando os retornos não são autocorrelacionados.
-

Etapas onde FAC e FACP são úteis

1. **Identificação de heterocedasticidade:**

- Antes de ajustar um modelo ARCH, verifica-se a presença de autocorrelação nos **retornos ao quadrado**.
- Se a FACP mostra autocorrelação significativa, isso indica **heterocedasticidade condicional**, justificando o uso de um modelo ARCH.

2. **Diagnóstico após ajuste do modelo:**

- Após estimar o modelo ARCH, calcula-se a FACP dos **resíduos padronizados ao quadrado**.
 - Se não houver autocorrelação significativa restante, isso indica que o modelo ajustado capturou bem a estrutura de volatilidade da série.
-

Resumo

FAC e FACP são ferramentas diagnósticas essenciais:

- A **FAC** ajuda a avaliar a estrutura dos retornos (usada para verificar estacionariedade).
 - A **FACP** ajuda a detectar e validar a presença de efeitos ARCH, sendo central para:
 - Confirmar a necessidade de modelos ARCH;
 - Avaliar a qualidade do ajuste do modelo.
-

Exercício 3:

Iniciamos com a obtenção dos dados de preços e retornos diários da Petrobras e do IBOVESPA desde o dia 01/01/2019:

```
ativos <- c('PETR4.SA', '^BVSP')

dados <- BatchGetSymbols(tickers = ativos,
                        first.date = '2019-01-01',
                        last.date = Sys.Date(),
                        type.return = "log",
                        freq.data = "daily")[[2]]

# Separando os dados dos ativos
serie_ibov <- dados %>% filter(ticker == '^BVSP') %>%
  pull(ret.closing.prices) %>%
  na.omit() %>%
  ts()

serie_petroleira <- dados %>% filter(ticker == 'PETR4.SA') %>%
  pull(ret.closing.prices) %>%
  na.omit() %>%
  ts()
```

Vamos começar com a Petrobras

```
ArchTest(serie_petroleira, lags = 10, demean = TRUE)
```

Começamos fazendo o teste para a heterocedasticidade condicional

```
##
## ARCH LM-test; Null hypothesis: no ARCH effects
##
## data: serie_petroleira
## Chi-squared = 294.49, df = 10, p-value < 2.2e-16
```

Podemos observar que existe uma heterocedasticidade condicional

Ajustando modelos de volatilidade para PETROBRAS

```
spec_garch_petro <- ugarchspec(  
  variance.model = list(model = "sGARCH", garchOrder = c(1, 1)),  
  mean.model = list(armaOrder = c(0, 0)),  
  distribution.model = "norm"  
)  
ajuste_garch_petro <- ugarchfit(spec = spec_garch_petro, data = serie_petroleira)  
ajuste_garch_petro
```

