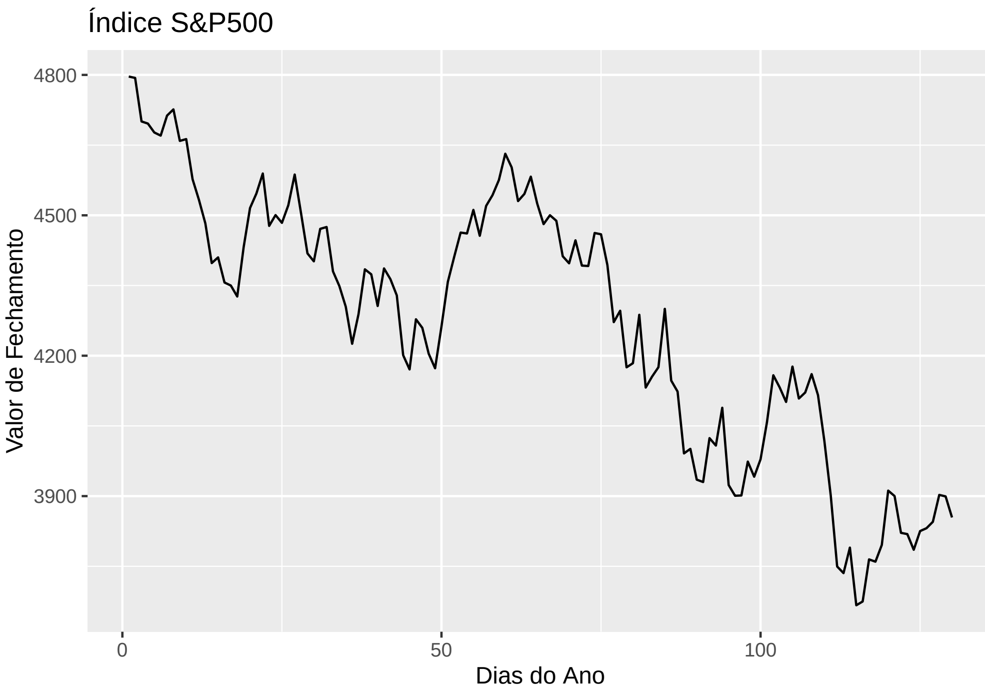
Prova GED-17

Arthur Stevenson

2022-07-05

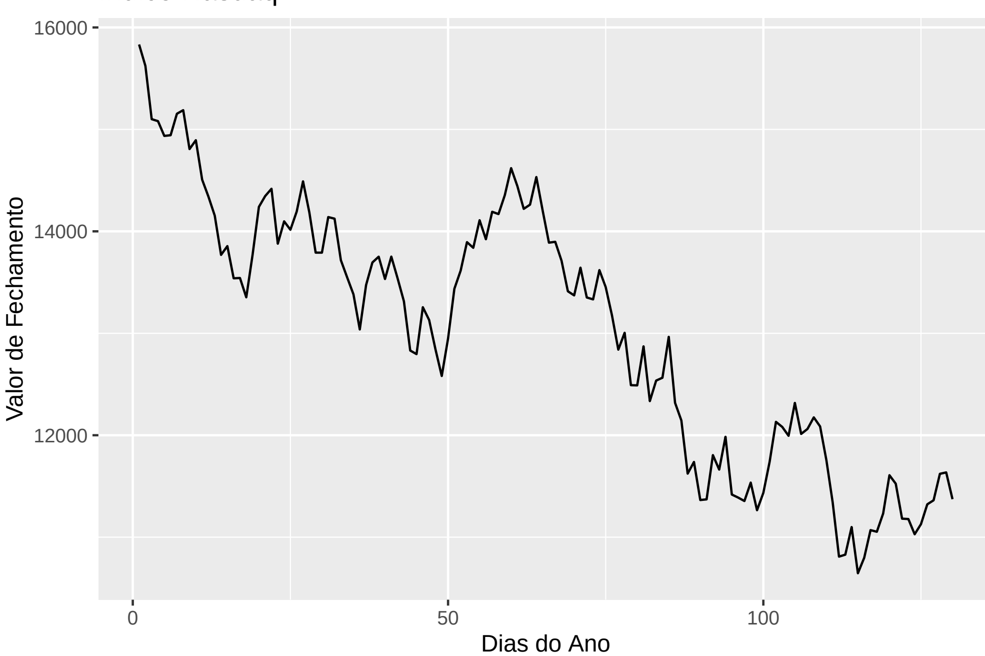
Introdução

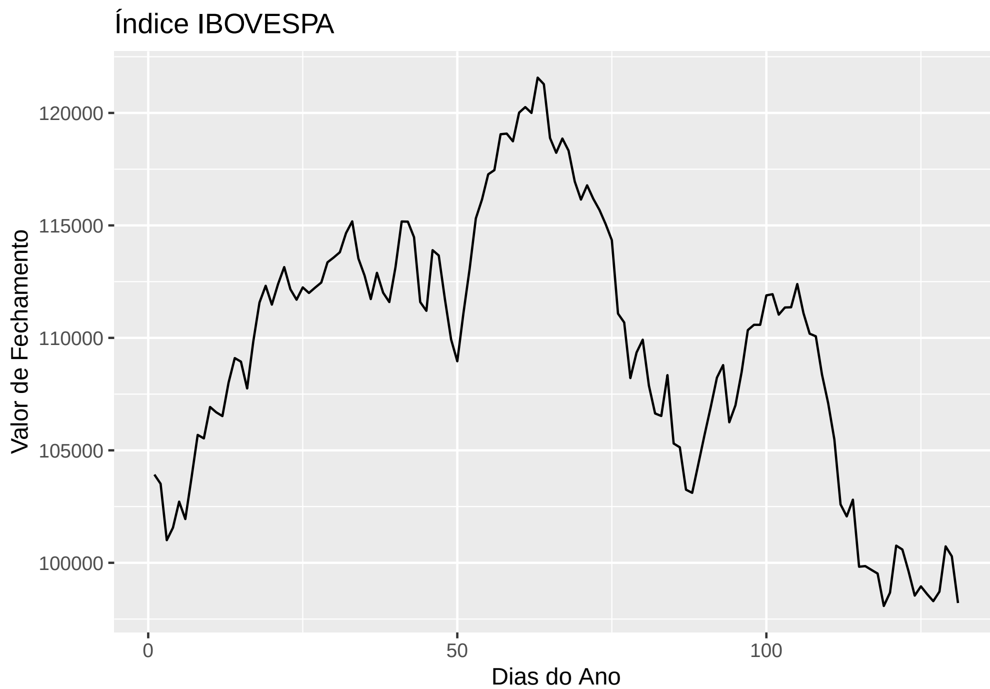
Inicialmente, são fornecidas várias séries temporais, representando os valores de diferentes índices, como o Bovespa, e a cotação de diferentes moedas, como o Dólar. Assim, tratando dos valores de fechamento para essas séries, podemos gerar os seus respectivos plots.

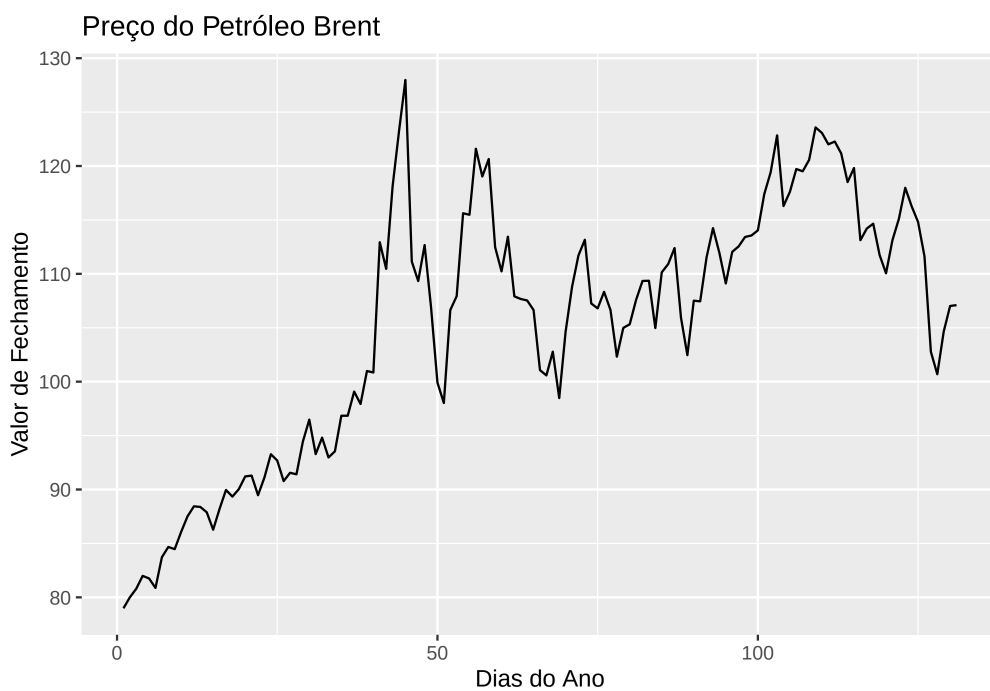


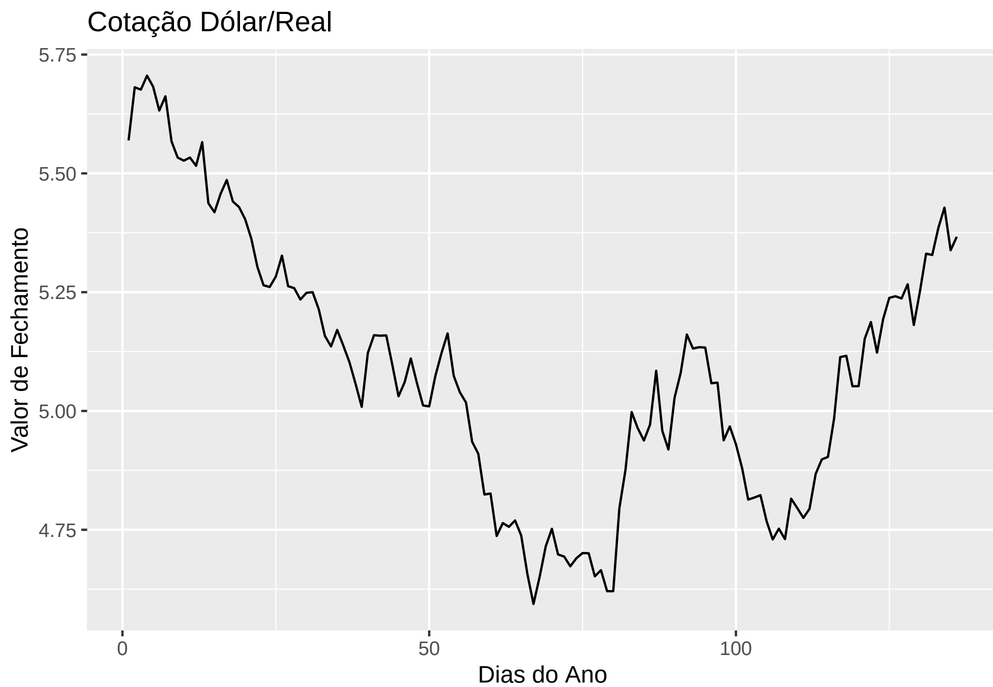
Índice Dow Jones

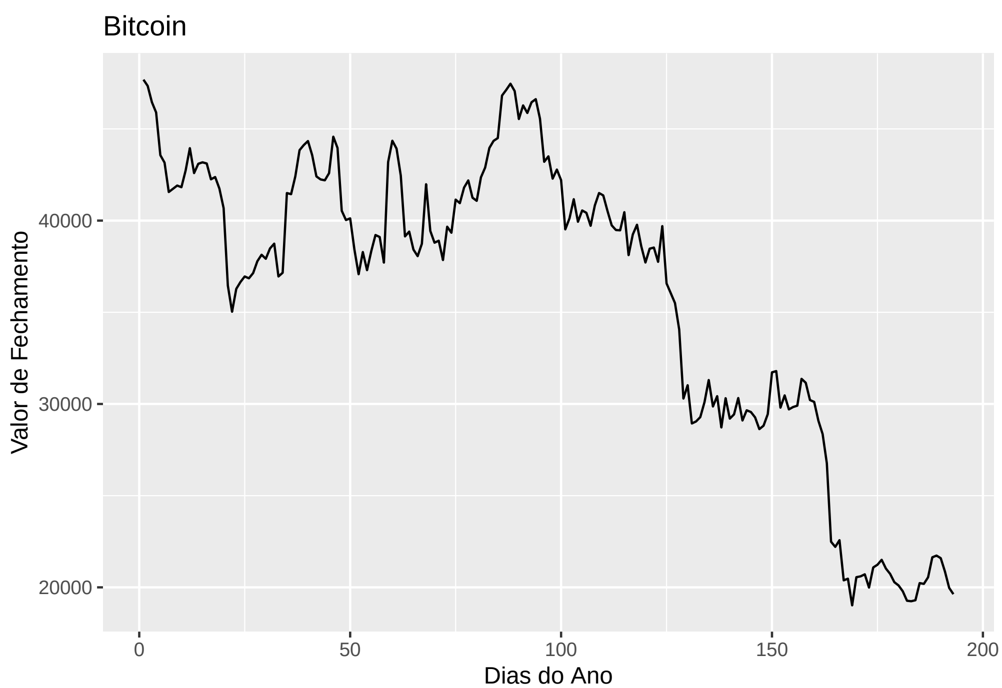
Índice Nasdaq




Índice IBOVESPA
120000
115000
110000
105000
100000
0
50
100
Valor de Fechamento
Dias do Ano


Preço do Petróleo Brent
130
120
110
100
90
80
0
50
100
Valor de Fechamento
Dias do Ano





Em seguida, podemos fazer o teste ADF para as séries temporais originais obtidas.

```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag   ADF p.value
## [1,]   0 -1.37   0.188
## [2,]   1 -1.33   0.201
## [3,]   2 -1.25   0.229
## [4,]   3 -1.27   0.224
```

```
## [5,]   4 -1.17   0.261
## Type 2: with drift no trend
##      lag   ADF p.value
## [1,]   0 -1.61   0.481
## [2,]   1 -1.66   0.462
## [3,]   2 -1.40   0.557
## [4,]   3 -1.40   0.558
## [5,]   4 -1.42   0.548
## Type 3: with drift and trend
##      lag   ADF p.value
## [1,]   0 -2.66   0.298
## [2,]   1 -2.78   0.250
## [3,]   2 -2.60   0.325
## [4,]   3 -2.59   0.329
## [5,]   4 -2.77   0.256
## ----
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag   ADF p.value
## [1,]   0 -1.22   0.239
## [2,]   1 -1.22   0.240
## [3,]   2 -1.20   0.247
## [4,]   3 -1.17   0.257
## [5,]   4 -1.10   0.284
## Type 2: with drift no trend
##      lag   ADF p.value
## [1,]   0 -1.66   0.462
## [2,]   1 -1.84   0.392
## [3,]   2 -1.61   0.481
## [4,]   3 -1.58   0.491
## [5,]   4 -1.68   0.452
## Type 3: with drift and trend
##      lag   ADF p.value
## [1,]   0 -2.78   0.252
## [2,]   1 -3.02   0.150
## [3,]   2 -2.80   0.243
## [4,]   3 -2.83   0.229
## [5,]   4 -3.08   0.128
## ----
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag   ADF p.value
## [1,]   0 -1.60   0.103
## [2,]   1 -1.53   0.129
## [3,]   2 -1.38   0.182
## [4,]   3 -1.46   0.155
## [5,]   4 -1.35   0.194
```

```
## Type 2: with drift no trend
##      lag   ADF p.value
## [1,]   0 -1.91   0.362
## [2,]   1 -1.80   0.405
## [3,]   2 -1.48   0.528
## [4,]   3 -1.45   0.538
## [5,]   4 -1.39   0.560
## Type 3: with drift and trend
##      lag   ADF p.value
## [1,]   0 -2.96   0.177
## [2,]   1 -2.92   0.194
## [3,]   2 -2.71   0.277
## [4,]   3 -2.59   0.327
## [5,]   4 -2.68   0.293
## ----
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag    ADF p.value
## [1,]   0 -0.423   0.522
## [2,]   1 -0.371   0.537
## [3,]   2 -0.228   0.578
## [4,]   3 -0.278   0.563
## [5,]   4 -0.299   0.558
## Type 2: with drift no trend
##      lag    ADF p.value
## [1,]   0 -0.679   0.809
## [2,]   1 -1.018   0.690
## [3,]   2 -1.008   0.694
## [4,]   3 -1.064   0.674
## [5,]   4 -1.284   0.597
## Type 3: with drift and trend
##      lag   ADF p.value
## [1,]   0 -1.55   0.761
## [2,]   1 -1.84   0.639
## [3,]   2 -2.11   0.524
## [4,]   3 -2.11   0.525
## [5,]   4 -2.26   0.465
## ----
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag   ADF p.value
## [1,]   0 0.402   0.758
## [2,]   1 0.399   0.757
## [3,]   2 0.399   0.757
## [4,]   3 0.417   0.762
## [5,]   4 0.350   0.743
## Type 2: with drift no trend
```

```
##      lag   ADF p.value
## [1,]   0 -2.48   0.141
## [2,]   1 -2.45   0.153
## [3,]   2 -2.43   0.161
## [4,]   3 -2.35   0.192
## [5,]   4 -2.50   0.134
## Type 3: with drift and trend
##      lag   ADF p.value
## [1,]   0 -2.68   0.293
## [2,]   1 -2.66   0.299
## [3,]   2 -2.66   0.301
## [4,]   3 -2.50   0.367
## [5,]   4 -2.97   0.172
## ----
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag    ADF p.value
## [1,]   0 -0.398   0.529
## [2,]   1 -0.565   0.477
## [3,]   2 -0.542   0.485
## [4,]   3 -0.614   0.459
## [5,]   4 -0.536   0.487
## Type 2: with drift no trend
##      lag   ADF p.value
## [1,]   0 -1.61   0.480
## [2,]   1 -2.06   0.306
## [3,]   2 -1.96   0.345
## [4,]   3 -2.12   0.281
## [5,]   4 -2.15   0.272
## Type 3: with drift and trend
##      lag    ADF p.value
## [1,]   0 -0.713   0.967
## [2,]   1 -1.098   0.920
## [3,]   2 -0.868   0.953
## [4,]   3 -0.904   0.950
## [5,]   4 -1.037   0.929
## ----
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag   ADF p.value
## [1,]   0 -1.69  0.0885
## [2,]   1 -1.66  0.0938
## [3,]   2 -1.57  0.1150
## [4,]   3 -1.45  0.1582
## [5,]   4 -1.34  0.1993
## Type 2: with drift no trend
##      lag    ADF p.value
```

```
## [1,]   0 -0.652   0.819
## [2,]   1 -0.639   0.823
## [3,]   2 -0.597   0.838
## [4,]   3 -0.667   0.813
## [5,]   4 -0.456   0.887
## Type 3: with drift and trend
##      lag   ADF p.value
## [1,]   0 -1.71   0.697
## [2,]   1 -1.71   0.695
## [3,]   2 -1.74   0.684
## [4,]   3 -1.89   0.619
## [5,]   4 -1.86   0.634
## ----
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

Diferença do Logaritmo

Analisando-se os resultados dos ADFs de cada uma das séries temporais, nota-se que todos os p-valores encontrados são superiores a 5%. Portanto, não podemos rejeitar a hipótese nula e, assim, é necessária uma transformação na série para que seja possível trabalhá-la.

Com os plots das séries temporais, podemos, então, obter as séries do log, e da diferença do log e, assim realizar o teste ADF novamente. Analisando-se os resultados, nota-se que há uma queda considerável nos p-valores de todas as séries analisadas.

Nesses novos casos, constata-se que os p-valores das séries são, novamente, significativamente menores e, mais importante que isso, que eles são inferiores à margem significativa de 5%. Dessa forma, podemos rejeitar a hipótese nula, e é seguro supor que a série é estacionária.

