CE064 - MODELOS MARKOVIANOS
(TÓPICOS ESPECIAIS EM ESTATÍSTICA IV)
Departamento de Estatística
Universidade Federal do Paraná

Trabalho No.2

Henrique Aparecido Laureano
GRR: 20115307

Junho de 2016

Sumário

Objetivo **2**

Análise descritiva **2**

Modelos Ocultos de Markov **4**

Modelo de dois estados 4

Modelo de três estados 9

Objetivo

Modelar uma série de log-retornos correspondente à cotação do Euro em relação ao Dólar Americano entre 03 de janeiro de 2012 e 30 de março de 2015, utilizando Modelos Ocultos de Markov.

O log-retorno é definido como

$$r_t = \log(y_t / x_t),$$

em que y_t representa, aqui, o valor do câmbio no fechamento do dia de operações na bolsa de valores e x_t representa o valor na abertura dos negócios.

Os dados são contínuos, o que sugere uma resposta gaussiana para o log-retorno e dois (decisão de vender ou comprar dólares) ou três (acrescentar, ainda, a situação de não mexer na carteira de Euros) estados para a Cadeia de Markov oculta.

Análise descritiva

Com o auxílio da Figura 1, página 3, observa-se que durante toda a série temporal os valores de abertura, fechamento, mínimo e máximo não diferem muito, i.e., nenhuma discrepância é observada.
Nos últimos meses do primeiro semestre de 2012 observa-se uma queda, seguida de uma subida e de uma constância até o final do primeiro semestre de 2014. Desse período em diante uma queda considerável é observada.

```
# <code r> ================================================================= #
path <- "C:/Users/henri/Dropbox/Scripts/markovian models/list 2/"

dados <- read.csv(paste0(path, "EURUSD1d.csv"))

datas <- as.Date(substr(dados$Open.Timestamp, 1, 10), format = "%Y.%m.%d")

library(latticeExtra)

xyplot(dados$Open + dados$High + dados$Low + dados$Close ~ datas
       , type = c("l", "g")
       , col = 1:4
       , lwd = 2
       , xlab = "Tempo: ano-mês"
       , ylab = NULL
```

```
       , key = list(text = list(c("Abertura", "Maior", "Menor", "Fechamento"))
                    , lines = TRUE
                    , col = 1:4
                    , lwd = 2
                    , columns = 4))
# </code r> =============================================================== #
```

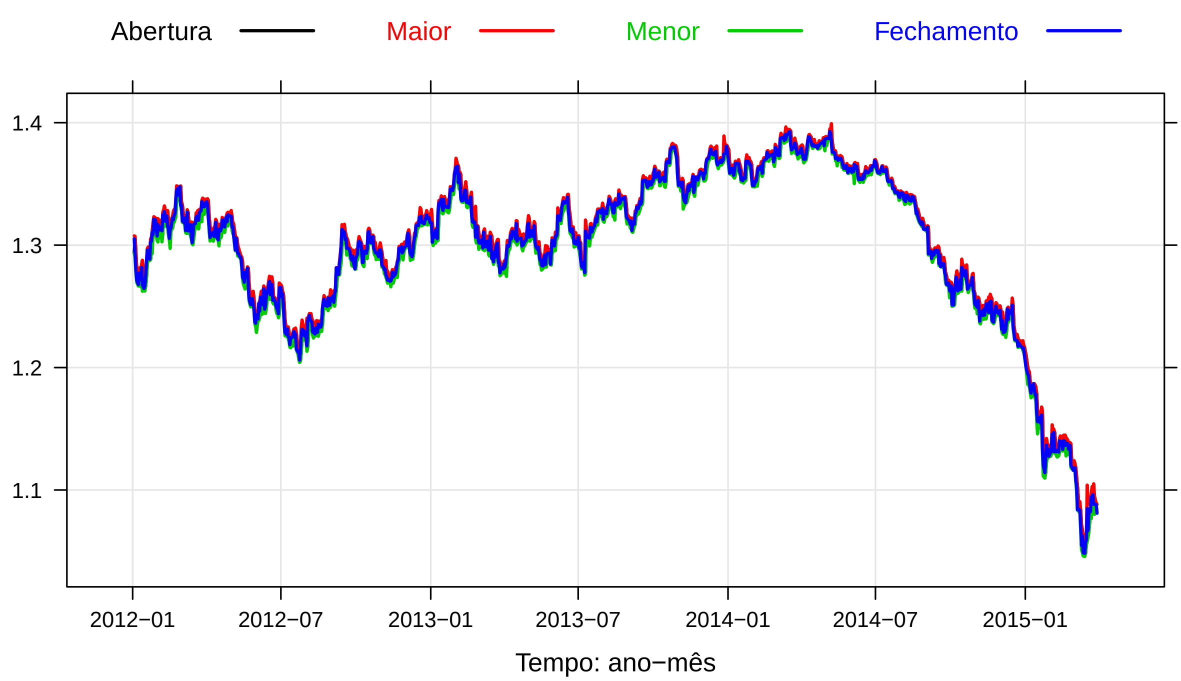

Figura 1: Valores diários do câmbio na abertura, maior e menor valor da cotação da moeda e do valor no fechamento do mercado de câmbio