Modelo GARCH(1,1)

```
##  
## *-----*  
## *          GARCH Model Fit          *  
## *-----*  
##  
## Conditional Variance Dynamics  
## -----  
## GARCH Model   : sGARCH(1,1)  
## Mean Model    : ARFIMA(0,0,0)  
## Distribution   : norm  
##  
## Optimal Parameters  
## -----  
##      Estimate  Std. Error  t value Pr(>|t|)  
## mu      0.000547   0.000519   1.0538 0.291979  
## omega   0.000036   0.000011   3.1965 0.001391  
## alpha1  0.153201   0.025232   6.0716 0.000000  
## beta1   0.806579   0.034245  23.5533 0.000000  
##  
## Robust Standard Errors:  
##      Estimate  Std. Error  t value Pr(>|t|)  
## mu      0.000547   0.000603   0.90709 0.36436  
## omega   0.000036   0.000032   1.12597 0.26018  
## alpha1  0.153201   0.093196   1.64386 0.10020  
## beta1   0.806579   0.116393   6.92981 0.00000  
##  
## LogLikelihood : 3790.578  
##  
## Information Criteria  
## -----  
##  
## Akaike          -4.6690  
## Bayes           -4.6557  
## Shibata         -4.6690  
## Hannan-Quinn    -4.6641  
##  
## Weighted Ljung-Box Test on Standardized Residuals  
## -----  
##  
##                      statistic p-value  
## Lag[1]              0.2589 0.6109
```

```

## Lag[2*(p+q)+(p+q)-1][2]    0.4353  0.7247
## Lag[4*(p+q)+(p+q)-1][5]    1.5029  0.7392
## d.o.f=0
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## -----
##                statistic p-value
## Lag[1]                0.8118  0.3676
## Lag[2*(p+q)+(p+q)-1][5]    1.7523  0.6780
## Lag[4*(p+q)+(p+q)-1][9]    4.1681  0.5606
## d.o.f=2
##
## Weighted ARCH LM Tests
## -----
##                Statistic Shape Scale P-Value
## ARCH Lag[3]           0.4888 0.500 2.000  0.4844
## ARCH Lag[5]           1.0457 1.440 1.667  0.7194
## ARCH Lag[7]           1.5957 2.315 1.543  0.8020
##
## Nyblom stability test
## -----
## Joint Statistic:   1.2648
## Individual Statistics:
## mu      0.1097
## omega   0.6093
## alpha1  0.2211
## beta1   0.3811
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:           1.07 1.24 1.6
## Individual Statistic:      0.35 0.47 0.75
##
## Sign Bias Test
## -----
##                t-value    prob sig
## Sign Bias           0.5457 0.58536
## Negative Sign Bias  2.3359 0.01962 **
## Positive Sign Bias  0.5165 0.60558
## Joint Effect        10.9269 0.01213 **
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## -----
##    group statistic p-value(g-1)
## 1    20      113.0    2.255e-15
## 2    30      127.0    3.207e-14
## 3    40      142.5    1.028e-13
## 4    50      163.7    2.839e-14
##
##
## Elapsed time : 0.06198931

```

```
spec_egarch_petro <- ugarchspec(
  variance.model = list(model = "eGARCH", garchOrder = c(1, 1)),
  mean.model = list(armaOrder = c(0, 0)),
  distribution.model = "norm"
)
ajuste_egarch_petro <- ugarchfit(spec = spec_egarch_petro, data = serie_petroleira)
ajuste_egarch_petro
```

Modelo EGARCH(1,1)

```
##
## *-----*
## *          GARCH Model Fit          *
## *-----*
##
## Conditional Variance Dynamics
## -----
## GARCH Model   : eGARCH(1,1)
## Mean Model    : ARFIMA(0,0,0)
## Distribution   : norm
##
## Optimal Parameters
## -----
##      Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000139   0.000526   0.26416 0.791655
## omega   -0.422280   0.110609  -3.81776 0.000135
## alpha1  -0.051579   0.019053  -2.70720 0.006785
## beta1    0.940673   0.014863  63.28954 0.000000
## gamma1   0.277240   0.033355   8.31176 0.000000
##
## Robust Standard Errors:
##      Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000139   0.000517   0.26877 0.788108
## omega   -0.422280   0.222351  -1.89916 0.057543
## alpha1  -0.051579   0.045819  -1.12572 0.260284
## beta1    0.940673   0.030676  30.66475 0.000000
## gamma1   0.277240   0.096592   2.87022 0.004102
##
## LogLikelihood : 3800.366
##
## Information Criteria
## -----
##
## Akaike          -4.6799
## Bayes           -4.6632
## Shibata         -4.6799
## Hannan-Quinn    -4.6737
##
## Weighted Ljung-Box Test on Standardized Residuals
## -----
##                                statistic p-value
## Lag[1]                      0.3355 0.5624
```

```

## Lag[2*(p+q)+(p+q)-1][2]    0.4832  0.7015
## Lag[4*(p+q)+(p+q)-1][5]    1.1481  0.8253
## d.o.f=0
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## -----
##
##                statistic p-value
## Lag[1]                0.8346  0.3609
## Lag[2*(p+q)+(p+q)-1][5]    1.3587  0.7746
## Lag[4*(p+q)+(p+q)-1][9]    3.3128  0.7064
## d.o.f=2
##
## Weighted ARCH LM Tests
## -----
##                Statistic Shape Scale P-Value
## ARCH Lag[3]    0.2393 0.500 2.000 0.6247
## ARCH Lag[5]    0.5111 1.440 1.667 0.8803
## ARCH Lag[7]    1.1782 2.315 1.543 0.8829
##
## Nyblom stability test
## -----
## Joint Statistic: 1.1633
## Individual Statistics:
## mu    0.1220
## omega 0.3611
## alpha1 0.5415
## beta1 0.3744
## gamma1 0.2724
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:    1.28 1.47 1.88
## Individual Statistic: 0.35 0.47 0.75
##
## Sign Bias Test
## -----
##                t-value    prob sig
## Sign Bias        0.3940 0.69366
## Negative Sign Bias 2.4034 0.01635 **
## Positive Sign Bias 0.2535 0.79989
## Joint Effect      9.5397 0.02291 **
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## -----
##    group statistic p-value(g-1)
## 1    20    106.0    4.301e-14
## 2    30    124.2    9.445e-14
## 3    40    135.8    1.226e-12
## 4    50    140.5    9.055e-11
##
##
## Elapsed time : 0.07425404