```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag    ADF p.value
## [1,]   0 -10.91    0.01
## [2,]   1  -8.18    0.01
## [3,]   2  -6.58    0.01
## [4,]   3  -5.32    0.01
## [5,]   4  -5.07    0.01
## Type 2: with drift no trend
##      lag    ADF p.value
## [1,]   0 -10.99    0.01
## [2,]   1  -8.27    0.01
## [3,]   2  -6.69    0.01
## [4,]   3  -5.43    0.01
## [5,]   4  -5.19    0.01
## Type 3: with drift and trend
##      lag    ADF p.value
## [1,]   0 -10.95    0.01
## [2,]   1  -8.23    0.01
## [3,]   2  -6.66    0.01
## [4,]   3  -5.40    0.01
## [5,]   4  -5.17    0.01
```

```
## ----
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag    ADF p.value
## [1,]   0 -10.67    0.01
## [2,]   1  -8.27    0.01
## [3,]   2  -6.54    0.01
## [4,]   3  -5.25    0.01
## [5,]   4  -5.00    0.01
## Type 2: with drift no trend
##      lag    ADF p.value
## [1,]   0 -10.74    0.01
## [2,]   1  -8.35    0.01
## [3,]   2  -6.63    0.01
## [4,]   3  -5.34    0.01
## [5,]   4  -5.11    0.01
## Type 3: with drift and trend
##      lag    ADF p.value
## [1,]   0 -10.70    0.01
## [2,]   1  -8.32    0.01
## [3,]   2  -6.61    0.01
## [4,]   3  -5.32    0.01
## [5,]   4  -5.09    0.01
## ----
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag    ADF p.value
## [1,]   0 -11.35    0.01
## [2,]   1  -8.14    0.01
## [3,]   2  -6.85    0.01
## [4,]   3  -5.63    0.01
## [5,]   4  -5.26    0.01
## Type 2: with drift no trend
##      lag    ADF p.value
## [1,]   0 -11.46    0.01
## [2,]   1  -8.25    0.01
## [3,]   2  -6.99    0.01
## [4,]   3  -5.76    0.01
## [5,]   4  -5.43    0.01
## Type 3: with drift and trend
##      lag    ADF p.value
## [1,]   0 -11.43    0.01
## [2,]   1  -8.21    0.01
## [3,]   2  -6.96    0.01
## [4,]   3  -5.74    0.01
## [5,]   4  -5.41    0.01
## ----
```

```
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

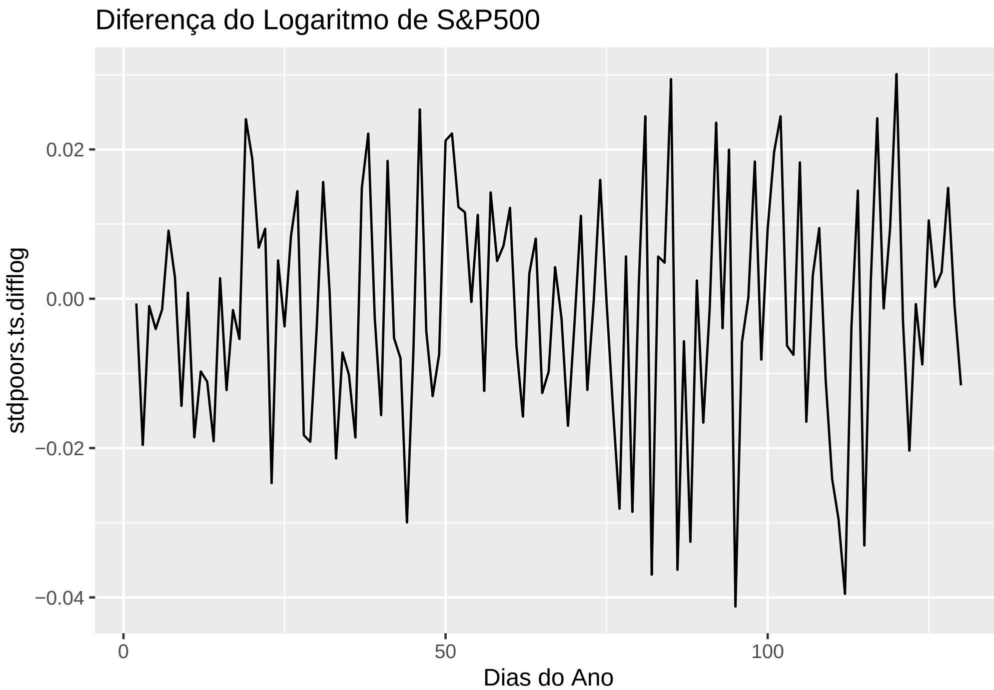
```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag   ADF p.value
## [1,]   0 -9.28    0.01
## [2,]   1 -8.17    0.01
## [3,]   2 -6.01    0.01
## [4,]   3 -4.60    0.01
## [5,]   4 -4.01    0.01
## Type 2: with drift no trend
##      lag   ADF p.value
## [1,]   0 -9.26    0.01
## [2,]   1 -8.14    0.01
## [3,]   2 -5.99    0.01
## [4,]   3 -4.58    0.01
## [5,]   4 -4.00    0.01
## Type 3: with drift and trend
##      lag   ADF p.value
## [1,]   0 -9.51    0.01
## [2,]   1 -8.63    0.01
## [3,]   2 -6.46    0.01
## [4,]   3 -4.99    0.01
## [5,]   4 -4.48    0.01
## ----
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

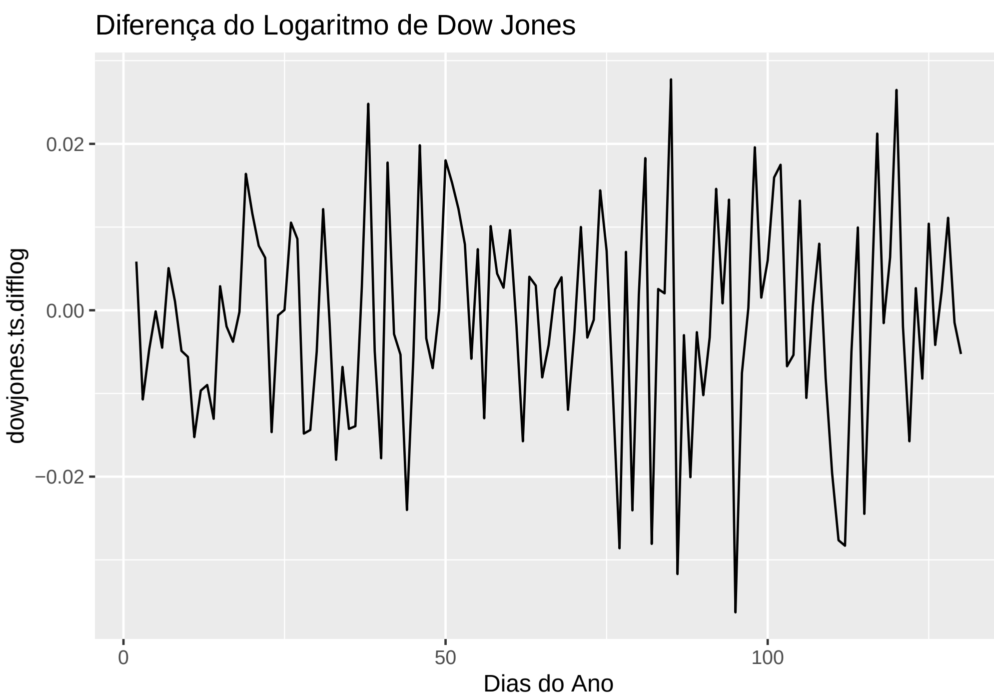
```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag    ADF p.value
## [1,]   0 -11.48    0.01
## [2,]   1  -8.17    0.01
## [3,]   2  -6.96    0.01
## [4,]   3  -5.31    0.01
## [5,]   4  -5.79    0.01
## Type 2: with drift no trend
##      lag    ADF p.value
## [1,]   0 -11.49    0.01
## [2,]   1  -8.20    0.01
## [3,]   2  -6.99    0.01
## [4,]   3  -5.35    0.01
## [5,]   4  -5.86    0.01
## Type 3: with drift and trend
##      lag    ADF p.value
## [1,]   0 -11.57    0.01
## [2,]   1  -8.29    0.01
## [3,]   2  -7.11    0.01
## [4,]   3  -5.47    0.01
## [5,]   4  -6.06    0.01
## ----
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

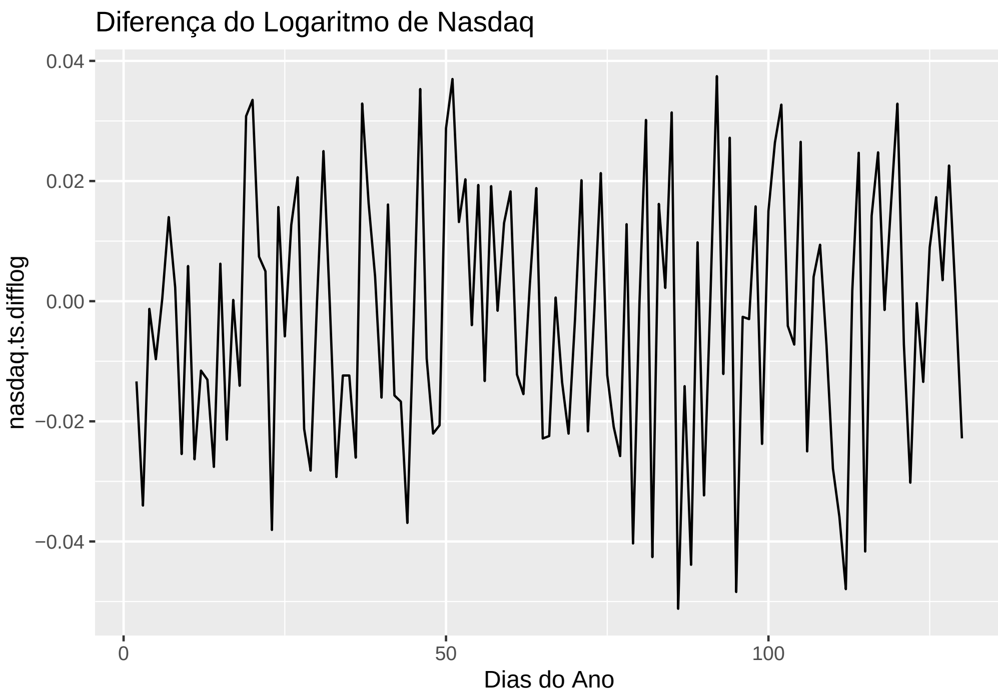
```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag    ADF p.value
## [1,]   0 -10.42    0.01
## [2,]   1  -8.24    0.01
## [3,]   2  -6.80    0.01
## [4,]   3  -5.20    0.01
## [5,]   4  -4.25    0.01
## Type 2: with drift no trend
##      lag    ADF p.value
## [1,]   0 -10.40    0.01
## [2,]   1  -8.22    0.01
## [3,]   2  -6.79    0.01
## [4,]   3  -5.20    0.01
## [5,]   4  -4.24    0.01
## Type 3: with drift and trend
##      lag    ADF p.value
## [1,]   0 -10.75    0.01
## [2,]   1  -8.68    0.01
## [3,]   2  -7.40    0.01
## [4,]   3  -5.80    0.01
## [5,]   4  -4.78    0.01
## ----
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

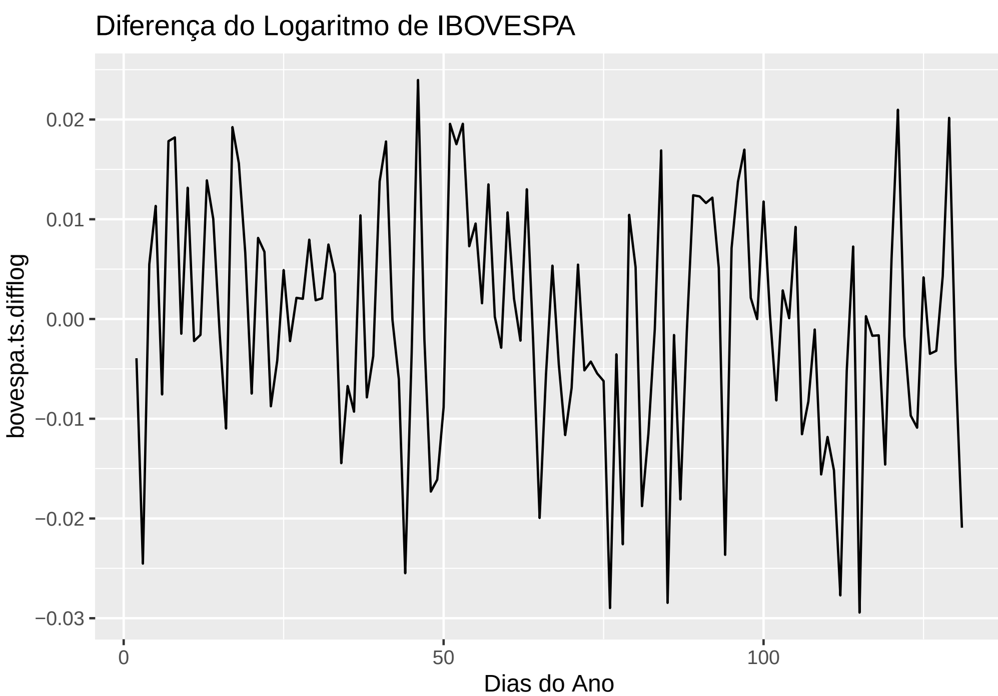
```
## Augmented Dickey-Fuller Test
## alternative: stationary
##
## Type 1: no drift no trend
##      lag    ADF p.value
## [1,]   0 -13.74    0.01
## [2,]   1  -9.00    0.01
## [3,]   2  -6.90    0.01
## [4,]   3  -6.10    0.01
## [5,]   4  -5.65    0.01
## Type 2: with drift no trend
##      lag    ADF p.value
## [1,]   0 -13.92    0.01
## [2,]   1  -9.17    0.01
## [3,]   2  -7.07    0.01
## [4,]   3  -6.26    0.01
## [5,]   4  -5.83    0.01
## Type 3: with drift and trend
##      lag    ADF p.value
## [1,]   0 -13.94    0.01
## [2,]   1  -9.21    0.01
## [3,]   2  -7.12    0.01
## [4,]   3  -6.34    0.01
## [5,]   4  -5.92    0.01
## ----
## Note: in fact, p.value = 0.01 means p.value <= 0.01
```

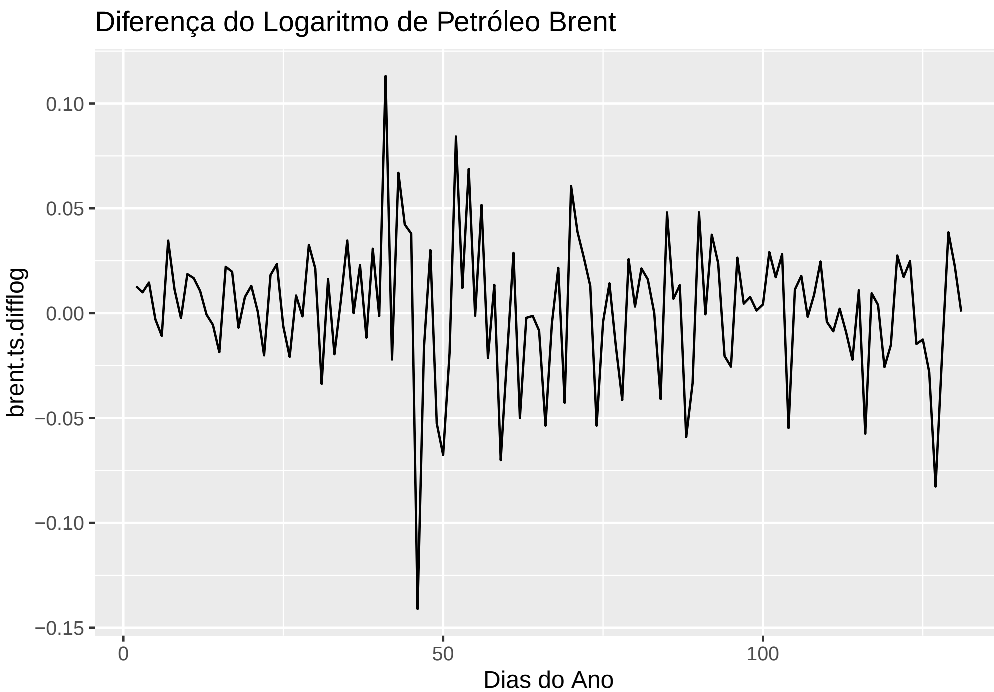
Assim, com os resultados do teste ADF, podemos então realizar o plot das séries da diferença do logaritmo.

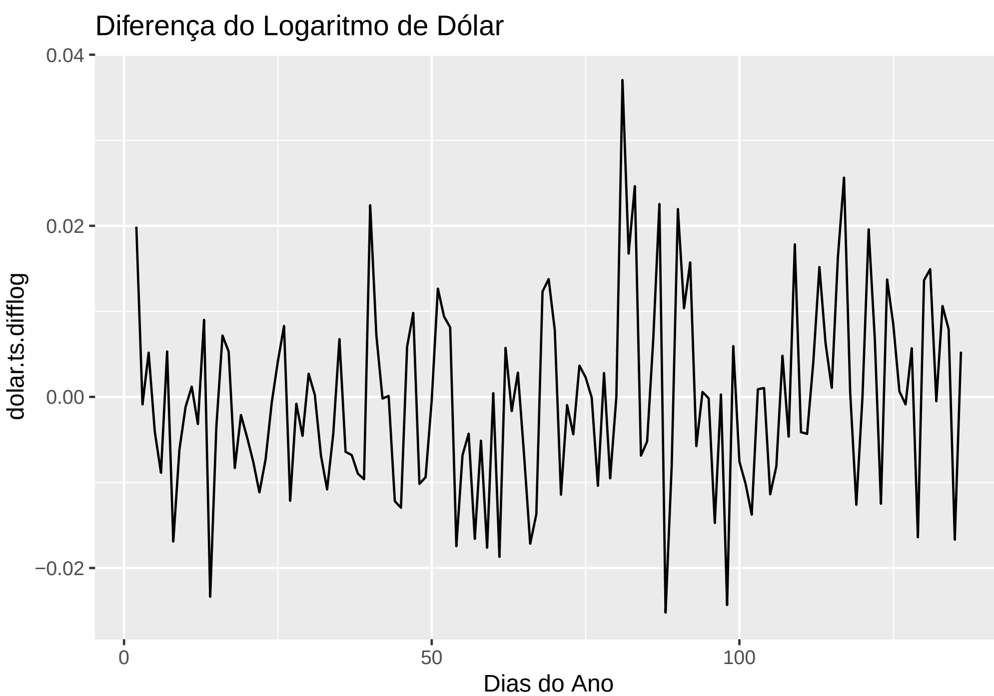


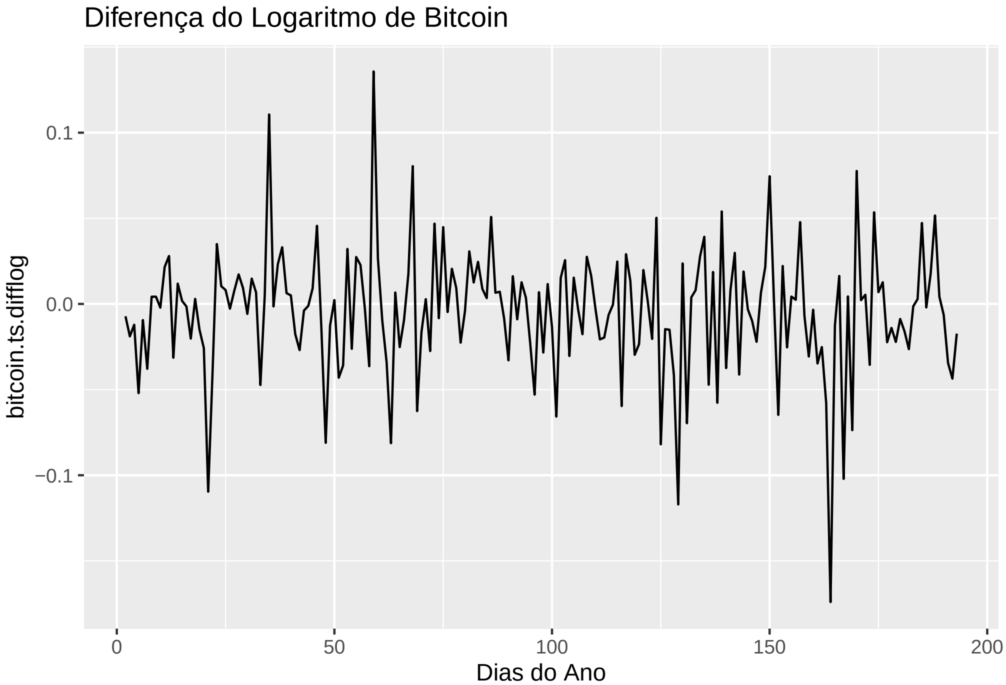



Diferença do Logaritmo de Nasdaq
nasdaq.ts.difflog
0.04
0.02
0.00
−0.02
−0.04
0
50
100
Dias do Ano


Diferença do Logaritmo de IBOVESPA
bovespa.ts.difflog
0.02
0.01
0.00
−0.01
−0.02
−0.03
0
50
100
Dias do Ano

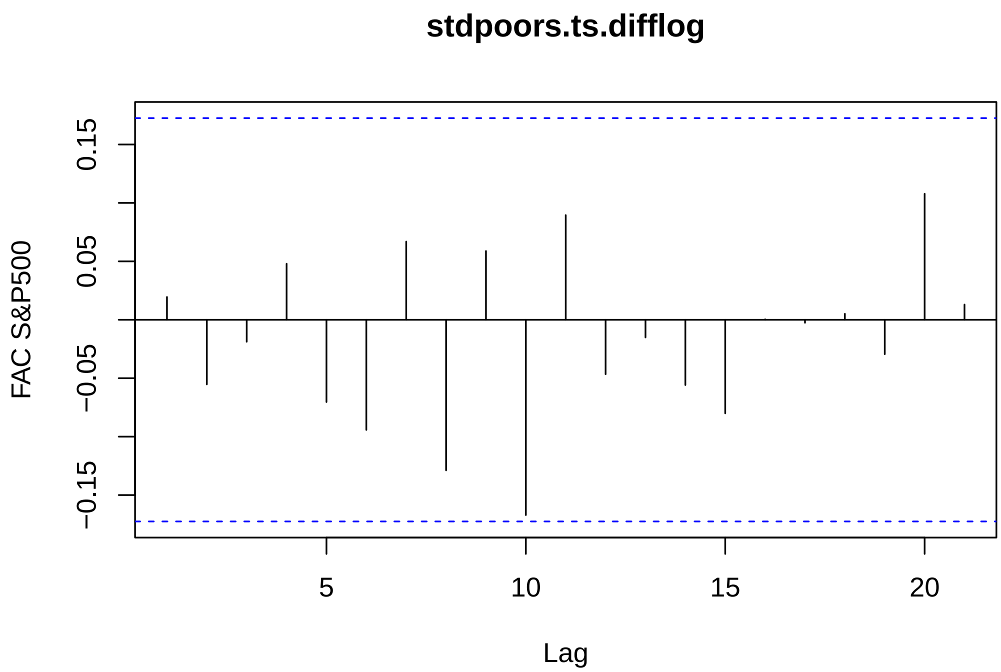

Diferença do Logaritmo de Petróleo Brent
brent.ts.difflog
0.10
0.05
0.00
−0.05
−0.10
−0.15
0
50
100
Dias do Ano

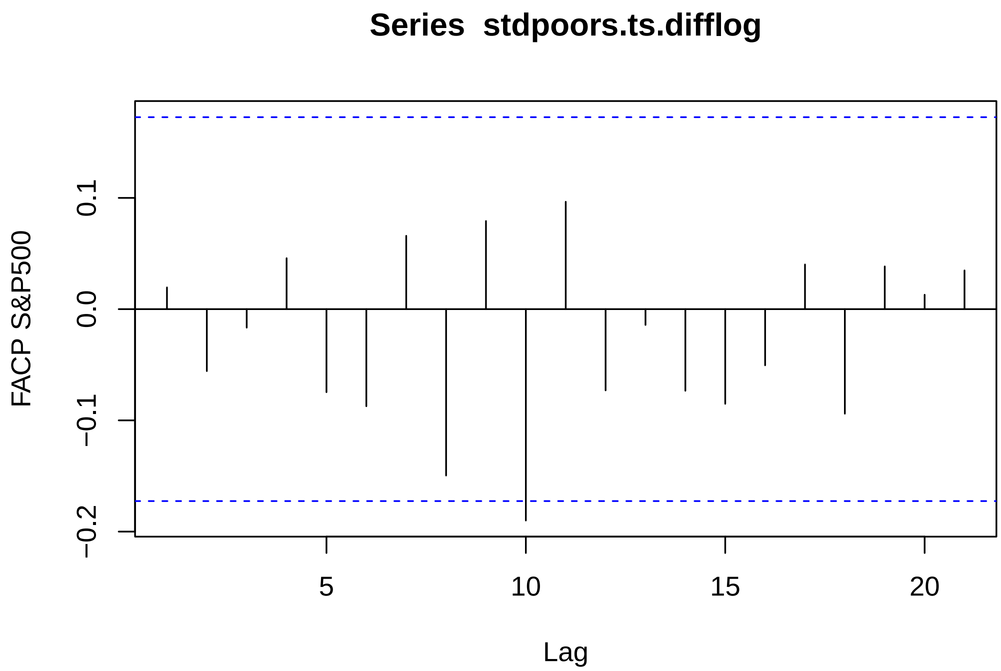

Diferença do Logaritmo de Dólar
dolar.ts.difflog
0.04
0.02
0.00
−0.02
0
50
100
Dias do Ano

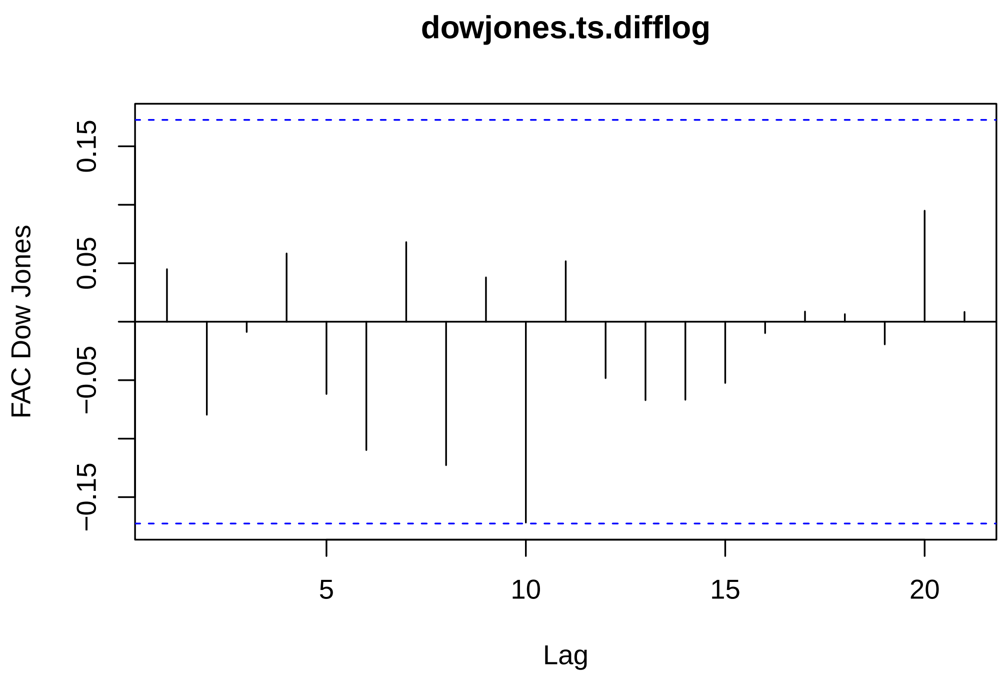


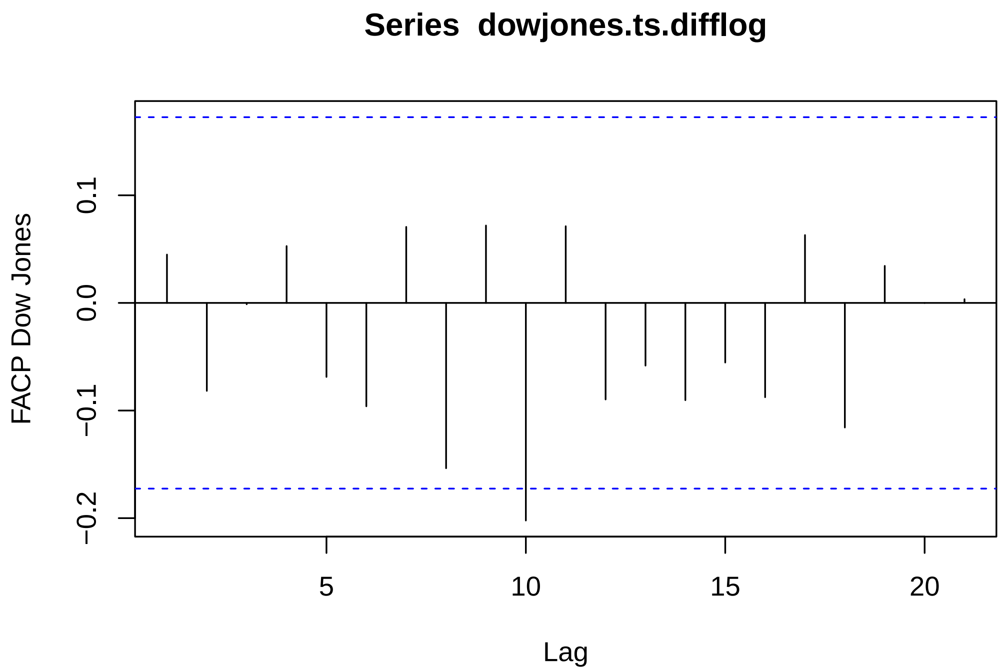
Correlogramas

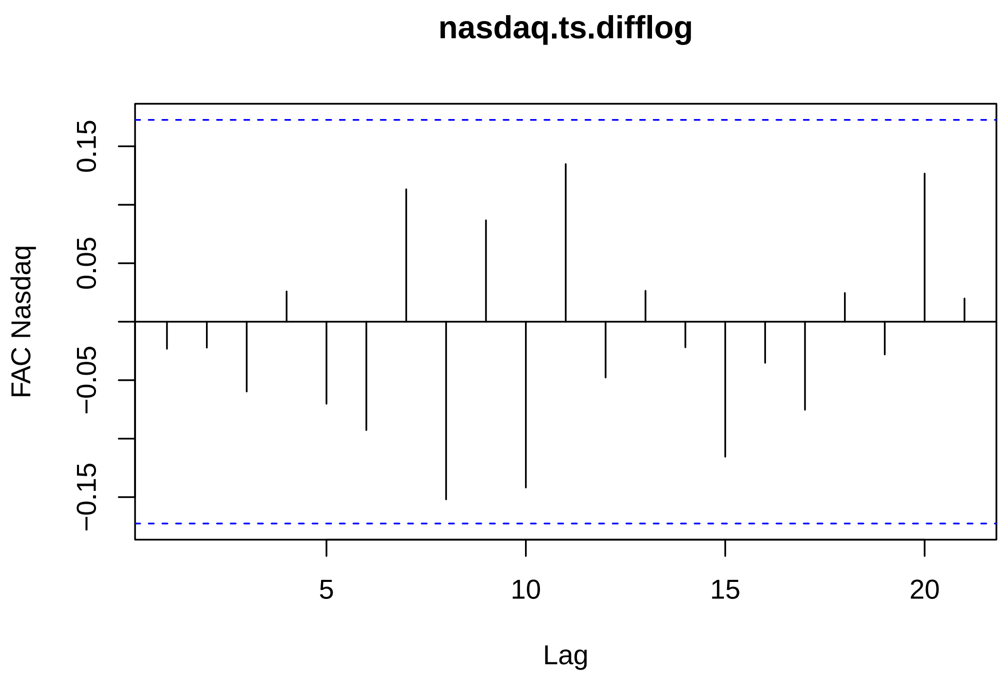
Feitos os plots das diferenças dos logaritmos, podemos, então, construir o correlograma da série da diferença do logaritmo de cada uma das séries criadas.

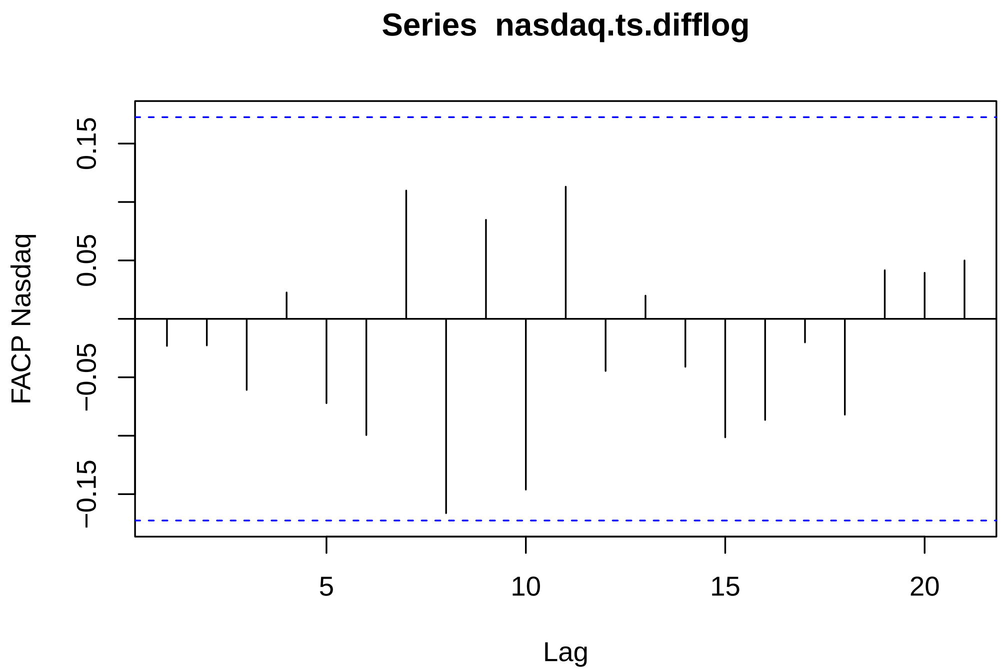


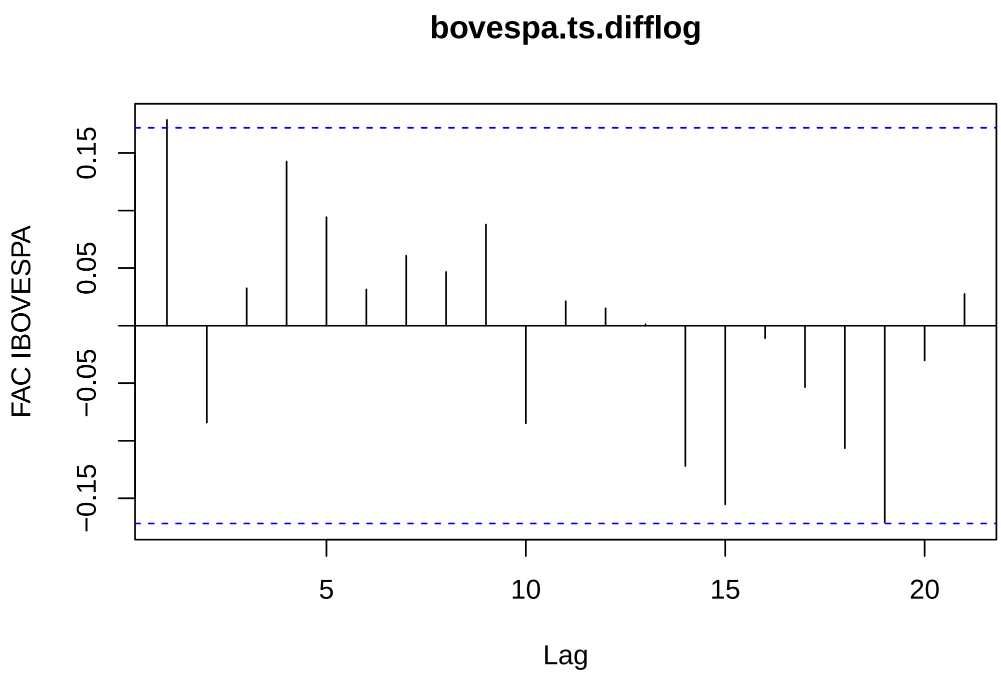

Series stdpoors.ts.difflog
FACP S&P500
0.1
0.0
−0.1
−0.2
5
10
15
20
Lag

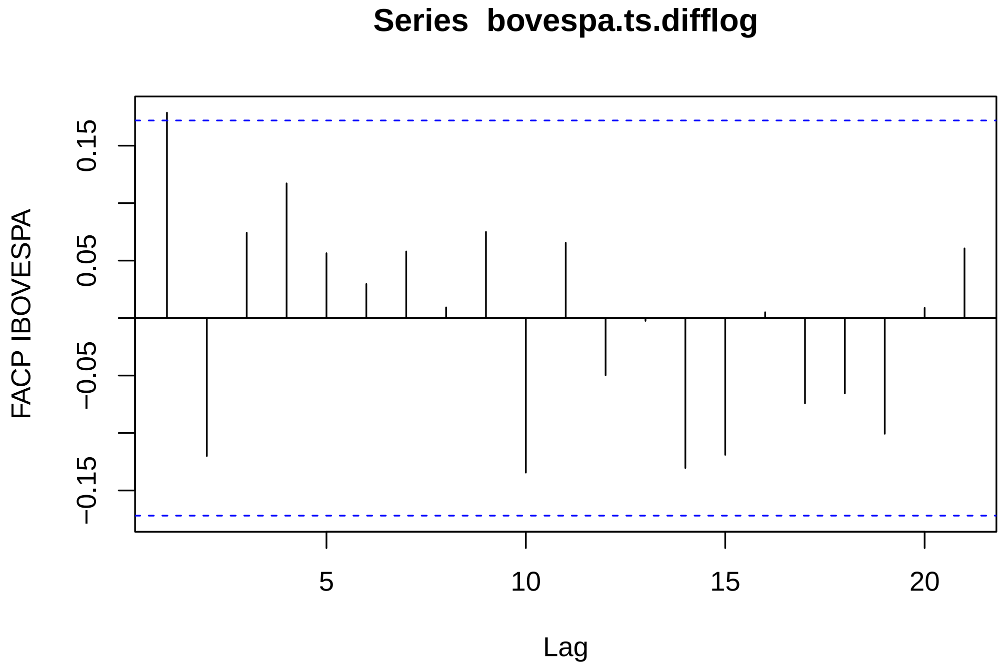

dowjones.ts.difflog
FAC Dow Jones
0.15
0.05
−0.05
−0.15
5
10
15
20
Lag

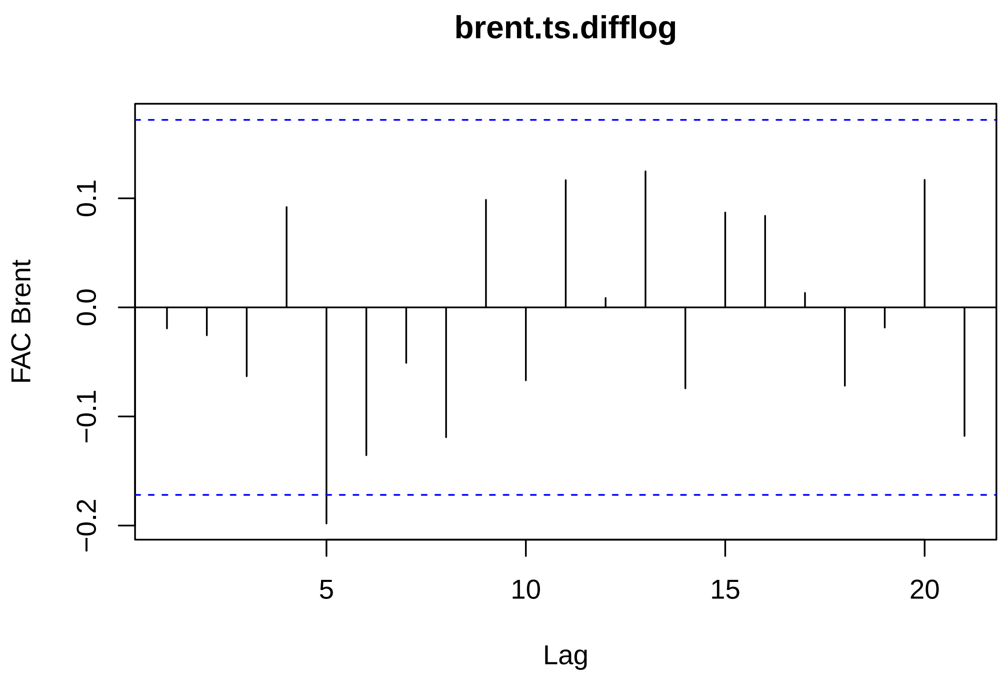

Series dowjones.ts.difflog
FACP Dow Jones
0.1
0.0
−0.1
−0.2
5
10
15
20
Lag

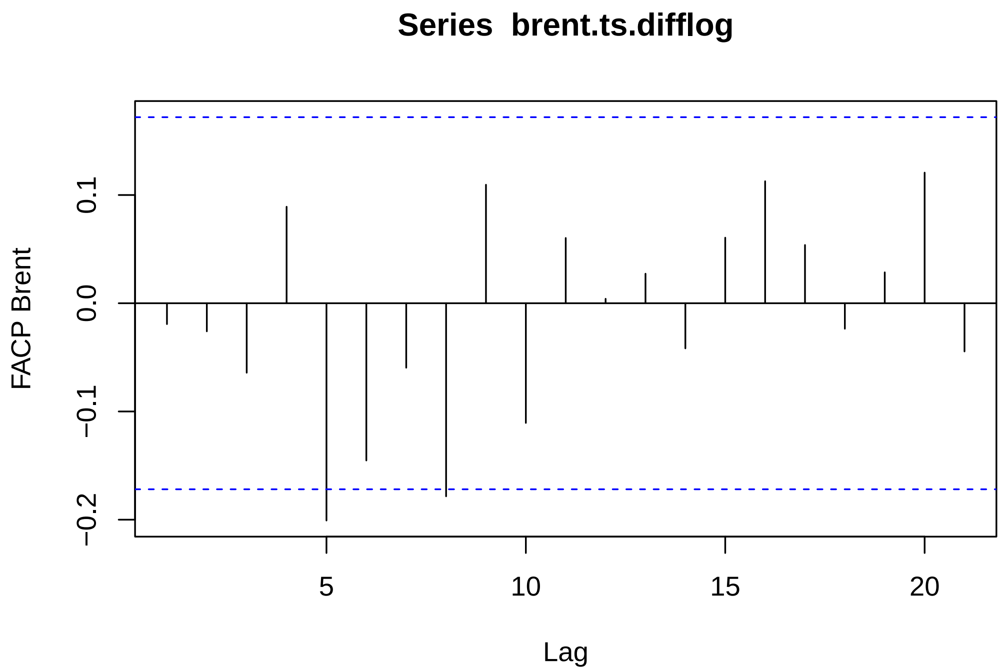

nasdaq.ts.difflog
FAC Nasdaq
0.15
0.05
−0.05
−0.15
5
10
15
20
Lag

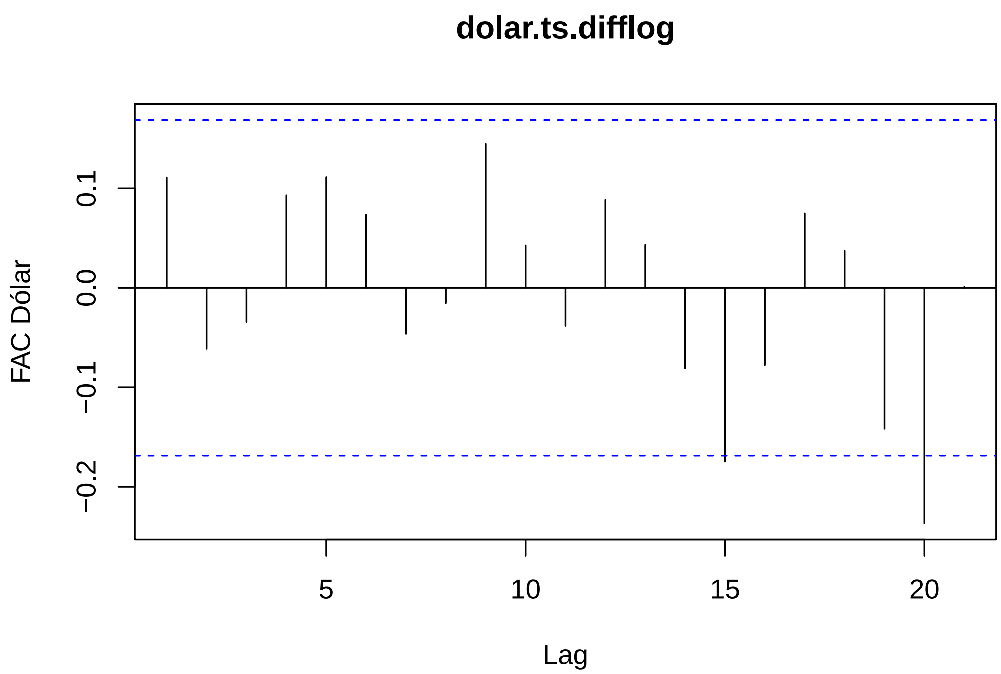

Series nasdaq.ts.difflog
FACP Nasdaq
0.15
0.05
−0.05
−0.15
5
10
15
20
Lag

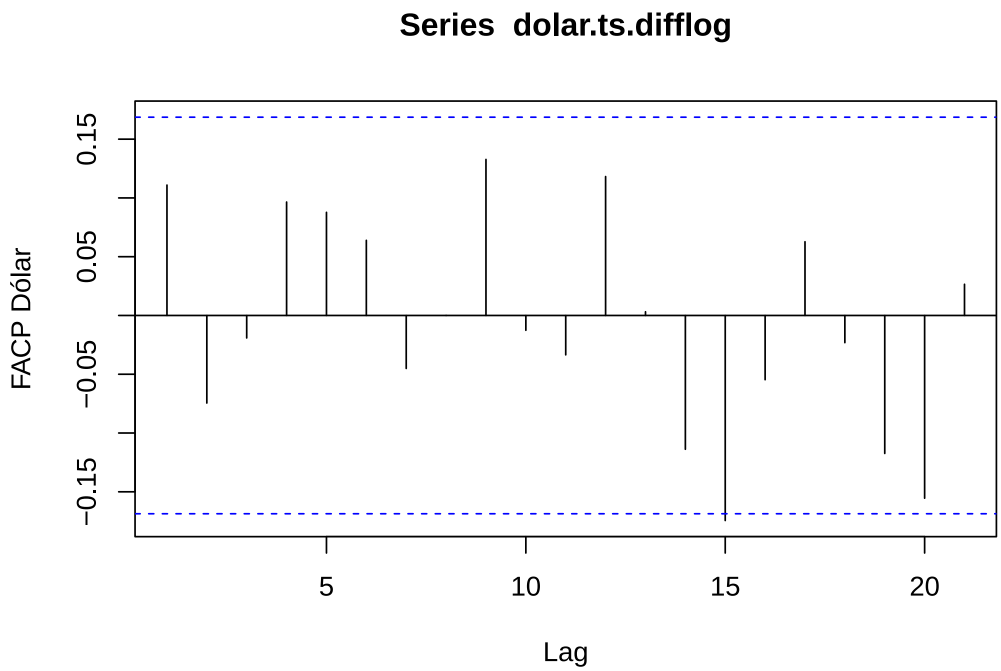

bovespa.ts.difflog
FAC IBOVESPA
0.15
0.05
−0.05
−0.15
5
10
15
20
Lag

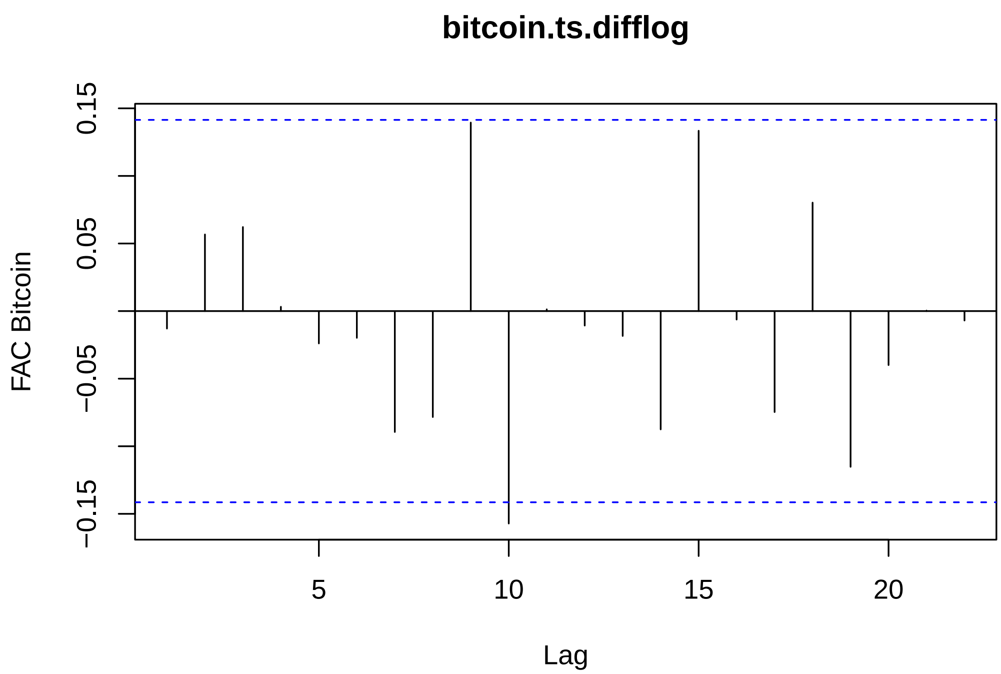

Series bovespa.ts.difflog
FACP IBOVESPA
0.15
0.05
−0.05
−0.15
5
10
15
20
Lag

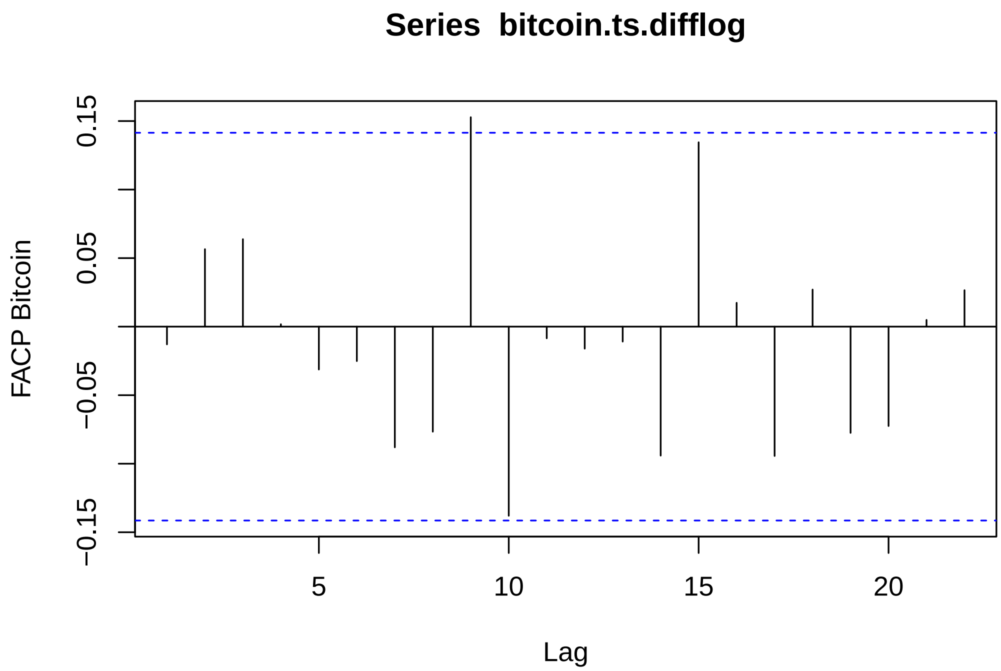

brent.ts.difflog
FAC Brent
0.1
0.0
−0.1
−0.2
5
10
15
20
Lag


Series brent.ts.difflog
FACP Brent
0.1
0.0
−0.1
−0.2
5
10
15
20
Lag


dolar.ts.difflog
FAC Dólar
0.1
0.0
−0.1
−0.2
5
10
15
20
Lag


Series dolar.ts.difflog
FACP Dólar
0.15
0.05
−0.05
−0.15
5
10
15
20
Lag


bitcoin.ts.difflog
FAC Bitcoin
0.15
0.05
−0.05
−0.15
5
10
15
20
Lag



Assim, observando-se os correlogramas gerados, pode ser extrapolado o melhor modelo para cada uma das séries analisadas. Dessa forma, temos que:

- S&P500: FAC e FACP ambas declinantes. Consideramos ARMA(1,1).
- Dow Jones: FAC e FACP declinantes. Consideramos ARMA(1,1) a ARMA(2,2), com as combinações intermediárias.
- Nasdaq: FAC e FACP declinantes. Consideramos ARMA(3,3).
- IBOVESPA: FAC e FACP declinantes. Consideramos de ARMA(1,1) a ARMA(2,2), com as combinações intermediárias.
- Brent: FAC e FACP declinantes. Consideramos de ARMA(1, 1) a ARMA(3,3), com as combinações intermediárias.
- Dólar: FAC e FACP declinantes. Consideramos ARMA(1,1).
- Bitcoin: FAC e FACP declinantes. Consideramos ARMA(1,1).

Portanto, podemos então estimar os diferentes modelos ARMA para cada uma das séries analisadas, e verificar a significância dos coeficientes de cada um. Ademais, podem ser feitos os testes e plotados o gráfico do teste de Ljung-Box, para analisar os resíduos de cada modelo.

p-valores

S&P500