```

```
spec_gjr_petro <- ugarchspec(
  variance.model = list(model = "gjrGARCH", garchOrder = c(1, 1)),
  mean.model = list(armaOrder = c(0, 0)),
  distribution.model = "norm"
)
ajuste_gjr_petro <- ugarchfit(spec = spec_gjr_petro, data = serie_petroleira)
ajuste_gjr_petro
```

Modelo GJR-GARCH(1,1)

```
##
## *-----*
## *          GARCH Model Fit          *
## *-----*
##
## Conditional Variance Dynamics
## -----
## GARCH Model   : gjrGARCH(1,1)
## Mean Model    : ARFIMA(0,0,0)
## Distribution   : norm
##
## Optimal Parameters
## -----
##      Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000338   0.000532   0.63495 0.525459
## omega    0.000044   0.000016   2.85119 0.004356
## alpha1   0.122784   0.027611   4.44689 0.000009
## beta1    0.787097   0.044458  17.70444 0.000000
## gamma1   0.068074   0.040428   1.68385 0.092210
##
## Robust Standard Errors:
##      Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000338   0.000513   0.65936 0.509661
## omega    0.000044   0.000059   0.75582 0.449756
## alpha1   0.122784   0.068391   1.79531 0.072603
## beta1    0.787097   0.185848   4.23516 0.000023
## gamma1   0.068074   0.144556   0.47092 0.637697
##
## LogLikelihood : 3792.387
##
## Information Criteria
## -----
##
## Akaike          -4.6700
## Bayes           -4.6534
## Shibata         -4.6700
## Hannan-Quinn -4.6639
##
## Weighted Ljung-Box Test on Standardized Residuals
## -----
##
##              statistic p-value
## Lag[1]          0.2543  0.6141
```



```

## Lag[2*(p+q)+(p+q)-1][2]    0.4164  0.7342
## Lag[4*(p+q)+(p+q)-1][5]    1.6834  0.6948
## d.o.f=0
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## -----
##               statistic p-value
## Lag[1]                0.4782  0.4893
## Lag[2*(p+q)+(p+q)-1][5]    1.6884  0.6936
## Lag[4*(p+q)+(p+q)-1][9]    3.5641  0.6634
## d.o.f=2
##
## Weighted ARCH LM Tests
## -----
##           Statistic Shape Scale P-Value
## ARCH Lag[3]    0.7435 0.500 2.000  0.3885
## ARCH Lag[5]    1.2299 1.440 1.667  0.6661
## ARCH Lag[7]    1.7307 2.315 1.543  0.7741
##
## Nyblom stability test
## -----
## Joint Statistic:  1.7164
## Individual Statistics:
## mu      0.1257
## omega   0.7255
## alpha1  0.2424
## beta1   0.4545
## gamma1  0.3075
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:      1.28 1.47 1.88
## Individual Statistic:  0.35 0.47 0.75
##
## Sign Bias Test
## -----
##               t-value   prob sig
## Sign Bias      0.6837 0.49428
## Negative Sign Bias 1.7860 0.07429  *
## Positive Sign Bias 0.2735 0.78452
## Joint Effect    7.3007 0.06291  *
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## -----
##   group statistic p-value(g-1)
## 1    20    110.7    6.047e-15
## 2    30    128.7    1.587e-14
## 3    40    141.7    1.404e-13
## 4    50    155.1    6.082e-13
##
##
## Elapsed time : 0.1567781

```

Agora vamos aplicar nos dados da IBOVESPA

```
ArchTest(serie_ibov, lags = 10, demean = TRUE)
```