```
##                     ar1           ma1     intercept
## ar1        0.0003214643 -0.0004325517     0.1405391
## ma1       -0.0004325517  0.0086887481    -0.3388139
## intercept  0.1405391064 -0.3388138600 51763.4659118
```

```
## intercept
##         0
```

```
## ar1
##   0
```

```
##       ma1
## 0.3062257
```

Analisando os p-valores, nota-se que o referente ao índice 1 do modelo MA é superior a 5%. Assim, consideramos um modelo AR(1) para a série S&P500.

Dow Jones

```
##                    ar1          ar2          ma1          ma2     intercept
## ar1        0.006816261 -0.005682737 -0.006886562 -0.004745388 -2.157770e-02
## ar2       -0.005682737  0.005966680  0.005548731  0.003539411 -2.967382e-01
## ma1       -0.006886562  0.005548731  0.014552679  0.012031417 -1.312437e+00
## ma2       -0.004745388  0.003539411  0.012031417  0.010793724 -1.261882e+00
## intercept -0.021577705 -0.296738185 -1.312437365 -1.261882091  1.424433e+06
```

```
## intercept
##         0
```

```
##          ma1
## 4.731771e-13
```

```
##       ma2
## 0.4875158
```

```
##        ar1
## 0.06735131
```

```
## ar2
##   0
```

Analisando os p-valores, nota-se que os referentes aos índice 2 do modelo MA e ao índice 1 do modelo AR são os únicos superiores a 5%. Portanto, consideramos um modelo MA(1) para a série Dow Jones.

Nasdaq

```
## intercept
##         0
```

```
## ma1
##   0
```

```
##          ma2
## 2.228117e-06
```

```
##       ma3
## 0.5036498
```

```
## ar1
##   1
```

```
## ar2
##   0
```

```
## ar3
##   0
```

Analisando os p-valores, nota-se que os referentes ao índice 3 do modelo MA e ao índice 1 do modelo AR são superiores a 5%. Logo, o modelo considerado para a série Nasdaq é um MA(2).

IBOVESPA

```
## intercept
##         0
```

```
##       ma1
## 0.3784477
```

```
##       ma2
## 0.6562594
```

```
##        ar1
## 0.06152252
```

```
##       ar2
## 0.5167551
```

Analisando os p-valores, temos que o modelo para a série referente a IBOVESPA é um ARMA(2,2).

Petróleo Brent

```
## intercept
##         0
```

```
##          ma1
## 2.054157e-11
```

```
##         ma2
## 0.000851899
```

```
##       ma3
## 0.8553896
```

```
##       ar1
## 0.9154927
```

```
##          ar2
## 0.0001005846
```

```
##          ar3
## 2.775853e-05
```

Analisando os p-valores, temos que o modelo para a série referente ao petróleo Brent é um MA(2).

Dólar

```
##                     ar1           ma1    intercept
## ar1        0.0002650200 -0.0003571131  0.001098815
## ma1       -0.0003571131  0.0084961123 -0.001586646
## intercept  0.0010988148 -0.0015866461  0.040712878
```

```
## intercept
##         0
```

```
## ar1
##   0
```

```
##        ma1
## 0.07977517
```

Analisando os p-valores, temos que o modelo para a série referente ao Dólar é um AR(1).

Bitcoin

```
##                     ar1           ma1     intercept
## ar1        5.657263e-05 -9.040425e-05 -7.678179e+00
## ma1       -9.040425e-05  5.067551e-03  9.508399e+00
## intercept -7.678179e+00  9.508399e+00  7.690328e+07
```

```
##   intercept
## 6.56487e-05
```

```
## ar1
##   0
```

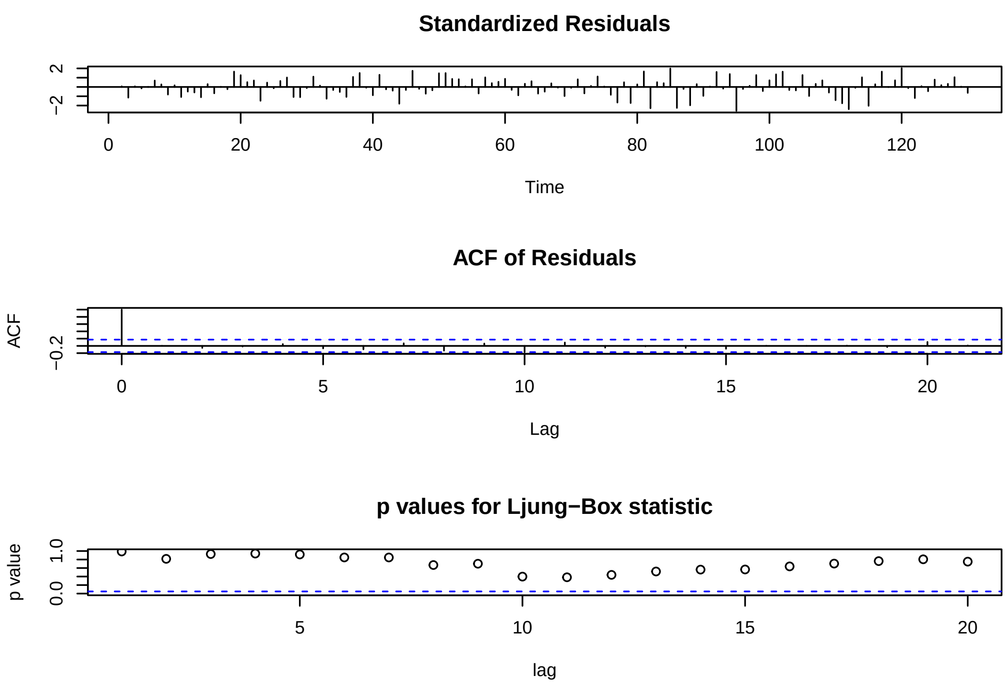
```
##       ma1
## 0.4092939
```

Analisando os p-valores, temos que o modelo para a série referente ao Bitcoin é um AR(1).

Teste de Ljung-Box

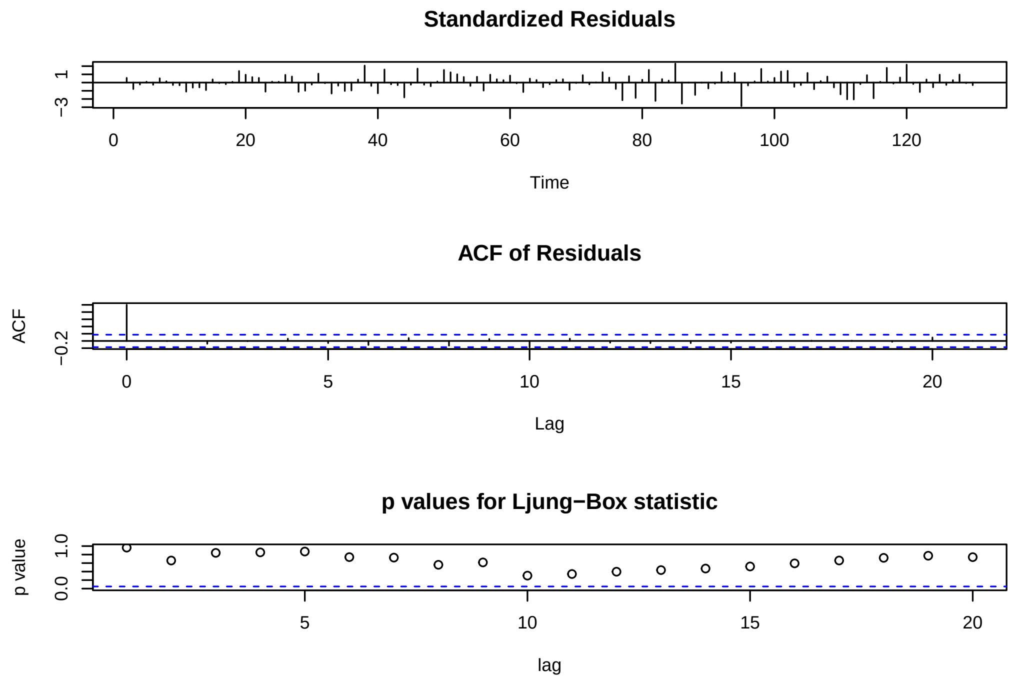
Assim, podemos então plotar os gráficos referentes ao teste de Ljung-Box.

S&P 500



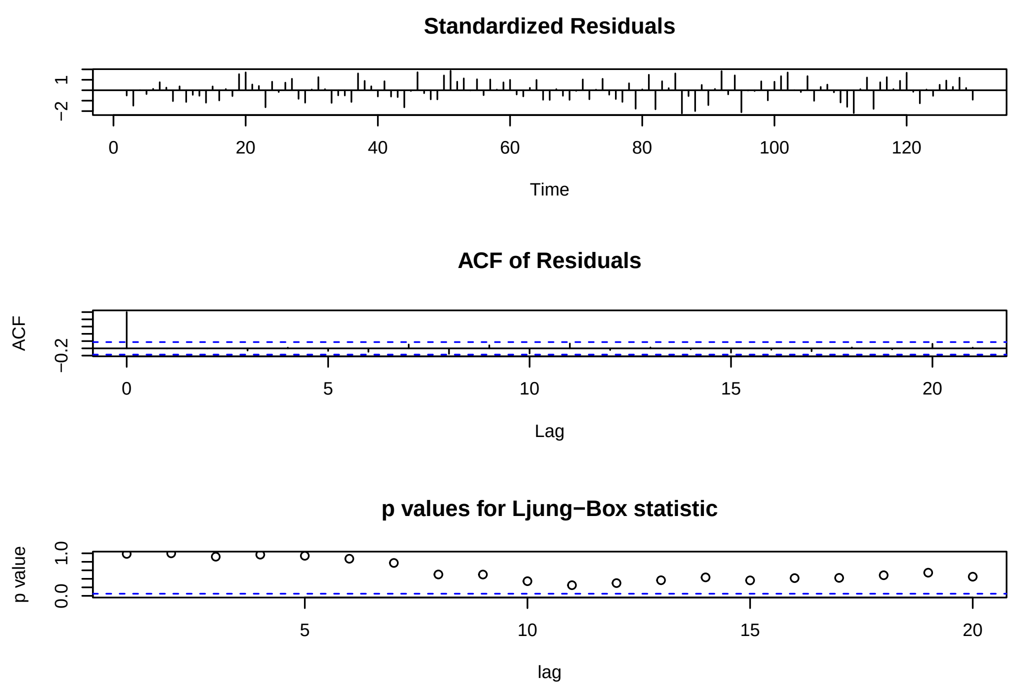

```
##
##  Box-Ljung test
##
## data:  rstandard(modelo.arma)
## X-squared = 15.45, df = 20, p-value = 0.7501
```

Dow Jones




```
##
##  Box-Ljung test
##
## data:  rstandard(modelo.arma)
## X-squared = 15.62, df = 20, p-value = 0.7399
```

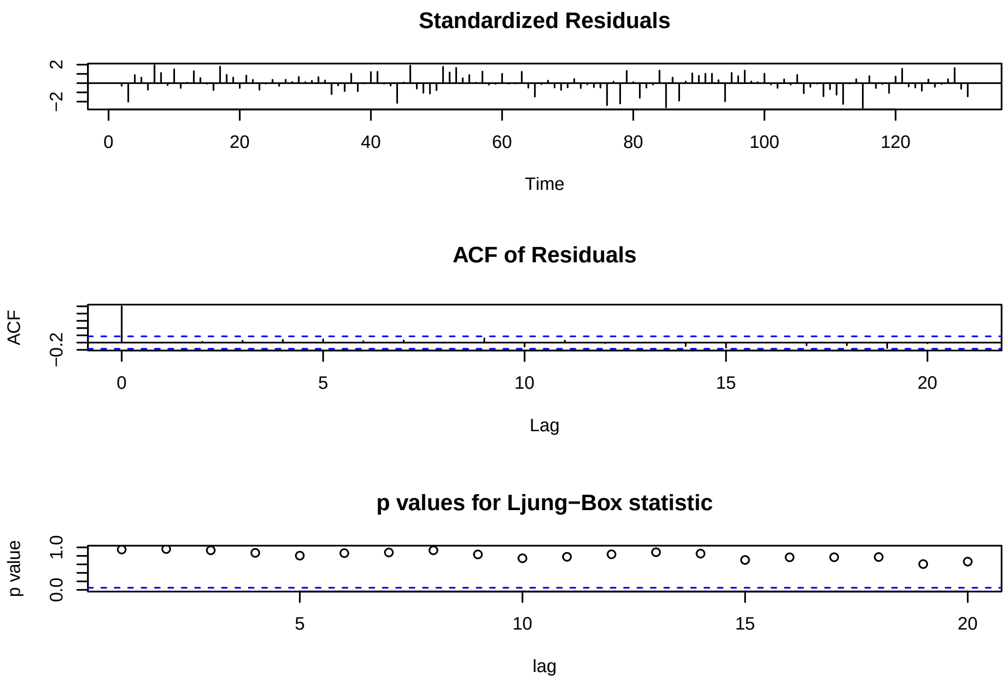
Nasdaq




```
##
```

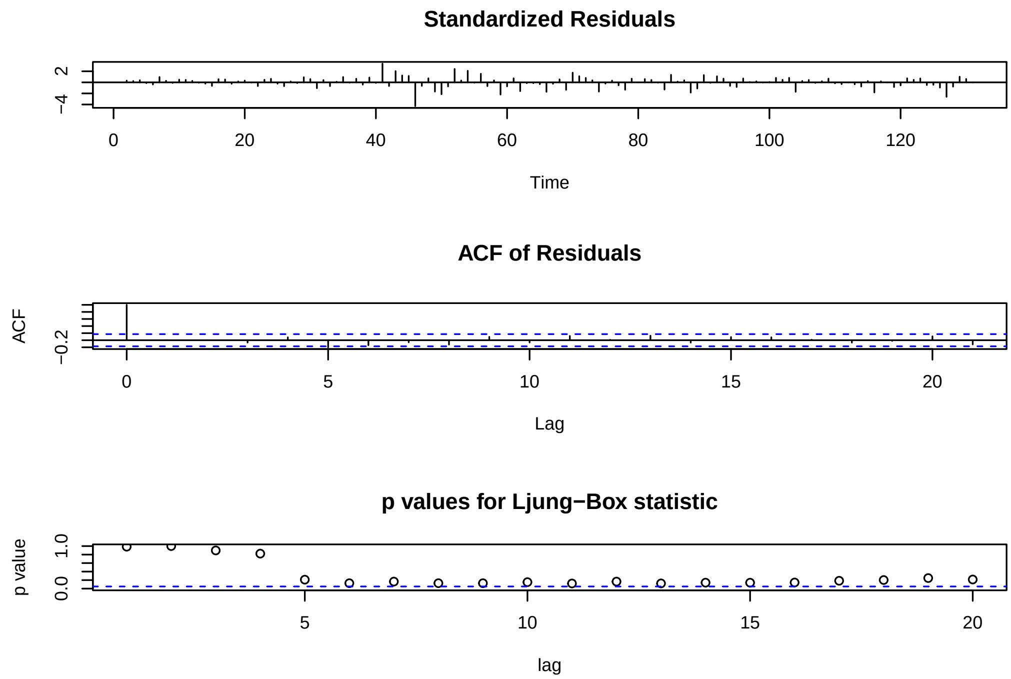
```
##  Box-Ljung test
##
## data:  rstandard(modelo.arma)
## X-squared = 20.125, df = 20, p-value = 0.4501
```

IBOVESPA



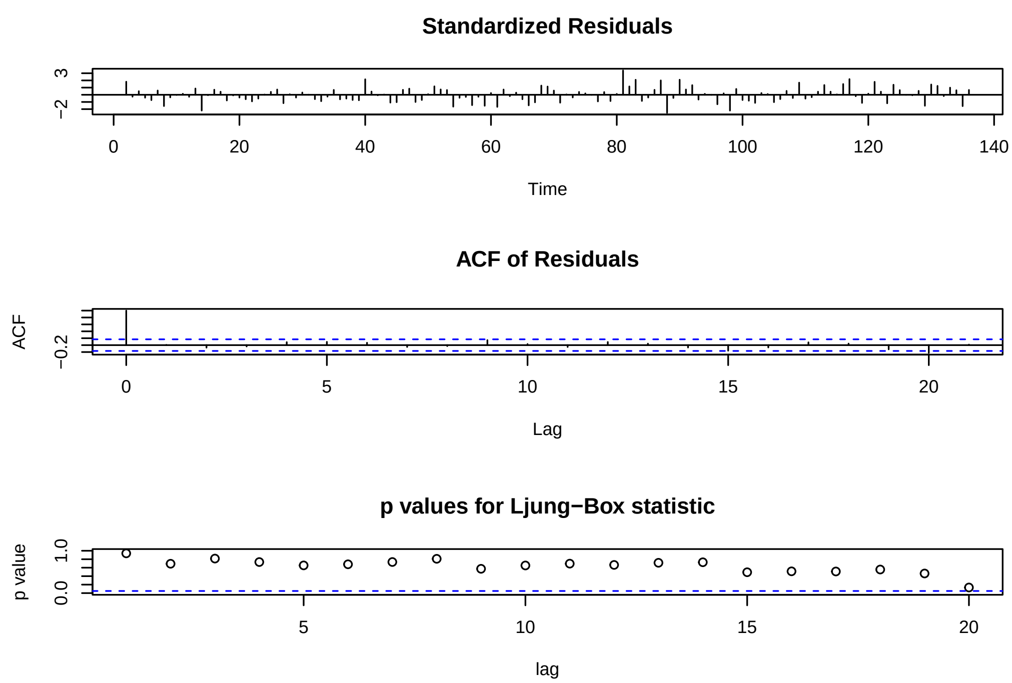

```
##
##  Box-Ljung test
##
## data:  rstandard(modelo.arma)
## X-squared = 16.819, df = 20, p-value = 0.6647
```

Petróleo Brent




```
##
##  Box-Ljung test
##
## data:  rstandard(modelo.arma)
## X-squared = 24.671, df = 20, p-value = 0.2143
```

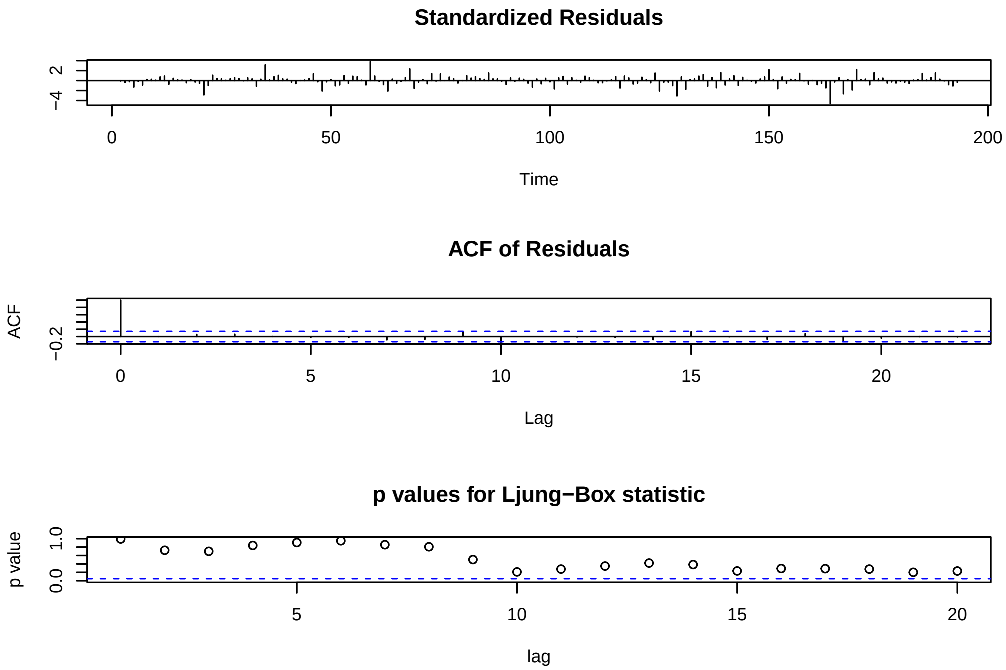
Dólar




```
##
##  Box-Ljung test
```

```
##
## data:  rstandard(modelo.arma)
## X-squared = 27.099, df = 20, p-value = 0.1325
```

Bitcoin



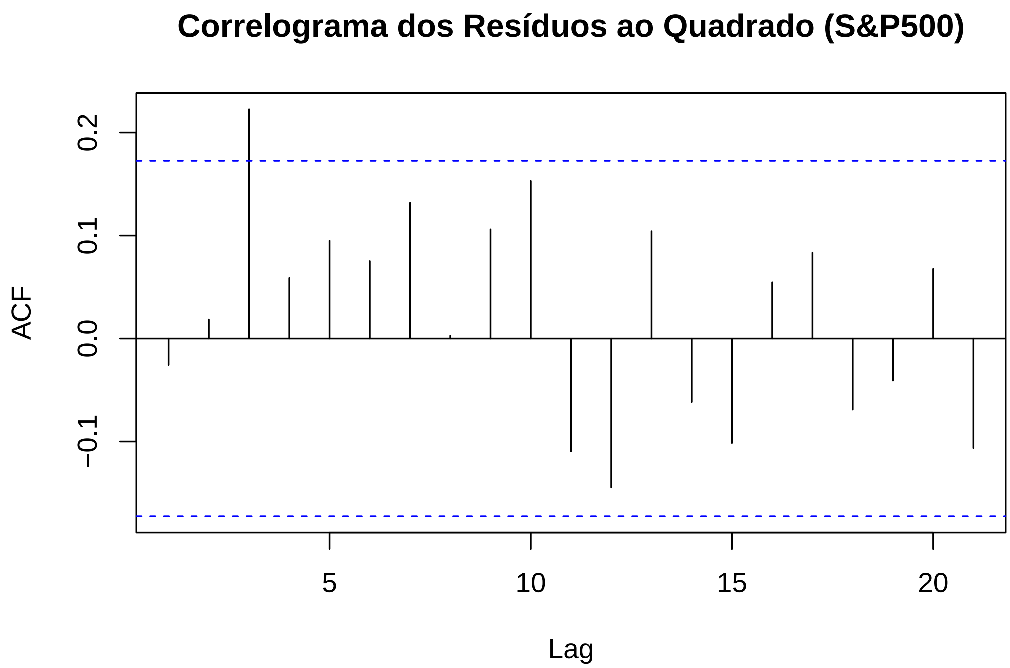

```
##
##  Box-Ljung test
##
## data:  rstandard(modelo.arma)
## X-squared = 24.257, df = 20, p-value = 0.2314
```

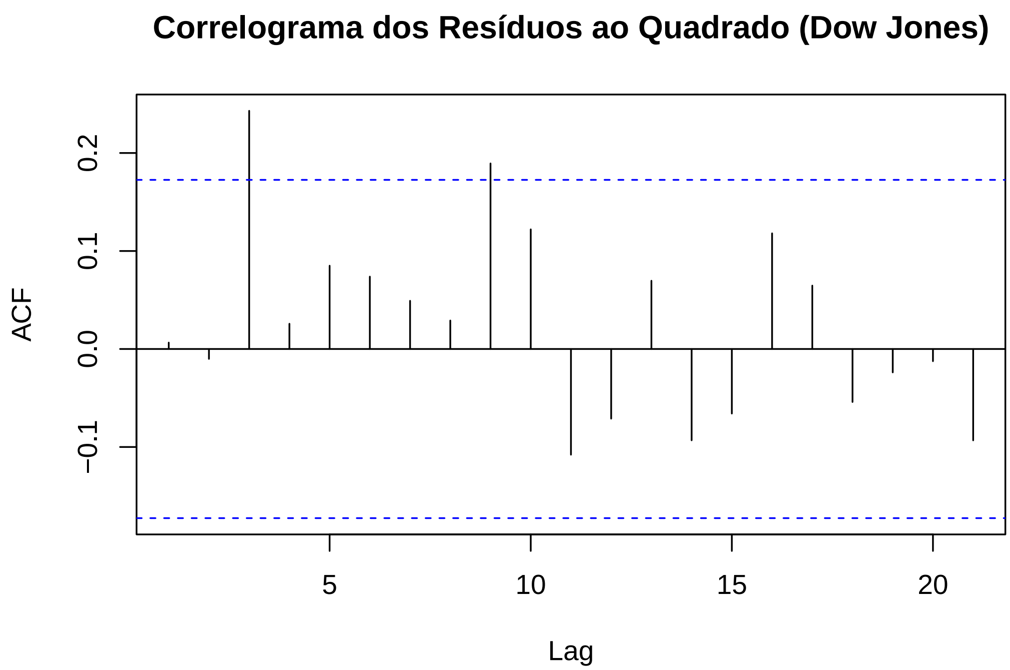
Portanto, analisando-se os gráficos gerados acima, constata-se que todos os modelos supercedem o requisito dos p-valores serem superiores a 5% e, portanto, são adequados para serem utilizados.

Ademais, temos que os resultados dos testes adicionais realizados nos dizem para não rejeitarmos a Hipótese Nula de que todas as funções de autocorrelação até 20 são zero, para todas as séries analisadas.

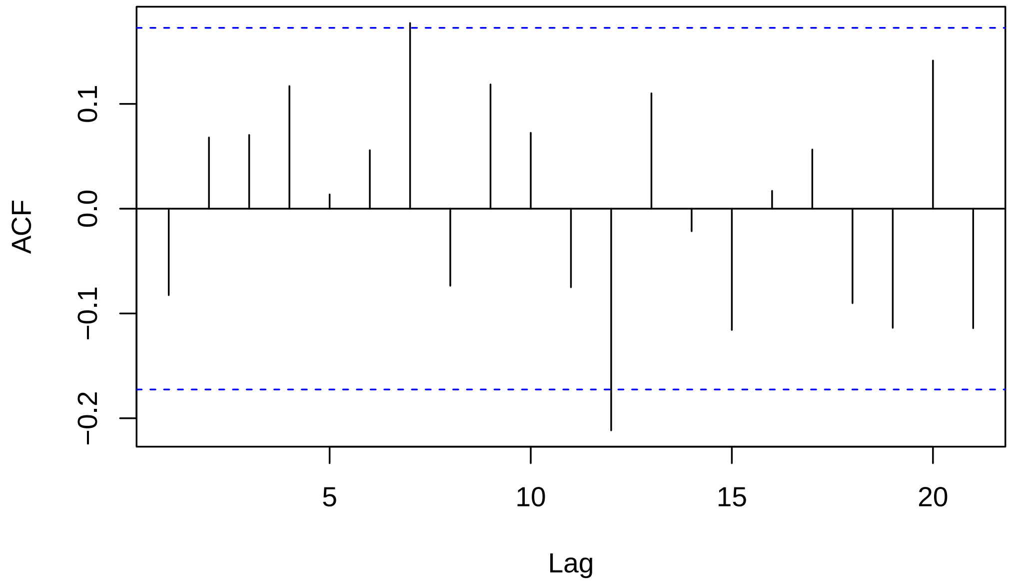
Correlograma dos Resíduos ao Quadrado

Portanto, temos o correlograma dos resíduos ao quadrado para cada uma das séries analisadas.

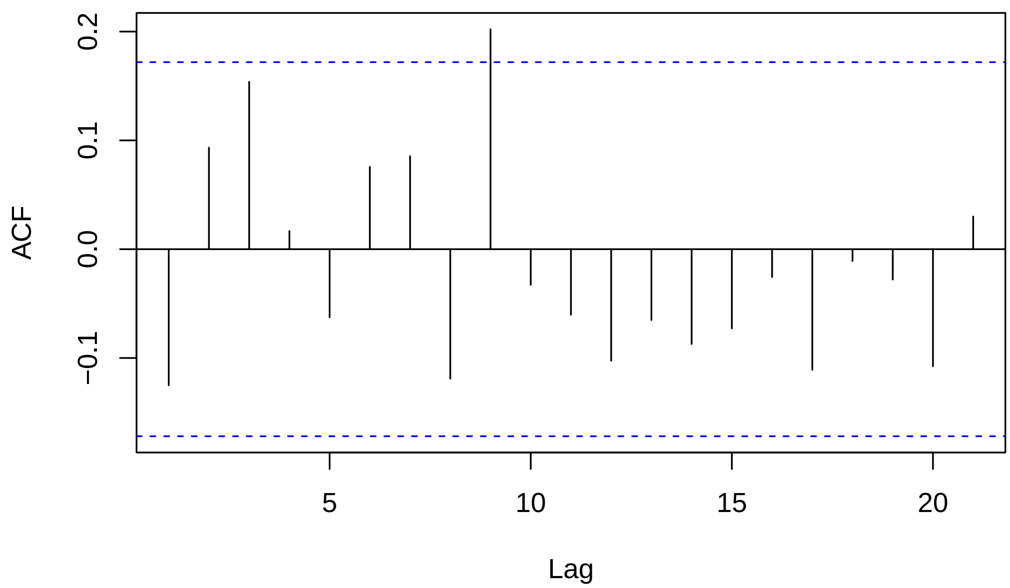

Correlograma dos Resíduos ao Quadrado (S&P500)
ACF
0.2
0.1
0.0
−0.1
5
10
15
20
Lag

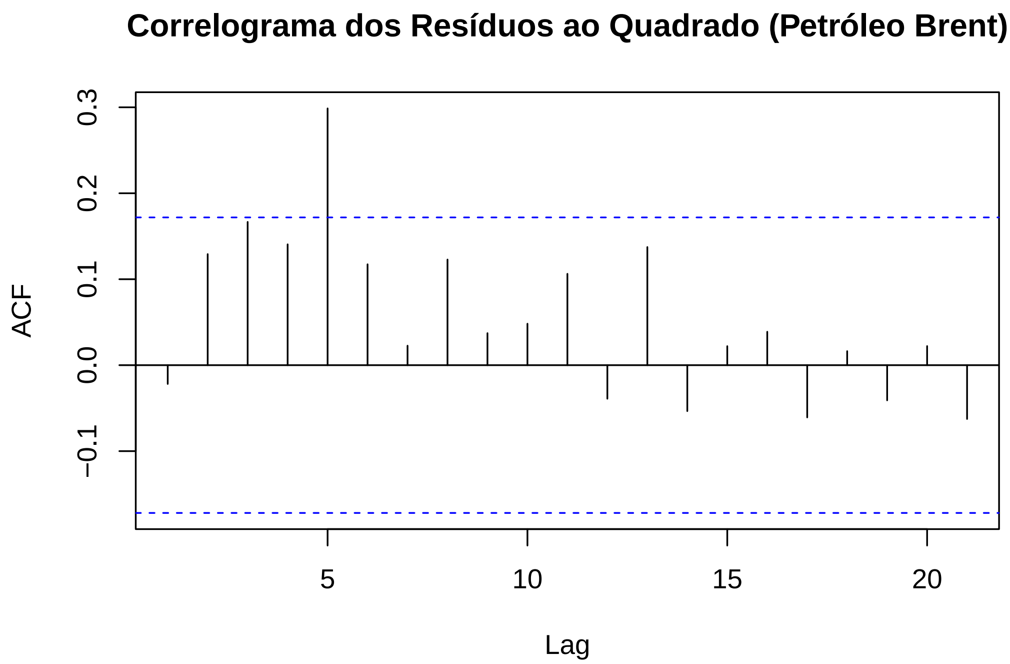

Correlograma dos Resíduos ao Quadrado (Dow Jones)
ACF
0.2
0.1
0.0
−0.1
5
10
15
20
Lag

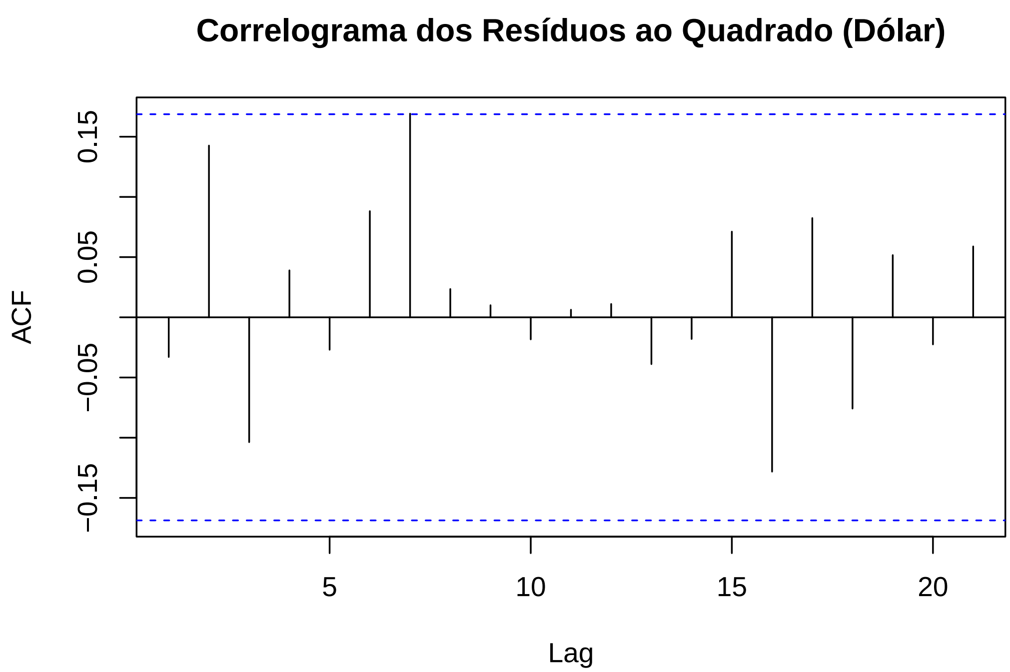
Correlograma dos Resíduos ao Quadrado (Nasdaq)

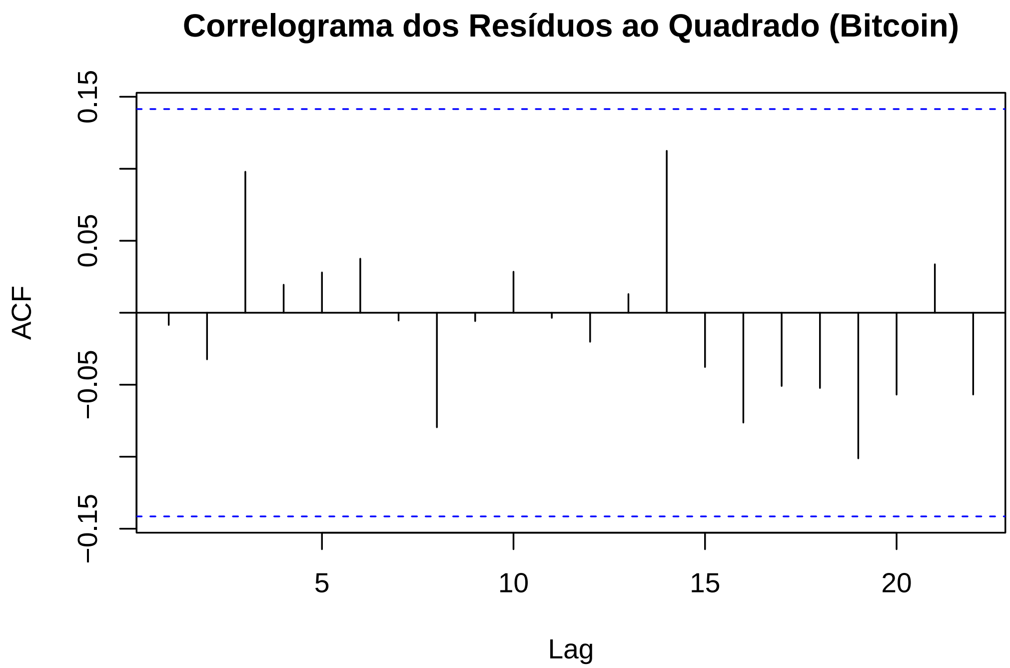


Correlograma dos Resíduos ao Quadrado (IBOVESPA)




Correlograma dos Resíduos ao Quadrado (Petróleo Brent)
ACF
−0.1
0.0
0.1
0.2
0.3
5
10
15
20
Lag


Correlograma dos Resíduos ao Quadrado (Dólar)
ACF
−0.15
−0.05
0.05
0.15
5
10
15
20
Lag



Portanto, podem ser utilizados os correlogramas gerados para auxiliar na escolha dos modelos ARCH/GARCH que será feita a seguir.

Teste de Efeitos ARCH e Escolha do Processo

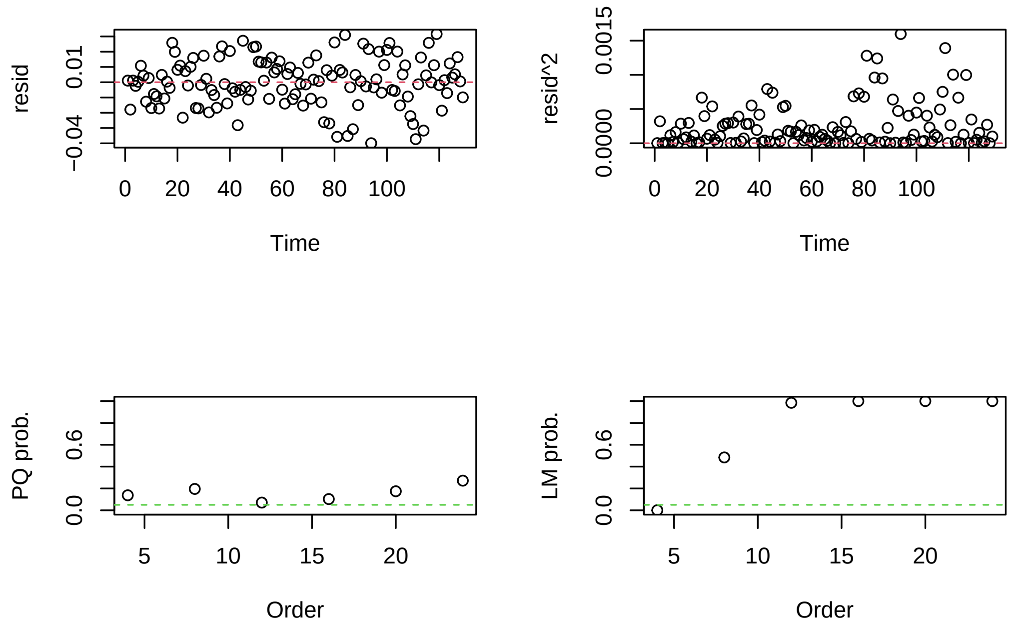
Para tal, usaremos as funções fornecidas nas bibliotecas do R. Temos os seguintes resultados para cada uma das séries temporais estudadas:

S&P500

```
##
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  sqresidual
## Chi-squared = 6.7513, df = 6, p-value = 0.3445
```

```
## ARCH heteroscedasticity test for residuals
## alternative: heteroscedastic
##
## Portmanteau-Q test:
##      order    PQ p.value
## [1,]     4  6.97  0.1375
## [2,]     8 11.11  0.1956
## [3,]    12 19.82  0.0705
## [4,]    16 23.43  0.1028
## [5,]    20 25.75  0.1743
## [6,]    24 27.73  0.2715
## Lagrange-Multiplier test:
##      order     LM  p.value
## [1,]     4 17.303 0.000612
## [2,]     8  6.487 0.484134
## [3,]    12  3.395 0.984402
## [4,]    16  1.606 0.999993
```

```
## [5,]    20  1.031 1.000000
## [6,]    24  0.634 1.000000
```




```
##
##  Box-Ljung test
##
## data:  y^2
## X-squared = 571.32, df = 6, p-value < 2.2e-16
## alternative hypothesis: y is heteroscedastic
```

Do primeiro teste, temos um p-valor superior a 5%, portanto, não rejeitamos que haja efeito ARCH. Do segundo teste, temos que há valores superiores a 5%, então, rejeitamos que haja heterocedasticidade no modelo. Do terceiro teste, temos que o p-valor é inferior a 5%, então não rejeitamos a hipótese de que a série seja heterocedástica.

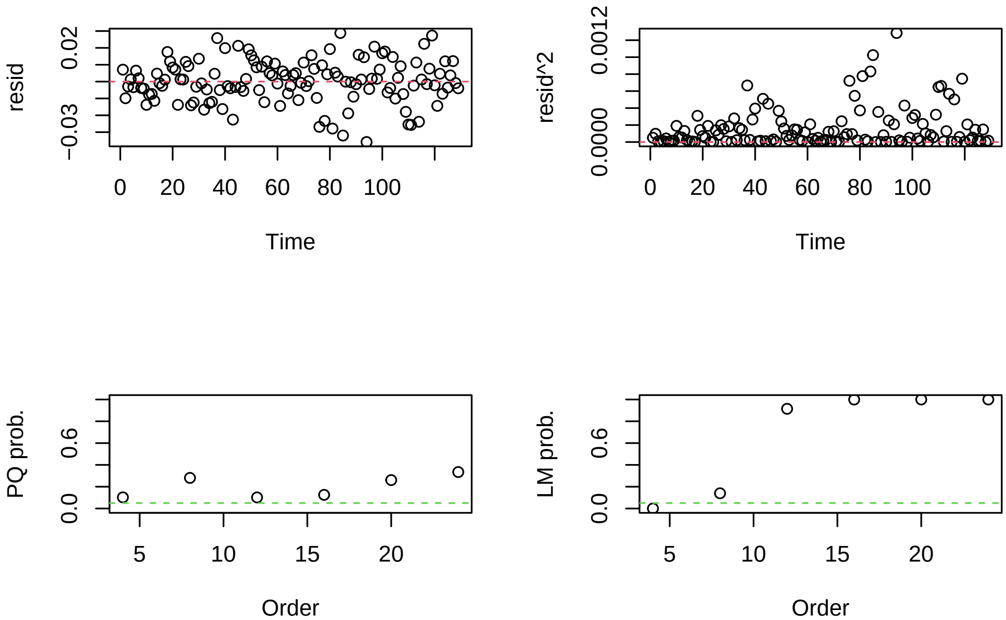
Assim, tomamos um modelo GARCH para representar a série.

Dow Jones