Verificando heterocedasticidade condicional - IBOVESPA

```
##
## ARCH LM-test; Null hypothesis: no ARCH effects
##
## data: serie_ibov
## Chi-squared = 771.85, df = 10, p-value < 2.2e-16
```

Ajustando modelos de volatilidade para IBOVESPA

```
spec_garch_ibov <- ugarchspec(
  variance.model = list(model = "sGARCH", garchOrder = c(1, 1)),
  mean.model = list(armaOrder = c(0, 0)),
  distribution.model = "std"
)
ajuste_garch_ibov <- ugarchfit(spec = spec_garch_ibov, data = serie_ibov)
ajuste_garch_ibov
```

Modelo GARCH(1,1)

```
##
## *-----*
## *          GARCH Model Fit          *
## *-----*
##
## Conditional Variance Dynamics
## -----
## GARCH Model   : sGARCH(1,1)
## Mean Model    : ARFIMA(0,0,0)
## Distribution   : std
##
## Optimal Parameters
## -----
##      Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000543   0.000271   2.0056 0.044897
## omega    0.000004   0.000002   1.6811 0.092736
## alpha1   0.073597   0.013552   5.4305 0.000000
## beta1    0.903367   0.016341  55.2835 0.000000
## shape    9.313780   1.832134   5.0836 0.000000
##
## Robust Standard Errors:
##      Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000543   0.000276   1.96498 0.049416
## omega    0.000004   0.000005   0.71106 0.477049
```

```

## alpha1  0.073597    0.024271  3.03225 0.002427
## beta1   0.903367    0.031386 28.78285 0.000000
## shape   9.313780    2.056330  4.52932 0.000006
##
## LogLikelihood : 4868.948
##
## Information Criteria
## -----
##
## Akaike          -5.9975
## Bayes           -5.9808
## Shibata         -5.9975
## Hannan-Quinn    -5.9913
##
## Weighted Ljung-Box Test on Standardized Residuals
## -----
##
##                statistic p-value
## Lag[1]                2.144  0.1431
## Lag[2*(p+q)+(p+q)-1][2] 2.229  0.2274
## Lag[4*(p+q)+(p+q)-1][5] 2.789  0.4470
## d.o.f=0
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## -----
##
##                statistic p-value
## Lag[1]                0.0002065 0.98853
## Lag[2*(p+q)+(p+q)-1][5] 3.9063770 0.26598
## Lag[4*(p+q)+(p+q)-1][9] 9.8121406 0.05501
## d.o.f=2
##
## Weighted ARCH LM Tests
## -----
##
##                Statistic Shape Scale P-Value
## ARCH Lag[3]        5.609 0.500 2.000 0.01787
## ARCH Lag[5]        5.672 1.440 1.667 0.07172
## ARCH Lag[7]        6.209 2.315 1.543 0.12778
##
## Nyblom stability test
## -----
## Joint Statistic:  5.8117
## Individual Statistics:
## mu      0.2826
## omega   0.3127
## alpha1  0.4999
## beta1   0.3077
## shape   0.1330
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:      1.28 1.47 1.88
## Individual Statistic:  0.35 0.47 0.75
##
## Sign Bias Test
## -----

```

```
##              t-value    prob sig
## Sign Bias      1.1737 0.24070
## Negative Sign Bias 1.8308 0.06732 *
## Positive Sign Bias 0.8256 0.40918
## Joint Effect    8.1367 0.04327 **
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## -----
##   group statistic p-value(g-1)
## 1     20      25.13     0.15640
## 2     30      37.03     0.14556
## 3     40      45.18     0.22961
## 4     50      68.26     0.03576
##
##
## Elapsed time : 0.1003635
```

```
spec_egarch_ibov <- ugarchspec(
  variance.model = list(model = "eGARCH", garchOrder = c(1, 1)),
  mean.model = list(armaOrder = c(0, 0)),
  distribution.model = "std"
)
ajuste_egarch_ibov <- ugarchfit(spec = spec_egarch_ibov, data = serie_ibov)
ajuste_egarch_ibov
```