```
##
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  sqresidual
## Chi-squared = 3.6166, df = 6, p-value = 0.7284
```

```
## ARCH heteroscedasticity test for residuals
## alternative: heteroscedastic
##
## Portmanteau-Q test:
##      order    PQ p.value
## [1,]     4  7.72   0.102
## [2,]     8  9.78   0.281
## [3,]    12 18.48   0.102
## [4,]    16 22.58   0.125
## [5,]    20 23.59   0.261
## [6,]    24 26.38   0.334
```

```
## Lagrange-Multiplier test:
##      order    LM  p.value
## [1,]     4 26.29 8.29e-06
## [2,]     8 10.98 1.40e-01
## [3,]    12  5.32 9.15e-01
## [4,]    16  2.94 1.00e+00
## [5,]    20  1.74 1.00e+00
## [6,]    24  1.16 1.00e+00
```




```
##
##  Box-Ljung test
##
## data:  y^2
## X-squared = 545.94, df = 6, p-value < 2.2e-16
## alternative hypothesis: y is heteroscedastic
```

Do primeiro teste, temos um p-valor superior a 5%, portanto, não rejeitamos que haja efeito ARCH. Do segundo teste, temos que há valores superiores a 5%, então, rejeitamos que haja heterocedasticidade no modelo. Do terceiro teste, temos que o p-valor é inferior a 5%, então não rejeitamos a hipótese de que a série seja heterocedástica.

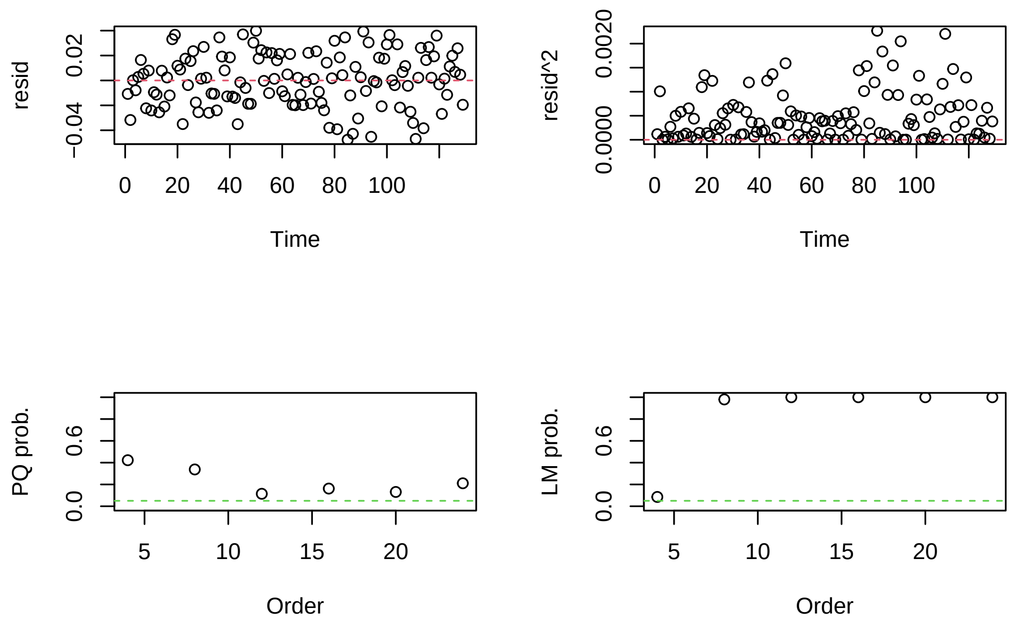
Assim, tomamos um modelo GARCH para representar a série.

Nasdaq

```
##
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  sqresidual
## Chi-squared = 4.7651, df = 6, p-value = 0.5743
```

```
## ARCH heteroscedasticity test for residuals
## alternative: heteroscedastic
##
## Portmanteau-Q test:
##      order    PQ p.value
```

```
## [1,]     4  3.88   0.422
## [2,]     8  9.06   0.338
## [3,]    12 18.05   0.114
## [4,]    16 21.44   0.162
## [5,]    20 27.15   0.131
## [6,]    24 29.25   0.211
## Lagrange-Multiplier test:
##      order      LM p.value
## [1,]     4  6.6124  0.0853
## [2,]     8  1.5667  0.9799
## [3,]    12  0.6612  1.0000
## [4,]    16 -0.0212  1.0000
## [5,]    20 -0.0856  1.0000
## [6,]    24 -0.1311  1.0000
```




```
##
##  Box-Ljung test
##
## data:  y^2
## X-squared = 584.14, df = 6, p-value < 2.2e-16
## alternative hypothesis: y is heteroscedastic
```

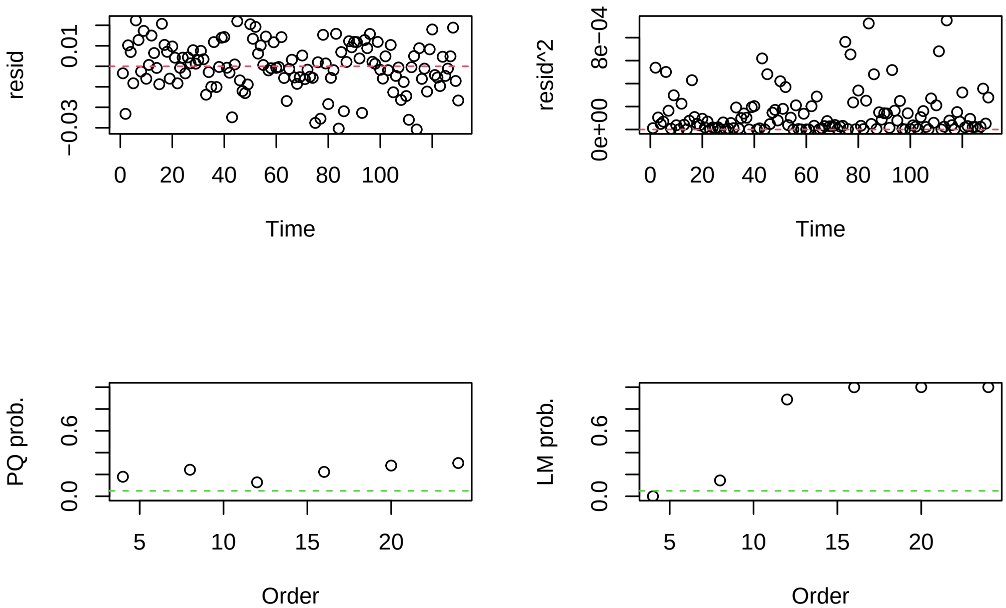
Do primeiro teste, temos um p-valor superior a 5%, portanto, não rejeitamos que haja efeito ARCH. Do segundo teste, temos que há valores superiores a 5%, então, rejeitamos que haja heterocedasticidade no modelo. Do terceiro teste, temos que o p-valor é inferior a 5%, então não rejeitamos a hipótese de que a série seja heterocedástica.

Assim, tomamos um modelo ARCH para representar a série.

IBOVESPA

```
##
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  sqresidual
## Chi-squared = 6.215, df = 6, p-value = 0.3995
```

```
## ARCH heteroscedasticity test for residuals
## alternative: heteroscedastic
##
## Portmanteau-Q test:
##      order    PQ p.value
## [1,]     4  6.28   0.179
## [2,]     8 10.33   0.243
## [3,]    12 17.62   0.128
## [4,]    16 19.93   0.223
## [5,]    20 23.15   0.281
## [6,]    24 27.00   0.305
## Lagrange-Multiplier test:
##      order    LM  p.value
## [1,]     4 25.40 1.27e-05
## [2,]     8 10.85 1.45e-01
## [3,]    12  5.79 8.87e-01
## [4,]    16  3.33 9.99e-01
## [5,]    20  2.39 1.00e+00
## [6,]    24  1.37 1.00e+00
```




```
##
##  Box-Ljung test
##
## data:  y^2
## X-squared = 588.25, df = 6, p-value < 2.2e-16
## alternative hypothesis: y is heteroscedastic
```

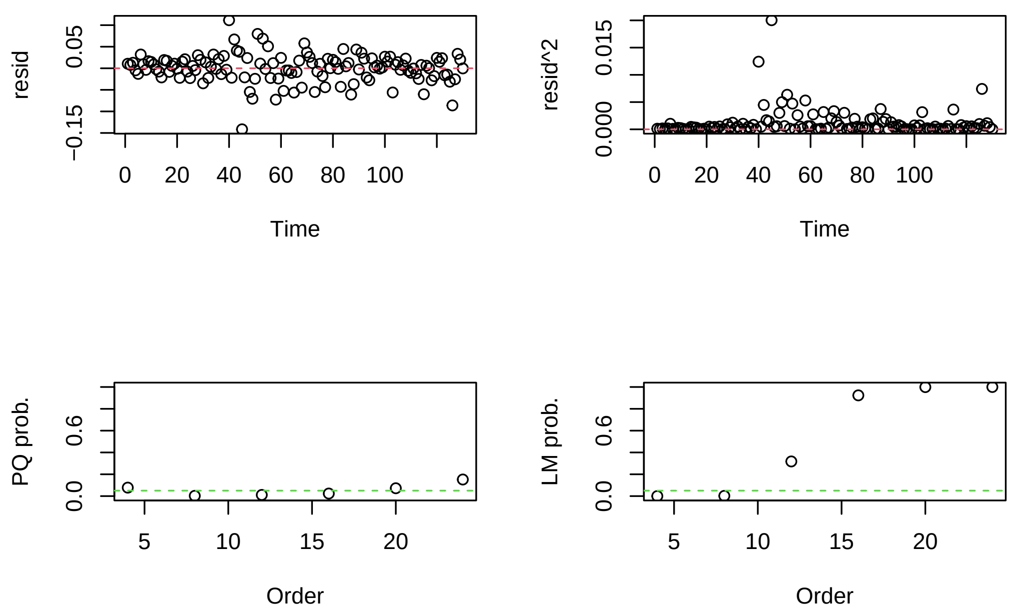
Do primeiro teste, temos um p-valor superior a 5%, portanto, não rejeitamos que haja efeito ARCH. Do segundo teste, temos que há valores superiores a 5%, então, rejeitamos que haja heterocedasticidade no modelo. Do terceiro teste, temos que o p-valor é inferior a 5%, então não rejeitamos a hipótese de que a série seja heterocedástica.

Assim, tomamos um modelo ARCH para representar a série.

Petróleo Brent

```
##
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  sqresidual
## Chi-squared = 13.105, df = 6, p-value = 0.0414
```

```
## ARCH heteroscedasticity test for residuals
## alternative: heteroscedastic
##
## Portmanteau-Q test:
##      order    PQ p.value
## [1,]     4  8.41 0.07754
## [2,]     8 23.84 0.00244
## [3,]    12 25.99 0.01078
## [4,]    16 29.07 0.02344
## [5,]    20 29.87 0.07197
## [6,]    24 31.06 0.15211
## Lagrange-Multiplier test:
##      order    LM p.value
## [1,]     4 85.33 0.00000
## [2,]     8 22.57 0.00202
## [3,]    12 12.64 0.31771
## [4,]    16  8.01 0.92337
## [5,]    20  5.38 0.99904
## [6,]    24  3.93 1.00000
```



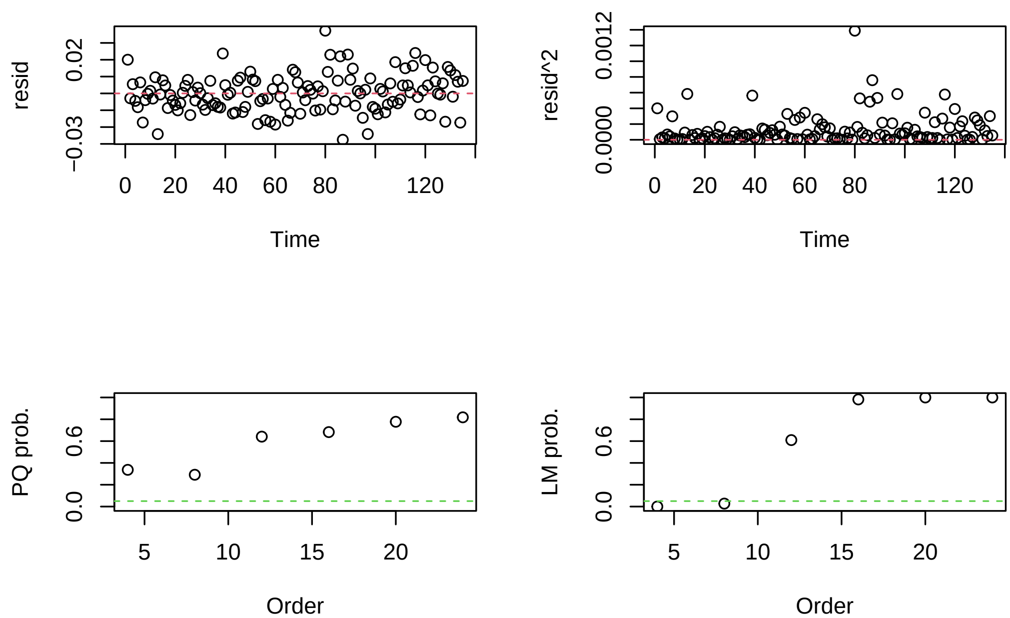

```
##
##  Box-Ljung test
##
## data:  y^2
## X-squared = 499.15, df = 6, p-value < 2.2e-16
## alternative hypothesis: y is heteroscedastic
```

Do primeiro teste, temos um p-valor inferior a 5%, portanto, rejeitamos que haja efeito ARCH. Do segundo teste, nota-se que há p-valores inferioresa 5% apenas para os casos de ordem 8, 12 e 16. Do terceiro teste, temos que o p-valor é inferior a 5%, então não rejeitamos a hipótese de que a série seja heterocedástica. Assim, tomamos um modelo ARCH para representar a série.

Dólar

```
##
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  sqresidual
## Chi-squared = 2.2599, df = 6, p-value = 0.8943
```

```
## ARCH heteroscedasticity test for residuals
## alternative: heteroscedastic
##
## Portmanteau-Q test:
##      order    PQ p.value
## [1,]     4  4.55   0.337
## [2,]     8  9.64   0.292
## [3,]    12  9.72   0.641
## [4,]    16 12.87   0.683
## [5,]    20 14.99   0.777
## [6,]    24 17.68   0.818
## Lagrange-Multiplier test:
##      order    LM  p.value
## [1,]     4 38.48 2.24e-08
## [2,]     8 15.80 2.70e-02
## [3,]    12  9.14 6.09e-01
## [4,]    16  5.86 9.82e-01
## [5,]    20  3.80 1.00e+00
## [6,]    24  2.54 1.00e+00
```




```
##
##  Box-Ljung test
##
## data:  y^2
```

```
## X-squared = 624.66, df = 6, p-value < 2.2e-16
## alternative hypothesis: y is heteroscedastic
```

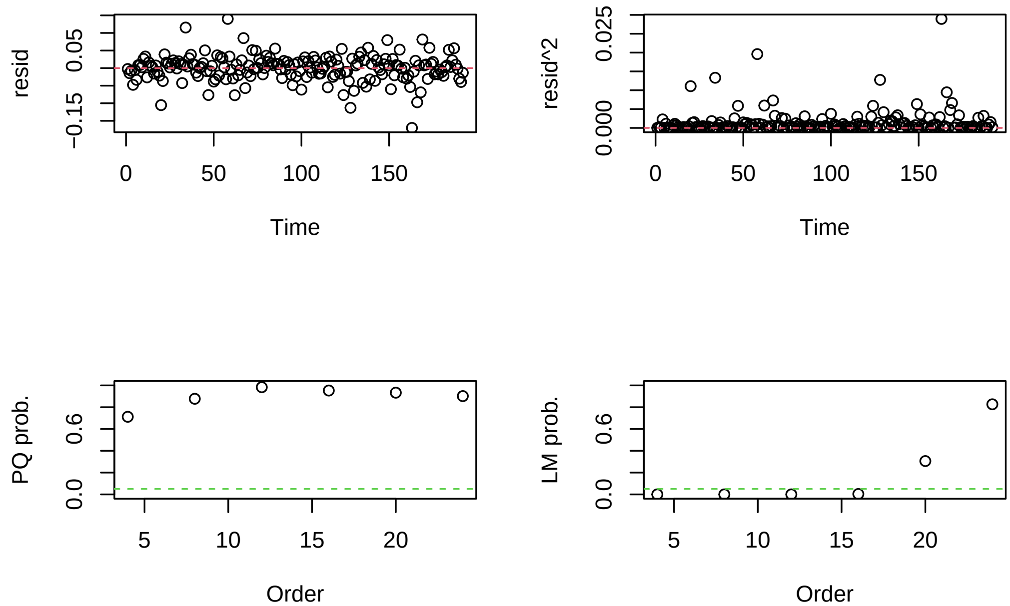
Do primeiro teste, temos um p-valor superior a 5%, portanto, não rejeitamos que haja efeito ARCH. Do segundo teste, temos que há valores superiores a 5%, então, rejeitamos que haja heterocedasticidade no modelo. Do terceiro teste, temos que o p-valor é inferior a 5%, então não rejeitamos a hipótese de que a série seja heterocedástica.

Assim, tomamos um modelo GARCH para representar a série.

Bitcoin

```
##
##  ARCH LM-test; Null hypothesis: no ARCH effects
##
## data:  sqresidual
## Chi-squared = 0.79725, df = 6, p-value = 0.9921
```

```
## ARCH heteroscedasticity test for residuals
## alternative: heteroscedastic
##
## Portmanteau-Q test:
##      order    PQ p.value
## [1,]     4  2.13   0.712
## [2,]     8  3.77   0.877
## [3,]    12  4.01   0.983
## [4,]    16  7.86   0.953
## [5,]    20 11.47   0.933
## [6,]    24 15.61   0.902
## Lagrange-Multiplier test:
##      order    LM  p.value
## [1,]     4 186.4 0.00e+00
## [2,]     8  88.0 3.33e-16
## [3,]    12  52.7 2.03e-07
## [4,]    16  33.8 3.65e-03
## [5,]    20  21.6 3.06e-01
## [6,]    24  16.6 8.27e-01
```




```
##
##  Box-Ljung test
##
## data:  y^2
## X-squared = 955.53, df = 6, p-value < 2.2e-16
## alternative hypothesis: y is heteroscedastic
```

Do primeiro teste, temos um p-valor superior a 5%, portanto, não rejeitamos que haja efeito ARCH. Do segundo teste, temos que há valores superiores a 5%, então, rejeitamos que haja heterocedasticidade no modelo. Do terceiro teste, temos que o p-valor é inferior a 5%, então não rejeitamos a hipótese de que a série seja heterocedástica.

Assim, tomamos um modelo ARCH para representar a série.

Estimação do Modelo ARCH/GARCH

De acordo com o que foi selecionado na seção anterior, temos os seguintes modelos estimados a partir das informações coletadas até então. Nota-se que todos os coeficientes escolhidos são significativos para o modelo.

S&P500

```
##
## Series Initialization:
##  ARMA Model:                arma
##  Formula Mean:              ~ arma(0, 0)
##  GARCH Model:               garch
##  Formula Variance:          ~ garch(1, 1)
##  ARMA Order:                0 0
##  Max ARMA Order:            0
##  GARCH Order:               1 1
##  Max GARCH Order:           1
##  Maximum Order:             1
##  Conditional Dist:          norm
##  h.start:                   2
##  llh.start:                 1
```

```
##  Length of Series:           130
##  Recursion Init:             mci
##  Series Scale:               281.7138
##
## Parameter Initialization:
##  Initial Parameters:           $params
##  Limits of Transformations:   $U, $V
##  Which Parameters are Fixed?  $includes
##  Parameter Matrix:
##                       U         V   params includes
##     mu     -151.39987478 151.3999 15.13999     TRUE
##     omega     0.00000100 100.0000  0.10000     TRUE
##     alpha1    0.00000001   1.0000  0.10000     TRUE
##     gamma1   -0.99999999   1.0000  0.10000    FALSE
##     beta1     0.00000001   1.0000  0.80000     TRUE
##     delta     0.00000000   2.0000  2.00000    FALSE
##     skew      0.10000000  10.0000  1.00000    FALSE
##     shape     1.00000000  10.0000  4.00000    FALSE
##  Index List of Parameters to be Optimized:
##     mu  omega alpha1  beta1
##      1      2      3      5
##  Persistence:                  0.9
##
##
## --- START OF TRACE ---
## Selected Algorithm: nlminb
##
## R coded nlminb Solver:
##
##   0:     169.92302:  15.1400 0.100000 0.100000 0.800000
##   1:     165.01499:  15.5926 0.0905762 0.100693 0.793076
##   2:     160.74947:  15.3667 0.0660338 0.109720 0.781921
##   3:     155.71179:  15.6209 0.0436858 0.124437 0.776171
##   4:     152.78916:  15.4860 0.0201081 0.144326 0.776496
##   5:     150.70905:  15.7101 0.0167092 0.172123 0.781755
##   6:     149.92751:  15.5156 0.0142042 0.201293 0.778797
##   7:     149.19110:  15.6082 0.0121688 0.201463 0.777347
##   8:     149.00225:  15.6539 0.0109124 0.203464 0.771967
##   9:     148.80893:  15.6194 0.0132326 0.206161 0.766887
##  10:     148.63450:  15.6571 0.0119287 0.208776 0.761510
##  11:     148.44879:  15.6189 0.0136125 0.212112 0.756680
##  12:     148.29730:  15.6642 0.0141160 0.215251 0.751720
##  13:     148.11967:  15.6310 0.0120828 0.218761 0.746986
##  14:     147.93431:  15.6400 0.0153866 0.222345 0.742566
##  15:     147.76159:  15.6457 0.0128171 0.225826 0.737586
##  16:     147.59620:  15.6507 0.0157603 0.229912 0.733321
##  17:     147.44312:  15.6268 0.0138640 0.233321 0.728228
##  18:     147.26235:  15.6672 0.0148172 0.237355 0.723828
##  19:     147.08505:  15.6363 0.0151624 0.241273 0.718927
##  20:     143.46623:  15.7308 0.0152493 0.359421 0.576055
##  21:     141.99625:  15.6594 0.0524271 0.463718 0.427300
##  22:     141.55574:  15.5871 0.0453002 0.464133 0.426064
##  23:     141.04149:  15.6672 0.0417262 0.469978 0.425476
##  24:     140.72914:  15.6149 0.0355220 0.474947 0.425119
```

```
##  25:     140.41485:  15.6775 0.0330341 0.482108 0.425940
##  26:     140.24088:  15.6445 0.0294198 0.489655 0.426626
##  27:     140.08943:  15.6829 0.0277246 0.497773 0.426868
##  28:     139.97073:  15.6624 0.0259600 0.505929 0.424921
##  29:     138.13741:  15.7097 0.0220184 0.661351 0.261587
##  30:     137.77885:  15.7067 0.0260066 0.749137 0.264597
##  31:     137.42456:  15.6725 0.0389830 0.759769 0.178313
##  32:     136.94687:  15.6925 0.0370334 0.830029 0.125500
##  33:     136.74135:  15.7039 0.0371639 0.893384 0.0937262
##  34:     136.56765:  15.7344 0.0330781 0.914662 0.0647693
##  35:     136.47162:  15.7391 0.0314967 0.998113 0.0199112
##  36:     136.47087:  15.7463 0.0317560  1.00000 0.0356148
##  37:     136.45914:  15.7406 0.0314311  1.00000 0.0304421
##  38:     136.45910:  15.7412 0.0312881  1.00000 0.0302693
##  39:     136.45909:  15.7411 0.0313150  1.00000 0.0303327
##  40:     136.45909:  15.7411 0.0313148  1.00000 0.0303331
##
## Final Estimate of the Negative LLH:
##  LLH:  869.775    norm LLH:  6.690577
##           mu        omega       alpha1        beta1
## 4.434475e+03 2.485226e+03 1.000000e+00 3.033312e-02
##
## R-optimhess Difference Approximated Hessian Matrix:
##                   mu         omega        alpha1         beta1
## mu     -3.038520e-03 -2.613396e-05  -0.015793076 -2.870430e-01
## omega  -2.613396e-05 -1.104215e-06  -0.002550427 -8.776022e-03
## alpha1 -1.579308e-02 -2.550427e-03 -45.728118451 -6.028913e+01
## beta1  -2.870430e-01 -8.776022e-03 -60.289134780 -1.880734e+02
## attr(,"time")
## Time difference of 0.005948782 secs
##
## --- END OF TRACE ---
##
##
## Time to Estimate Parameters:
##  Time difference of 0.03409791 secs

## Warning: Using formula(x) is deprecated when x is a character vector of length > 1.
##   Consider formula(paste(x, collapse = " ")) instead.

##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~garch(1, 1), data = stdpoors.ts, trace = T)
##
## Mean and Variance Equation:
##  data ~ garch(1, 1)
## <environment: 0x5630506cbbc8>
##  [data = stdpoors.ts]
##
## Conditional Distribution:
##  norm
##
```

```
## Coefficient(s):
##         mu       omega      alpha1       beta1
## 4.4345e+03  2.4852e+03  1.0000e+00  3.0333e-02
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu     4.434e+03   2.145e+01  206.713  < 2e-16 ***
## omega  2.485e+03   1.258e+03    1.975   0.0482 *
## alpha1 1.000e+00   2.035e-01    4.914 8.92e-07 ***
## beta1  3.033e-02   1.182e-01    0.257   0.7974
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  -869.775    normalized:  -6.690577
##
## Description:
##  Tue Jul 12 16:49:40 2022 by user: stevenson
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  9.389576  0.009142805
##  Shapiro-Wilk Test  R    W      0.9073156 1.927483e-07
##  Ljung-Box Test     R    Q(10)  363.5365  0
##  Ljung-Box Test     R    Q(15)  370.3866  0
##  Ljung-Box Test     R    Q(20)  371.0373  0
##  Ljung-Box Test     R^2  Q(10)  15.24529  0.1233716
##  Ljung-Box Test     R^2  Q(15)  21.3739   0.1253024
##  Ljung-Box Test     R^2  Q(20)  24.99033  0.2018012
##  LM Arch Test       R    TR^2   14.74789  0.255514
##
## Information Criterion Statistics:
##      AIC      BIC      SIC     HQIC
## 13.44269 13.53092 13.44087 13.47854
```

Dow Jones

```
##
## Series Initialization:
##  ARMA Model:                arma
##  Formula Mean:              ~ arma(0, 0)
##  GARCH Model:               garch
##  Formula Variance:          ~ garch(1, 1)
##  ARMA Order:                0 0
##  Max ARMA Order:            0
##  GARCH Order:               1 1
##  Max GARCH Order:           1
##  Maximum Order:             1
##  Conditional Dist:          norm
##  h.start:                   2
```

```
##  llh.start:                 1
##  Length of Series:          130
##  Recursion Init:            mci
##  Series Scale:              1657.502
##
## Parameter Initialization:
##  Initial Parameters:          $params
##  Limits of Transformations:   $U, $V
##  Which Parameters are Fixed?  $includes
##  Parameter Matrix:
##                       U        V   params includes
##     mu     -202.51865692 202.5187 20.25187     TRUE
##     omega     0.00000100 100.0000  0.10000     TRUE
##     alpha1    0.00000001   1.0000  0.10000     TRUE
##     gamma1   -0.99999999   1.0000  0.10000    FALSE
##     beta1     0.00000001   1.0000  0.80000     TRUE
##     delta     0.00000000   2.0000  2.00000    FALSE
##     skew      0.10000000  10.0000  1.00000    FALSE
##     shape     1.00000000  10.0000  4.00000    FALSE
##  Index List of Parameters to be Optimized:
##     mu  omega alpha1  beta1
##      1      2      3      5
##  Persistence:                  0.9
##
##
## --- START OF TRACE ---
## Selected Algorithm: nlminb
##
## R coded nlminb Solver:
##
##   0:     168.69308:  20.2519 0.100000 0.100000 0.800000
##   1:     165.14826:  20.5260 0.0961688 0.100215 0.797199
##   2:     158.98158:  20.4975 0.0469705 0.111274 0.769843
##   3:     157.72829:  20.6446 0.0465757 0.114894 0.770448
##   4:     156.29630:  20.5374 0.0377502 0.131046 0.771739
##   5:     154.11389:  20.7699 0.0291718 0.143811 0.771446
##   6:     152.50908:  20.6898 0.0165060 0.157681 0.771311
##   7:     151.19851:  20.8346 0.0112696 0.173775 0.765753
##   8:     150.17183:  20.7539 0.0151653 0.185990 0.752035
##   9:     147.44881:  20.8642 0.0183629 0.244032 0.692033
##  10:     146.12846:  20.7920 0.00936855 0.262848 0.680739
##  11:     145.80383:  20.8603 0.00812615 0.269061 0.676967
##  12:     145.45861:  20.8257 0.00786607 0.274494 0.671202
##  13:     145.21297:  20.8429 0.0118898 0.276312 0.664456
##  14:     144.95970:  20.8190 0.0109285 0.282119 0.659008
##  15:     144.09845:  20.8616 0.00630392 0.314942 0.627479
##  16:     142.80121:  20.8136 0.0104342 0.338184 0.588309
##  17:     142.07445:  20.8739 0.0146593 0.360123 0.548445
##  18:     137.73874:  20.8610 0.0139648 0.525751 0.341408
##  19:     136.07973:  20.8166 0.00923350 0.790631 0.351857
##  20:     134.17211:  20.8320 0.0155993 0.939519 0.132564
##  21:     134.05887:  20.8611 0.0156667 0.939527 0.132588
##  22:     133.99782:  20.8503 0.0155139 0.939237 0.133882
##  23:     133.95400:  20.8510 0.0123801 0.938415 0.154546
```

```
##  24:     133.92319:  20.8507 0.0126582 0.926627 0.171822
##  25:     133.92104:  20.8509 0.0128484 0.932285 0.169257
##  26:     133.91993:  20.8509 0.0128997 0.940186 0.167032
##  27:     133.91992:  20.8509 0.0128795 0.940878 0.166909
##  28:     133.91992:  20.8509 0.0128750 0.940906 0.166916
##  29:     133.91992:  20.8509 0.0128748 0.940902 0.166918
##
## Final Estimate of the Negative LLH:
##  LLH:  1097.619    norm LLH:  8.44322
##           mu        omega       alpha1        beta1
## 3.456043e+04 3.537102e+04 9.409021e-01 1.669177e-01
##
## R-optimhess Difference Approximated Hessian Matrix:
##                   mu         omega        alpha1         beta1
## mu     -3.897995e-04 -1.647294e-07 -4.639142e-03 -2.807536e-02
## omega  -1.647294e-07 -2.635658e-09 -1.637251e-04 -5.258003e-04
## alpha1 -4.639142e-03 -1.637251e-04 -5.728208e+01 -8.491416e+01
## beta1  -2.807536e-02 -5.258003e-04 -8.491416e+01 -2.450046e+02
## attr(,"time")
## Time difference of 0.005023241 secs
##
## --- END OF TRACE ---
##
##
## Time to Estimate Parameters:
##  Time difference of 0.03167033 secs

## Warning: Using formula(x) is deprecated when x is a character vector of length > 1.
##   Consider formula(paste(x, collapse = " ")) instead.

##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~garch(1, 1), data = dowjones.ts, trace = T)
##
## Mean and Variance Equation:
##  data ~ garch(1, 1)
## <environment: 0x56305209ba38>
##  [data = dowjones.ts]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##         mu       omega      alpha1       beta1
## 3.4560e+04  3.5371e+04  9.4090e-01  1.6692e-01
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu     3.456e+04   5.138e+01  672.656  < 2e-16 ***
```

```
## omega  3.537e+04   2.610e+04    1.355    0.175
## alpha1 9.409e-01   1.904e-01    4.942 7.74e-07 ***
## beta1  1.669e-01   1.103e-01    1.513    0.130
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  -1097.619    normalized:  -8.44322
##
## Description:
##  Tue Jul 12 16:49:40 2022 by user: stevenson
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  10.05235  0.006563859
##  Shapiro-Wilk Test  R    W      0.9142264 4.752138e-07
##  Ljung-Box Test     R    Q(10)  264.365   0
##  Ljung-Box Test     R    Q(15)  269.913   0
##  Ljung-Box Test     R    Q(20)  272.8326  0
##  Ljung-Box Test     R^2  Q(10)  25.85465  0.003940676
##  Ljung-Box Test     R^2  Q(15)  29.81394  0.0126104
##  Ljung-Box Test     R^2  Q(20)  30.81116  0.05773336
##  LM Arch Test       R    TR^2   22.03726  0.03710437
##
## Information Criterion Statistics:
##      AIC      BIC      SIC     HQIC
## 16.94798 17.03621 16.94616 16.98383
```

Nasdaq

```
##
## Series Initialization:
##  ARMA Model:                arma
##  Formula Mean:              ~ arma(0, 0)
##  GARCH Model:               garch
##  Formula Variance:          ~ garch(1, 0)
##  ARMA Order:                0 0
##  Max ARMA Order:            0
##  GARCH Order:               1 0
##  Max GARCH Order:           1
##  Maximum Order:             1
##  Conditional Dist:          norm
##  h.start:                   2
##  llh.start:                 1
##  Length of Series:          130
##  Recursion Init:            mci
##  Series Scale:              1270.244
##
## Parameter Initialization:
##  Initial Parameters:          $params
##  Limits of Transformations:   $U, $V
##  Which Parameters are Fixed?  $includes
##  Parameter Matrix:
```

```
##                        U        V   params includes
##     mu     -102.49908682 102.4991 10.24991     TRUE
##     omega     0.00000100 100.0000  0.10000     TRUE
##     alpha1    0.00000001   1.0000  0.10000     TRUE
##     gamma1   -0.99999999   1.0000  0.10000    FALSE
##     delta     0.00000000   2.0000  2.00000    FALSE
##     skew      0.10000000  10.0000  1.00000    FALSE
##     shape     1.00000000  10.0000  4.00000    FALSE
##  Index List of Parameters to be Optimized:
##     mu  omega alpha1
##      1      2      3
##  Persistence:                 0.1
##
##
## --- START OF TRACE ---
## Selected Algorithm: nlminb
##
## R coded nlminb Solver:
##
##   0:     278.99085:  10.2499 0.100000 0.100000
##   1:     274.61367:  12.2482 0.375607 0.431645
##   2:     175.06119:  9.83800 0.380651 0.457558
##   3:     148.99792:  10.6971 0.200786 0.940359
##   4:     145.51305:  10.6558 0.150362 0.936498
##   5:     138.29597:  10.6546 0.0494762 0.947328
##   6:     137.57172:  10.7642 0.0470353 0.947434
##   7:     137.25381:  10.7248 0.0368294 0.948624
##   8:     137.25064:  10.7201 0.0369655 0.948755
##   9:     137.24799:  10.7245 0.0373319 0.950097
##  10:     137.24210:  10.7194 0.0371465 0.951453
##  11:     137.22271:  10.7202 0.0354107 0.959830
##  12:     137.19620:  10.7158 0.0364266 0.968316
##  13:     137.18468:  10.7260 0.0364211 0.976814
##  14:     137.16081:  10.7174 0.0362029 0.985326
##  15:     137.15024:  10.7156 0.0344111 0.993691
##  16:     137.14129:  10.7147 0.0359654  1.00000
##  17:     137.13923:  10.7156 0.0352013  1.00000
##  18:     137.13921:  10.7150 0.0351548  1.00000
##  19:     137.13921:  10.7151 0.0351693  1.00000
##  20:     137.13921:  10.7151 0.0351692  1.00000
##
## Final Estimate of the Negative LLH:
##  LLH:  1066.245    norm LLH:  8.201881
##       mu    omega   alpha1
## 13610.81 56746.10     1.00
##
## R-optimhess Difference Approximated Hessian Matrix:
##                   mu         omega        alpha1
## mu     -1.743740e-04  1.868862e-07 -1.158453e-02
## omega   1.868862e-07 -2.382469e-09 -1.173147e-04
## alpha1 -1.158453e-02 -1.173147e-04 -4.633373e+01
## attr(,"time")
## Time difference of 0.002571106 secs
##
```

```
## --- END OF TRACE ---
##
##
## Time to Estimate Parameters:
##  Time difference of 0.01604986 secs

## Warning: Using formula(x) is deprecated when x is a character vector of length > 1.
##   Consider formula(paste(x, collapse = " ")) instead.

##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~garch(1, 0), data = nasdaq.ts, trace = T)
##
## Mean and Variance Equation:
##  data ~ garch(1, 0)
## <environment: 0x5630575f44a8>
##  [data = nasdaq.ts]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##     mu   omega  alpha1
##  13611   56746       1
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu     1.361e+04   8.191e+01  166.175  < 2e-16 ***
## omega  5.675e+04   2.349e+04    2.416   0.0157 *
## alpha1 1.000e+00   1.625e-01    6.152 7.63e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  -1066.245    normalized:  -8.201881
##
## Description:
##  Tue Jul 12 16:49:41 2022 by user: stevenson
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  10.30584  0.005782489
##  Shapiro-Wilk Test  R    W      0.9150084 5.276778e-07
##  Ljung-Box Test     R    Q(10)  451.1057  0
##  Ljung-Box Test     R    Q(15)  494.0965  0
##  Ljung-Box Test     R    Q(20)  509.7311  0
##  Ljung-Box Test     R^2  Q(10)  8.790383  0.552106
##  Ljung-Box Test     R^2  Q(15)  15.8023   0.3953116
```

```
##  Ljung-Box Test     R^2  Q(20)  17.75261  0.6037013
##  LM Arch Test       R    TR^2   8.048057  0.7813637
##
## Information Criterion Statistics:
##      AIC      BIC      SIC     HQIC
## 16.44992 16.51609 16.44888 16.47680
```

IBOVESPA

```
##
## Series Initialization:
##  ARMA Model:                arma
##  Formula Mean:              ~ arma(0, 0)
##  GARCH Model:               garch
##  Formula Variance:          ~ garch(1, 0)
##  ARMA Order:                0 0
##  Max ARMA Order:            0
##  GARCH Order:               1 0
##  Max GARCH Order:           1
##  Maximum Order:             1
##  Conditional Dist:          norm
##  h.start:                   2
##  llh.start:                 1
##  Length of Series:          131
##  Recursion Init:            mci
##  Series Scale:              6012.416
##
## Parameter Initialization:
##  Initial Parameters:          $params
##  Limits of Transformations:   $U, $V
##  Which Parameters are Fixed?  $includes
##  Parameter Matrix:
##                        U        V   params includes
##     mu     -182.17257374 182.1726 18.21726     TRUE
##     omega     0.00000100 100.0000  0.10000     TRUE
##     alpha1    0.00000001   1.0000  0.10000     TRUE
##     gamma1   -0.99999999   1.0000  0.10000    FALSE
##     delta     0.00000000   2.0000  2.00000    FALSE
##     skew      0.10000000  10.0000  1.00000    FALSE
##     shape     1.00000000  10.0000  4.00000    FALSE
##  Index List of Parameters to be Optimized:
##     mu  omega alpha1
##      1      2      3
##  Persistence:                  0.1
##
##
## --- START OF TRACE ---
## Selected Algorithm: nlminb
##
## R coded nlminb Solver:
##
##   0:     251.93855:  18.2173 0.100000 0.100000
##   1:     235.42142:  18.9596 0.129324 0.142576
##   2:     190.29040:  18.0910 0.151868 0.181962
```

```
##   3:     166.50594:  18.5655 0.180030 0.235456
##   4:     159.58285:  18.2352 0.176240 0.298623
##   5:     151.22015:  18.5405 0.132881 0.345230
##   6:     147.71651:  18.4866 0.0809566 0.385583
##   7:     142.66055:  18.6383 0.0609858 0.447752
##   8:     140.82603:  18.4648 0.112321 0.487842
##   9:     136.43383:  18.5543 0.0654048 0.533754
##  10:     136.33013:  18.5770 0.0652078 0.534217
##  11:     136.10678:  18.5583 0.0624456 0.538716
##  12:     135.75384:  18.5870 0.0596728 0.543048
##  13:     133.22176:  18.5975 0.0369784 0.594356
##  14:     129.05687:  18.5886 0.0480436 0.749497
##  15:     127.24643:  18.6303 0.0323786 0.818081
##  16:     126.28835:  18.6194 0.0197546 0.887324
##  17:     125.59325:  18.6129 0.0227167 0.957648
##  18:     125.55398:  18.6377 0.0210874  1.00000
##  19:     125.35905:  18.6198 0.0215701  1.00000
##  20:     125.35894:  18.6202 0.0215896  1.00000
##  21:     125.35894:  18.6202 0.0215769  1.00000
##  22:     125.35894:  18.6202 0.0215762  1.00000
##
## Final Estimate of the Negative LLH:
##  LLH:  1265.266    norm LLH:  9.65852
##       mu    omega   alpha1
## 111952.7 779962.0      1.0
##
## R-optimhess Difference Approximated Hessian Matrix:
##                   mu         omega        alpha1
## mu     -3.652678e-05 -5.193615e-09 -1.378788e-03
## omega  -5.193615e-09 -1.186153e-11 -8.806838e-06
## alpha1 -1.378788e-03 -8.806838e-06 -5.235485e+01
## attr(,"time")
## Time difference of 0.003089905 secs
##
## --- END OF TRACE ---
##
##
## Time to Estimate Parameters:
##  Time difference of 0.02644658 secs

## Warning: Using formula(x) is deprecated when x is a character vector of length > 1.
##   Consider formula(paste(x, collapse = " ")) instead.

##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~garch(1, 0), data = bovespa.ts, trace = T)
##
## Mean and Variance Equation:
##  data ~ garch(1, 0)
## <environment: 0x563055ea4680>
##  [data = bovespa.ts]
##
```

```
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##     mu   omega  alpha1
## 111953  779962       1
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu     1.120e+05   1.712e+02  653.930  < 2e-16 ***
## omega  7.800e+05   3.210e+05    2.430   0.0151 *
## alpha1 1.000e+00   1.480e-01    6.756 1.42e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  -1265.266    normalized:  -9.65852
##
## Description:
##  Tue Jul 12 16:49:41 2022 by user: stevenson
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  7.504487  0.02346504
##  Shapiro-Wilk Test  R    W      0.9276877 2.89362e-06
##  Ljung-Box Test     R    Q(10)  399.1648  0
##  Ljung-Box Test     R    Q(15)  472.7641  0
##  Ljung-Box Test     R    Q(20)  485.3374  0
##  Ljung-Box Test     R^2  Q(10)  21.192    0.0197938
##  Ljung-Box Test     R^2  Q(15)  25.27771  0.04633586
##  Ljung-Box Test     R^2  Q(20)  29.87073  0.07197562
##  LM Arch Test       R    TR^2   17.17361  0.1431826
##
## Information Criterion Statistics:
##      AIC      BIC      SIC     HQIC
## 19.36284 19.42869 19.36182 19.38960
```

Brent

```
##
## Series Initialization:
##  ARMA Model:                arma
##  Formula Mean:              ~ arma(0, 0)
##  GARCH Model:               garch
##  Formula Variance:          ~ garch(1, 0)
##  ARMA Order:                0 0
##  Max ARMA Order:            0
##  GARCH Order:               1 0
##  Max GARCH Order:           1
##  Maximum Order:             1
```

```
##  Conditional Dist:          norm
##  h.start:                   2
##  llh.start:                 1
##  Length of Series:          131
##  Recursion Init:            mci
##  Series Scale:              11.82754
##
## Parameter Initialization:
##  Initial Parameters:          $params
##  Limits of Transformations:   $U, $V
##  Which Parameters are Fixed?  $includes
##  Parameter Matrix:
##                      U         V   params includes
##     mu     -88.84480729  88.84481 8.884481     TRUE
##     omega    0.00000100 100.00000 0.100000     TRUE
##     alpha1   0.00000001   1.00000 0.100000     TRUE
##     gamma1  -0.99999999   1.00000 0.100000    FALSE
##     delta    0.00000000   2.00000 2.000000    FALSE
##     skew     0.10000000  10.00000 1.000000    FALSE
##     shape    1.00000000  10.00000 4.000000    FALSE
##  Index List of Parameters to be Optimized:
##     mu  omega alpha1
##      1      2      3
##  Persistence:                  0.1
##
##
## --- START OF TRACE ---
## Selected Algorithm: nlminb
##
## R coded nlminb Solver:
##
##   0:     268.45500:  8.88448 0.100000 0.100000
##   1:     201.67614:  9.87599 0.225578 0.255771
##   2:     166.97632:  8.94479 0.230935 0.277598
##   3:     155.20346:  9.28565 0.238207 0.377427
##   4:     150.35043:  9.14109 0.164592 0.453628
##   5:     149.35571:  9.33266 0.0741616 0.506995
##   6:     144.70864:  9.27360 0.133294 0.596156
##   7:     144.64542:  9.19917 0.127976 0.600212
##   8:     144.21560:  9.25647 0.124628 0.608093
##   9:     144.00543:  9.22607 0.117176 0.614997
##  10:     143.73525:  9.26757 0.114132 0.624152
##  11:     143.53193:  9.23712 0.110296 0.633556
##  12:     143.31033:  9.27024 0.107679 0.643260
##  13:     143.12519:  9.24448 0.104701 0.653141
##  14:     142.93522:  9.27189 0.102590 0.663188
##  15:     141.44288:  9.26435 0.0776858 0.819160
##  16:     141.23127:  9.28014 0.0900582 0.893657
##  17:     141.13717:  9.28966 0.0801907 0.912090
##  18:     141.13405:  9.28608 0.0788208 0.917371
##  19:     141.13388:  9.28651 0.0790700 0.918276
##  20:     141.13388:  9.28659 0.0790615 0.918701
##  21:     141.13388:  9.28658 0.0790619 0.918727
##  22:     141.13388:  9.28658 0.0790613 0.918728
```