Modelo EGARCH(1,1)

```
##
## *-----*
## *          GARCH Model Fit          *
## *-----*
##
## Conditional Variance Dynamics
## -----
## GARCH Model   : eGARCH(1,1)
## Mean Model    : ARFIMA(0,0,0)
## Distribution   : std
##
## Optimal Parameters
## -----
##      Estimate Std. Error t value Pr(>|t|)
## mu      0.000342  0.000191  1.7884 0.073707
## omega   -0.187665  0.010854 -17.2896 0.000000
## alpha1  -0.080994  0.014554  -5.5651 0.000000
## beta1    0.978969  0.001340 730.8304 0.000000
## gamma1   0.133268  0.025166   5.2956 0.000000
## shape   10.332454  1.908623   5.4136 0.000000
##
## Robust Standard Errors:
##      Estimate Std. Error t value Pr(>|t|)
```

```

## mu      0.000342    0.000214    1.5976 0.110137
## omega  -0.187665    0.030603   -6.1323 0.000000
## alpha1 -0.080994    0.024733   -3.2747 0.001058
## beta1   0.978969    0.003763  260.1712 0.000000
## gamma1  0.133268    0.058751    2.2684 0.023307
## shape  10.332454    2.792906    3.6995 0.000216
##
## LogLikelihood : 4879.272
##
## Information Criteria
## -----
##
## Akaike      -6.0090
## Bayes       -5.9890
## Shibata     -6.0090
## Hannan-Quinn -6.0016
##
## Weighted Ljung-Box Test on Standardized Residuals
## -----
##
##                statistic p-value
## Lag[1]                2.126  0.1449
## Lag[2*(p+q)+(p+q)-1] [2]    2.189  0.2333
## Lag[4*(p+q)+(p+q)-1] [5]    2.663  0.4721
## d.o.f=0
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## -----
##
##                statistic p-value
## Lag[1]                0.07516  0.7840
## Lag[2*(p+q)+(p+q)-1] [5]    3.61773  0.3057
## Lag[4*(p+q)+(p+q)-1] [9]    6.77345  0.2189
## d.o.f=2
##
## Weighted ARCH LM Tests
## -----
##                Statistic Shape Scale P-Value
## ARCH Lag[3]          3.831 0.500 2.000 0.05031
## ARCH Lag[5]          4.152 1.440 1.667 0.16066
## ARCH Lag[7]          5.143 2.315 1.543 0.21048
##
## Nyblom stability test
## -----
## Joint Statistic:  2.2267
## Individual Statistics:
## mu      0.6837
## omega   0.4863
## alpha1  0.4019
## beta1   0.5175
## gamma1  0.4678
## shape   0.1027
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:      1.49 1.68 2.12

```

```
## Individual Statistic:      0.35 0.47 0.75
##
## Sign Bias Test
## -----
##              t-value   prob sig
## Sign Bias      0.9596 0.3374
## Negative Sign Bias 1.6416 0.1009
## Positive Sign Bias 1.4603 0.1444
## Joint Effect      6.2160 0.1016
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## -----
##   group statistic p-value(g-1)
## 1     20      26.98    0.10520
## 2     30      38.17    0.11855
## 3     40      55.19    0.04452
## 4     50      48.96    0.47466
##
##
## Elapsed time : 0.1643481
```

```
spec_gjr_ibov <- ugarchspec(
  variance.model = list(model = "gjrGARCH", garchOrder = c(1, 1)),
  mean.model = list(armaOrder = c(0, 0)),
  distribution.model = "std"
)
ajuste_gjr_ibov <- ugarchfit(spec = spec_gjr_ibov, data = serie_ibov)
ajuste_gjr_ibov
```

Modelo GJR-GARCH(1,1)

```
##
## *-----*
## *          GARCH Model Fit          *
## *-----*
##
## Conditional Variance Dynamics
## -----
## GARCH Model   : gjrGARCH(1,1)
## Mean Model    : ARFIMA(0,0,0)
## Distribution   : std
##
## Optimal Parameters
## -----
##      Estimate Std. Error t value Pr(>|t|)
## mu      0.000399  0.000268  1.4908 0.136015
## omega    0.000004  0.000001  3.6500 0.000262
## alpha1   0.006844  0.000650 10.5317 0.000000
## beta1    0.918444  0.009632 95.3539 0.000000
## gamma1   0.093848  0.018351  5.1141 0.000000
```

```

## shape    9.843545    2.019076    4.8753 0.000001
##
## Robust Standard Errors:
##      Estimate Std. Error t value Pr(>|t|)
## mu      0.000399    0.000298  1.33884 0.180624
## omega   0.000004    0.000002  1.97225 0.048581
## alpha1  0.006844    0.022575  0.30319 0.761745
## beta1   0.918444    0.009369 98.02965 0.000000
## gamma1  0.093848    0.032601  2.87864 0.003994
## shape   9.843545    2.184126  4.50686 0.000007
##
## LogLikelihood : 4878.257
##
## Information Criteria
## -----
##
## Akaike          -6.0077
## Bayes           -5.9878
## Shibata         -6.0077
## Hannan-Quinn   -6.0003
##
## Weighted Ljung-Box Test on Standardized Residuals
## -----
##
##              statistic p-value
## Lag[1]              1.681  0.1948
## Lag[2*(p+q)+(p+q)-1] [2]    1.726  0.3127
## Lag[4*(p+q)+(p+q)-1] [5]    2.132  0.5875
## d.o.f=0
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## -----
##
##              statistic p-value
## Lag[1]              0.4058  0.5241
## Lag[2*(p+q)+(p+q)-1] [5]    1.5830  0.7195
## Lag[4*(p+q)+(p+q)-1] [9]    4.4826  0.5091
## d.o.f=2
##
## Weighted ARCH LM Tests
## -----
##
##      Statistic Shape Scale P-Value
## ARCH Lag[3]      1.611 0.500 2.000  0.2044
## ARCH Lag[5]      1.737 1.440 1.667  0.5324
## ARCH Lag[7]      2.347 2.315 1.543  0.6441
##
## Nyblom stability test
## -----
## Joint Statistic:  1.3531
## Individual Statistics:
## mu      0.49225
## omega   0.28626
## alpha1  0.41136
## beta1   0.34545
## gamma1  0.42888

```

```
## shape 0.09722
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:      1.49 1.68 2.12
## Individual Statistic: 0.35 0.47 0.75
##
## Sign Bias Test
## -----
##          t-value  prob sig
## Sign Bias      1.254 0.2101
## Negative Sign Bias 0.773 0.4397
## Positive Sign Bias 1.026 0.3051
## Joint Effect      3.593 0.3090
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## -----
##   group statistic p-value(g-1)
## 1    20      27.64    0.09056
## 2    30      39.39    0.09436
## 3    40      49.52    0.12060
## 4    50      55.87    0.23260
##
##
## Elapsed time : 0.235781
```

Exercício 4

Criando uma função para calcular persistência e tempo de meia-vida (half-life)

```
calcular_persistencia <- function(modelo_garch, modelo_egarch, modelo_gjr) {

  half_life <- function(persistencia) {
    log(0.5) / log(persistencia)
  }

  persistencias <- c(
    persistence(modelo_garch),
    persistence(modelo_egarch),
    persistence(modelo_gjr)
  )

  half_lives <- sapply(persistencias, half_life)

  tibble(
    Modelo = c("GARCH", "EGARCH", "GJR-GARCH"),
    Persistência = round(persistencias, 4),
    Half_Life = round(half_lives, 2)
  )
}
```


Aplicando a função às duas séries, Petrobras e IBOVESPA:

```
# Petrobras
resultado_petrobras <- calcular_persistencia(
  ajuste_garch_petro,
  ajuste_egarch_petro,
  ajuste_gjr_petro
)
print(resultado_petrobras)
```

```
## # A tibble: 3 x 3
##   Modelo      Persistência Half_Life
##   <chr>          <dbl>      <dbl>
## 1 GARCH          0.960        16.9
## 2 EGARCH         0.941        11.3
## 3 GJR-GARCH      0.944        12.0
```

```
# IBOVESPA
resultado_ibov <- calcular_persistencia(
  ajuste_garch_ibov,
  ajuste_egarch_ibov,
  ajuste_gjr_ibov
)
print(resultado_ibov)
```

```
## # A tibble: 3 x 3
##   Modelo      Persistência Half_Life
##   <chr>          <dbl>      <dbl>
## 1 GARCH          0.977        29.7
## 2 EGARCH         0.979        32.6
## 3 GJR-GARCH      0.972        24.6
```