```
##
## Final Estimate of the Negative LLH:
##  LLH:  464.7603    norm LLH:  3.547788
##          mu       omega      alpha1
## 109.8373401  11.0599393   0.9187277
##
## R-optimhess Difference Approximated Hessian Matrix:
##                  mu       omega      alpha1
## mu     -1.50769951 -0.09327188   0.5783378
## omega  -0.09327188 -0.11128899  -0.7031605
## alpha1  0.57833779 -0.70316050 -44.5553759
## attr(,"time")
## Time difference of 0.00270772 secs
##
## --- END OF TRACE ---
##
##
## Time to Estimate Parameters:
##  Time difference of 0.01753044 secs
```

```
## Warning: Using formula(x) is deprecated when x is a character vector of length > 1.
##   Consider formula(paste(x, collapse = " ")) instead.
```

```
##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~garch(1, 0), data = brent.ts, trace = T)
##
## Mean and Variance Equation:
##  data ~ garch(1, 0)
## <environment: 0x563054cbcfb8>
##  [data = brent.ts]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##        mu      omega     alpha1
## 109.83734   11.05994    0.91873
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      109.8373      0.8465  129.754  < 2e-16 ***
## omega    11.0599      3.2756    3.376 0.000734 ***
## alpha1    0.9187      0.1598    5.749 8.97e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  -464.7603    normalized:  -3.547788
```

```
##
## Description:
##  Tue Jul 12 16:49:41 2022 by user: stevenson
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  5.515338  0.06343948
##  Shapiro-Wilk Test  R    W      0.9418788 2.646811e-05
##  Ljung-Box Test     R    Q(10)  244.5538  0
##  Ljung-Box Test     R    Q(15)  325.2879  0
##  Ljung-Box Test     R    Q(20)  342.4151  0
##  Ljung-Box Test     R^2  Q(10)  8.747982  0.5561764
##  Ljung-Box Test     R^2  Q(15)  14.67914  0.474767
##  Ljung-Box Test     R^2  Q(20)  15.98992  0.7172495
##  LM Arch Test       R    TR^2   11.12336  0.5183776
##
## Information Criterion Statistics:
##      AIC      BIC      SIC     HQIC
## 7.141378 7.207222 7.140360 7.168134
```

Dólar

```
##
## Series Initialization:
##  ARMA Model:                arma
##  Formula Mean:              ~ arma(0, 0)
##  GARCH Model:               garch
##  Formula Variance:          ~ garch(1, 1)
##  ARMA Order:                0 0
##  Max ARMA Order:            0
##  GARCH Order:               1 1
##  Max GARCH Order:           1
##  Maximum Order:             1
##  Conditional Dist:          norm
##  h.start:                   2
##  llh.start:                 1
##  Length of Series:          136
##  Recursion Init:            mci
##  Series Scale:              0.2776031
##
## Parameter Initialization:
##  Initial Parameters:          $params
##  Limits of Transformations:   $U, $V
##  Which Parameters are Fixed?  $includes
##  Parameter Matrix:
##                         U        V   params includes
##     mu     -183.19360162 183.1936 18.31936     TRUE
##     omega     0.00000100 100.0000  0.10000     TRUE
##     alpha1    0.00000001   1.0000  0.10000     TRUE
##     gamma1   -0.99999999   1.0000  0.10000    FALSE
##     beta1     0.00000001   1.0000  0.80000     TRUE
##     delta     0.00000000   2.0000  2.00000    FALSE
##     skew      0.10000000  10.0000  1.00000    FALSE
```

```
##     shape     1.00000000  10.0000  4.00000    FALSE
##  Index List of Parameters to be Optimized:
##     mu  omega alpha1  beta1
##      1      2      3      5
##  Persistence:                  0.9
##
##
## --- START OF TRACE ---
## Selected Algorithm: nlminb
##
## R coded nlminb Solver:
##
##   0:     173.12871:  18.3194 0.100000 0.100000 0.800000
##   1:     170.07868:  18.1120 0.0838948 0.100490 0.787443
##   2:     167.66231:  18.4173 0.0701685 0.103390 0.779030
##   3:     163.92841:  18.1965 0.0535332 0.109751 0.769925
##   4:     160.81396:  18.3720 0.0374133 0.121994 0.763299
##   5:     157.97408:  18.2427 0.0241707 0.139447 0.759338
##   6:     155.65652:  18.3787 0.0186134 0.160687 0.756461
##   7:     153.69152:  18.2695 0.0160893 0.181201 0.747368
##   8:     152.92678:  18.3440 0.00699912 0.191082 0.728696
##   9:     150.91002:  18.3373 0.0176895 0.204425 0.712791
##  10:     150.73933:  18.3051 0.0159996 0.205661 0.711854
##  11:     150.63945:  18.3457 0.0149780 0.206984 0.711054
##  12:     150.44274:  18.3127 0.0145145 0.208597 0.709540
##  13:     150.31388:  18.3396 0.0140343 0.210576 0.708107
##  14:     150.14428:  18.3132 0.0138353 0.212374 0.706376
##  15:     150.00916:  18.3375 0.0136995 0.214370 0.704766
##  16:     149.84999:  18.3136 0.0136068 0.216185 0.702938
##  17:     149.71547:  18.3368 0.0135643 0.218140 0.701228
##  18:     149.56274:  18.3139 0.0135161 0.219929 0.699330
##  19:     149.42986:  18.3362 0.0135052 0.221839 0.697536
##  20:     149.28283:  18.3144 0.0134814 0.223600 0.695581
##  21:     149.15204:  18.3359 0.0134865 0.225469 0.693716
##  22:     149.01018:  18.3148 0.0134800 0.227207 0.691715
##  23:     148.88174:  18.3356 0.0134953 0.229041 0.689791
##  24:     148.74461:  18.3152 0.0135020 0.230760 0.687752
##  25:     148.61864:  18.3354 0.0135241 0.232565 0.685783
##  26:     148.48583:  18.3156 0.0135407 0.234272 0.683714
##  27:     148.36238:  18.3352 0.0135670 0.236055 0.681708
##  28:     148.23352:  18.3161 0.0135905 0.237755 0.679619
##  29:     148.11255:  18.3352 0.0136187 0.239525 0.677586
##  30:     147.98730:  18.3165 0.0136465 0.241226 0.675483
##  31:     147.86874:  18.3351 0.0136748 0.242990 0.673432
##  32:     147.74677:  18.3169 0.0137048 0.244698 0.671323
##  33:     147.63051:  18.3351 0.0137316 0.246464 0.669262
##  34:     147.51155:  18.3174 0.0137623 0.248184 0.667152
##  35:     147.39750:  18.3352 0.0137866 0.249957 0.665088
##  36:     147.28133:  18.3178 0.0138171 0.251694 0.662983
##  37:     147.16943:  18.3352 0.0138387 0.253480 0.660919
##  38:     147.05588:  18.3182 0.0138692 0.255235 0.658821
##  39:     146.94610:  18.3353 0.0138888 0.257034 0.656761
##  40:     146.83508:  18.3186 0.0139203 0.258807 0.654670
##  41:     146.72742:  18.3354 0.0139391 0.260618 0.652614
```

```
##  42:     146.61888:  18.3189 0.0139729 0.262407 0.650530
##  43:     146.51336:  18.3355 0.0139924 0.264228 0.648475
##  44:     146.40724:  18.3193 0.0140300 0.266030 0.646396
##  45:     146.30384:  18.3356 0.0140513 0.267859 0.644343
##  46:     146.20007:  18.3197 0.0140935 0.269672 0.642267
##  47:     146.09876:  18.3358 0.0141174 0.271508 0.640215
##  48:     145.99725:  18.3201 0.0141644 0.273331 0.638142
##  49:     145.89799:  18.3359 0.0141912 0.275175 0.636090
##  50:     145.79866:  18.3205 0.0142430 0.277008 0.634020
##  51:     145.70141:  18.3361 0.0142727 0.278858 0.631968
##  52:     145.60421:  18.3210 0.0143288 0.280699 0.629899
##  53:     145.50892:  18.3362 0.0143613 0.282556 0.627847
##  54:     145.41380:  18.3214 0.0144210 0.284405 0.625779
##  55:     145.32046:  18.3364 0.0144560 0.286268 0.623726
##  56:     145.22737:  18.3219 0.0145187 0.288125 0.621659
##  57:     145.13594:  18.3365 0.0145562 0.289992 0.619604
##  58:     145.04483:  18.3223 0.0146210 0.291856 0.617537
##  59:     144.95529:  18.3367 0.0146609 0.293729 0.615481
##  60:     144.86611:  18.3228 0.0147273 0.295599 0.613413
##  61:     144.77843:  18.3369 0.0147694 0.297476 0.611355
##  62:     144.69114:  18.3233 0.0148369 0.299353 0.609287
##  63:     144.60528:  18.3371 0.0148814 0.301234 0.607226
##  64:     144.51983:  18.3238 0.0149493 0.303117 0.605158
##  65:     144.43576:  18.3373 0.0149962 0.305002 0.603094
##  66:     144.35211:  18.3243 0.0150640 0.306891 0.601026
##  67:     144.26979:  18.3375 0.0151134 0.308779 0.598960
##  68:     143.82273:  18.3259 0.0108321 0.331353 0.568259
##  69:     142.61444:  18.3392 0.0194853 0.347202 0.534433
##  70:     142.11208:  18.2859 0.0172605 0.384427 0.525962
##  71:     141.14092:  18.3501 0.0188808 0.399552 0.490931
##  72:     140.93995:  18.3328 0.0206917 0.406278 0.453230
##  73:     139.46083:  18.2954 0.0236205 0.517776 0.403125
##  74:     138.28985:  18.4171 0.0269535 0.596181 0.309569
##  75:     136.74051:  18.3882 0.0343669 0.770947 0.103651
##  76:     136.61995:  18.3855 0.0340901 0.803038 0.0936581
##  77:     136.50530:  18.3845 0.0321184 0.844485 0.105182
##  78:     136.49520:  18.3898 0.0317291 0.861938 0.0976415
##  79:     136.49319:  18.3926 0.0312492 0.864198 0.0994304
##  80:     136.49246:  18.3940 0.0309082 0.863232 0.100624
##  81:     136.49221:  18.3941 0.0306833 0.861292 0.101788
##  82:     136.49217:  18.3937 0.0306379 0.860563 0.102132
##  83:     136.49217:  18.3936 0.0306391 0.860520 0.102154
##  84:     136.49217:  18.3936 0.0306400 0.860541 0.102147
##  85:     136.49217:  18.3936 0.0306400 0.860544 0.102146
##
## Final Estimate of the Negative LLH:
##  LLH:  -37.80039    norm LLH:  -0.2779441
##          mu       omega      alpha1       beta1
## 5.106114578 0.002361222 0.860544479 0.102146465
##
## R-optimhess Difference Approximated Hessian Matrix:
##                   mu        omega      alpha1        beta1
## mu      -2462.38567   -18091.289    26.42886   -154.50976
## omega  -18091.28929 -1383847.194 -3074.14652 -11129.56734
```

```
## alpha1     26.42886    -3074.147   -64.21477    -73.57294
## beta1    -154.50976   -11129.567   -73.57294   -192.65245
## attr(,"time")
## Time difference of 0.006896019 secs
##
## --- END OF TRACE ---
##
##
## Time to Estimate Parameters:
##  Time difference of 0.05016685 secs

## Warning: Using formula(x) is deprecated when x is a character vector of length > 1.
##   Consider formula(paste(x, collapse = " ")) instead.

##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~garch(1, 1), data = dolar.ts, trace = T)
##
## Mean and Variance Equation:
##  data ~ garch(1, 1)
## <environment: 0x563053a768f0>
##  [data = dolar.ts]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##        mu      omega     alpha1      beta1
## 5.1061146  0.0023612  0.8605445  0.1021465
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      5.106115    0.021935  232.784  < 2e-16 ***
## omega   0.002361    0.001210    1.952   0.0509 .
## alpha1  0.860544    0.176798    4.867 1.13e-06 ***
## beta1   0.102146    0.129586    0.788   0.4305
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  37.80039    normalized:  0.2779441
##
## Description:
##  Tue Jul 12 16:49:41 2022 by user: stevenson
##
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  11.46655  0.003236466
```

```
##  Shapiro-Wilk Test  R    W      0.8979165 3.472768e-08
##  Ljung-Box Test     R    Q(10)  553.0233  0
##  Ljung-Box Test     R    Q(15)  662.7706  0
##  Ljung-Box Test     R    Q(20)  703.8948  0
##  Ljung-Box Test     R^2  Q(10)  17.0636   0.07296837
##  Ljung-Box Test     R^2  Q(15)  22.48064  0.09580678
##  Ljung-Box Test     R^2  Q(20)  26.10934  0.162231
##  LM Arch Test       R    TR^2   18.35161  0.105424
##
## Information Criterion Statistics:
##        AIC        BIC        SIC       HQIC
## -0.4970646 -0.4113983 -0.4987297 -0.4622520
```

Bitcoin

```
##
## Series Initialization:
##  ARMA Model:                arma
##  Formula Mean:              ~ arma(0, 0)
##  GARCH Model:               garch
##  Formula Variance:          ~ garch(1, 0)
##  ARMA Order:                0 0
##  Max ARMA Order:            0
##  GARCH Order:               1 0
##  Max GARCH Order:           1
##  Maximum Order:             1
##  Conditional Dist:          norm
##  h.start:                   2
##  llh.start:                 1
##  Length of Series:          193
##  Recursion Init:            mci
##  Series Scale:              8210.711
##
## Parameter Initialization:
##  Initial Parameters:          $params
##  Limits of Transformations:   $U, $V
##  Which Parameters are Fixed?  $includes
##  Parameter Matrix:
##                        U         V   params includes
##     mu     -43.65922930  43.65923 4.365923     TRUE
##     omega    0.00000100 100.00000 0.100000     TRUE
##     alpha1   0.00000001   1.00000 0.100000     TRUE
##     gamma1  -0.99999999   1.00000 0.100000    FALSE
##     delta    0.00000000   2.00000 2.000000    FALSE
##     skew     0.10000000  10.00000 1.000000    FALSE
##     shape    1.00000000  10.00000 4.000000    FALSE
##  Index List of Parameters to be Optimized:
##     mu  omega alpha1
##      1      2      3
##  Persistence:                  0.1
##
##
## --- START OF TRACE ---
## Selected Algorithm: nlminb
```

```
##
## R coded nlminb Solver:
##
##   0:     381.89332:  4.36592 0.100000 0.100000
##   1:     372.19041:  6.22052 0.670928 0.802564
##   2:     267.95123:  4.03796 0.661614 0.799921
##   3:     195.25767:  4.64866 0.182868 0.835082
##   4:     186.85052:  4.76525 0.171715 0.834655
##   5:     178.51308:  5.00770 0.117074 0.833132
##   6:     160.05733:  4.82572 0.0513729 0.837360
##   7:     155.18893:  4.86079 0.0239382 0.837541
##   8:     154.63929:  4.82964 0.0228360 0.837998
##   9:     154.60023:  4.82705 0.0214955 0.838307
##  10:     154.58916:  4.82182 0.0212494 0.839174
##  11:     154.58117:  4.82627 0.0210850 0.840259
##  12:     154.56874:  4.82290 0.0215714 0.841448
##  13:     154.55654:  4.82396 0.0208746 0.842752
##  14:     154.54315:  4.82303 0.0213852 0.844144
##  15:     154.53188:  4.82397 0.0208551 0.845529
##  16:     154.51906:  4.82295 0.0213363 0.846929
##  17:     154.50793:  4.82381 0.0208221 0.848322
##  18:     154.49564:  4.82284 0.0212871 0.849729
##  19:     154.48468:  4.82366 0.0207876 0.851129
##  20:     154.47289:  4.82274 0.0212373 0.852543
##  21:     154.46215:  4.82351 0.0207514 0.853949
##  22:     154.45082:  4.82263 0.0211869 0.855368
##  23:     154.44033:  4.82337 0.0207139 0.856780
##  24:     154.42946:  4.82253 0.0211361 0.858205
##  25:     154.41924:  4.82322 0.0206754 0.859622
##  26:     154.40880:  4.82243 0.0210851 0.861052
##  27:     154.39887:  4.82308 0.0206362 0.862473
##  28:     154.38884:  4.82233 0.0210341 0.863907
##  29:     154.37924:  4.82295 0.0205968 0.865333
##  30:     154.36960:  4.82223 0.0209834 0.866772
##  31:     154.36032:  4.82283 0.0205579 0.868202
##  32:     154.35107:  4.82215 0.0209332 0.869644
##  33:     154.34211:  4.82271 0.0205201 0.871079
##  34:     154.33324:  4.82209 0.0208835 0.872525
##  35:     154.29823:  4.81860 0.0200538 0.880661
##  36:     154.25208:  4.82602 0.0206041 0.888682
##  37:     154.21592:  4.82220 0.0198866 0.896821
##  38:     154.19001:  4.82140 0.0208923 0.904974
##  39:     154.16338:  4.82516 0.0202267 0.913119
##  40:     154.12818:  4.82503 0.0204592 0.937916
##  41:     154.11996:  4.82225 0.0195458 0.937940
##  42:     154.11733:  4.82171 0.0198404 0.939007
##  43:     154.11519:  4.82148 0.0197641 0.943463
##  44:     154.11442:  4.82159 0.0197215 0.947919
##  45:     154.11442:  4.82156 0.0197171 0.948112
##  46:     154.11442:  4.82156 0.0197174 0.948111
##
## Final Estimate of the Negative LLH:
##  LLH:  1893.661    norm LLH:  9.811715
##           mu        omega       alpha1
```

```
## 3.958847e+04 1.329267e+06 9.481111e-01
##
## R-optimhess Difference Approximated Hessian Matrix:
##                   mu         omega        alpha1
## mu     -1.586489e-05  1.944552e-09 -3.944774e-04
## omega   1.944552e-09 -8.003528e-12 -6.180764e-06
## alpha1 -3.944774e-04 -6.180764e-06 -7.447625e+01
## attr(,"time")
## Time difference of 0.002733946 secs
##
## --- END OF TRACE ---
##
##
## Time to Estimate Parameters:
##  Time difference of 0.02901888 secs

## Warning: Using formula(x) is deprecated when x is a character vector of length > 1.
##   Consider formula(paste(x, collapse = " ")) instead.

##
## Title:
##  GARCH Modelling
##
## Call:
##  garchFit(formula = ~garch(1, 0), data = bitcoin.ts, trace = T)
##
## Mean and Variance Equation:
##  data ~ garch(1, 0)
## <environment: 0x5630513b7860>
##  [data = bitcoin.ts]
##
## Conditional Distribution:
##  norm
##
## Coefficient(s):
##         mu       omega      alpha1
## 3.9588e+04  1.3293e+06  9.4811e-01
##
## Std. Errors:
##  based on Hessian
##
## Error Analysis:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu     3.959e+04   2.553e+02  155.058  < 2e-16 ***
## omega  1.329e+06   3.715e+05    3.578 0.000347 ***
## alpha1 9.481e-01   1.200e-01    7.902 2.66e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log Likelihood:
##  -1893.661    normalized:  -9.811715
##
## Description:
##  Tue Jul 12 16:49:41 2022 by user: stevenson
##
```

```
##
## Standardised Residuals Tests:
##                                 Statistic p-Value
##  Jarque-Bera Test   R    Chi^2  16.56077  0.0002534399
##  Shapiro-Wilk Test  R    W      0.8880755 8.046027e-11
##  Ljung-Box Test     R    Q(10)  715.3048  0
##  Ljung-Box Test     R    Q(15)  815.9509  0
##  Ljung-Box Test     R    Q(20)  877.1481  0
##  Ljung-Box Test     R^2  Q(10)  5.785277  0.8329691
##  Ljung-Box Test     R^2  Q(15)  9.259779  0.8635473
##  Ljung-Box Test     R^2  Q(20)  12.85311  0.883597
##  LM Arch Test       R    TR^2   6.818285  0.8693847
##
## Information Criterion Statistics:
##      AIC      BIC      SIC     HQIC
## 19.65452 19.70523 19.65404 19.67506
```

Por fim, percebe-se que os p-valores dos coeficientes são todos inferiores a 5%, o que configura-os como cabíveis ao modelo.

Previsões da Série

Finalmente, temos a previsão para o valor seguinte de cada uma das séries, que usa os respectivos modelos ARCH/GARCH encontrados anteriormente, considerando apenas os últimos 14 valores da série.

S&P500

```
## Warning in sqrt(diag(fit$cvar)): NaNs produced

## Warning: Using formula(x) is deprecated when x is a character vector of length > 1.
##   Consider formula(paste(x, collapse = " ")) instead.

##   meanForecast meanError standardDeviation
## 1     3841.191  38.03641          38.03641
```

Dow Jones

```
## Warning in sqrt(diag(fit$cvar)): NaNs produced

## Warning: Using formula(x) is deprecated when x is a character vector of length > 1.
##   Consider formula(paste(x, collapse = " ")) instead.

##   meanForecast meanError standardDeviation
## 1     31059.81  216.4632          216.4632
```

Nasdaq

```
## Warning: Using formula(x) is deprecated when x is a character vector of length > 1.
##   Consider formula(paste(x, collapse = " ")) instead.

##   meanForecast meanError standardDeviation
## 1     11308.29  210.4978          210.4978
```

IBOVESPA

```
## Warning: Using formula(x) is deprecated when x is a character vector of length > 1.
##   Consider formula(paste(x, collapse = " ")) instead.
```

```
##   meanForecast meanError standardDeviation
## 1     99257.21  946.6562          946.6562
```

Brent

```
## Warning: Using formula(x) is deprecated when x is a character vector of length > 1.
##   Consider formula(paste(x, collapse = " ")) instead.
```

```
##   meanForecast meanError standardDeviation
## 1     111.1116  4.720819          4.720819
```

Dólar

```
## Warning in sqrt(diag(fit$cvar)): NaNs produced
```

```
## Warning: Using formula(x) is deprecated when x is a character vector of length > 1.
##   Consider formula(paste(x, collapse = " ")) instead.
```

```
##   meanForecast  meanError standardDeviation
## 1     5.280812 0.07819604        0.07819604
```

Bitcoin

```
## Warning: Using formula(x) is deprecated when x is a character vector of length > 1.
##   Consider formula(paste(x, collapse = " ")) instead.
```

```
##   meanForecast meanError standardDeviation
## 1     19785.37  406.9364          406.9364
```

Conclusão

Os códigos utilizados para a elaboração da solução da prova estão disponíveis no seguinte *repositório do git*.