Olhando para os valores de log-retorno na Figura 2, página 4, nota-se uma grande variação logo no início do segundo semestre de 2013, mas que se mostra passageira. O primeiro semestre de 2014 é o período que apresenta as menores variações do log-retorno. A partir desse semestre sua variabilidade volta a aumentar.

```
# <code r> ================================================================ #
logretorno <- log(dados$Close) - log(dados$Open)

xyplot(logretorno ~ datas
       , type = c("l", "g")
       , lwd = 2
       , xlab = "Tempo: ano-mês"
       , ylab = list(rot = 0, label = expression(r[t])))
# </code r> =============================================================== #
```

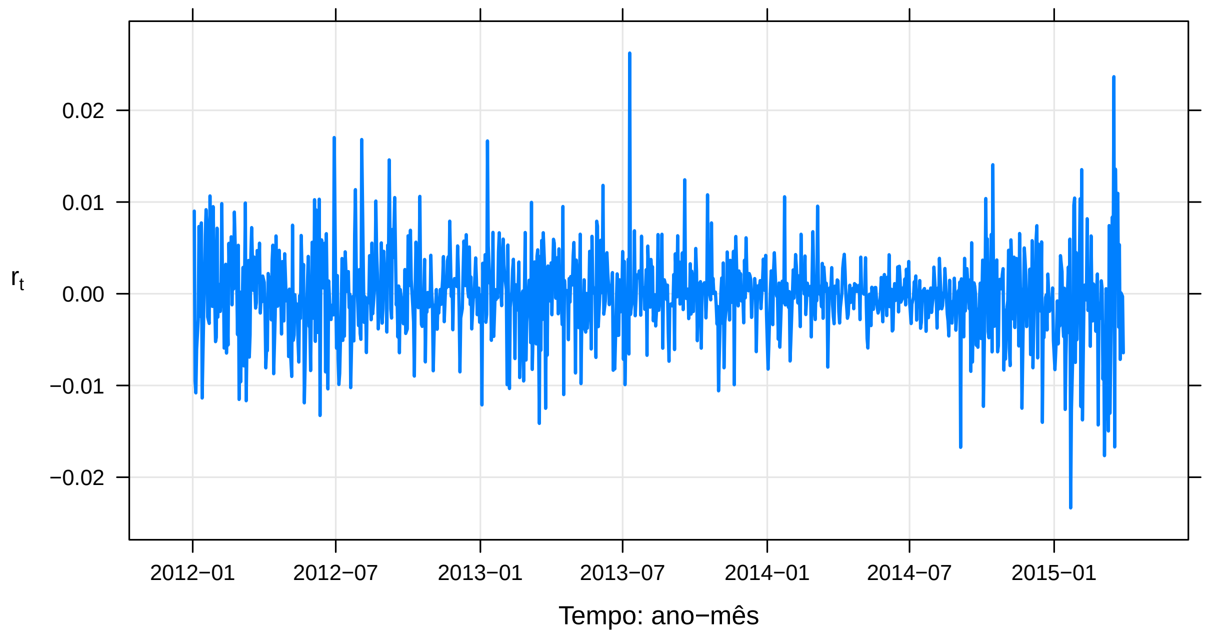

Figura 2: Valores do log-retorno

Modelos Ocultos de Markov

```r
# <code r> ================================================================== #
library(HiddenMarkov)
# </code r> ================================================================= #
```

Modelo de dois estados

Para a Cadeia de Markov oculta pode-se pensar em dois estados, vender e comprar dólares.

Um log-retorno negativo indica que o valor da cotação no fechamento é inferior ao valor na abertura, logo, a tomada de decisão mais indicada é vender dólares. Consequentemente, um log-retorno positivo indica à compra.

```r
# <code r> ================================================================== #
## 0: venda
## 1: compra
```

```r
logest2 <- ifelse(logretorno < 0, 0, 1)
# </code r> ============================================================== #
```

É necessário fornecer uma matriz de probabilidades de transição iniciais entre os estados.

Tenta-se propor uma boa matriz, no sentido de ser altamente coerente com o que acontece com os log-retornos ao longo da série temporal.

Para cada log-retorno foi obtido um valor, 0 ou 1, indicando se o mais adequado seria vender ou comprar dólares, respectivamente. Ou seja, para qual estado aquele log-retorno corresponde.

```r
# <code r> =============================================================== #
table(logest2)
# </code r> ============================================================== #
```

```
logest2
  0   1 
513 495 
```

Em 513 dias o mais adequado seria vender, e em 495, comprar.

Fazendo a diferença entre esses valores obtidos é retornado novos três valores:

```r
# <code r> =============================================================== #
unique(diff(logest2))
# </code r> ============================================================== #
```

```
[1] -1  0  1
```

O valor -1 é obtido quando o log-retorno do dia x é 1 e do dia $x+1$ é 0, i.e., no dia x a tomada de decisão indicada é a compra e no dia seguinte é a venda.

```r
# <code r> =============================================================== #
table(diff(logest2) == -1)
# </code r> ============================================================== #
```

```
FALSE  TRUE 
  749   258 
```

Dos 1008 log-retornos e consequentemente, 1007 diferenças, 258 correspondem a transição do estado 1 (compra) para o estado 0 (venda).

```r
# <code r> =============================================================== #
table(diff(logest2) == -1)[[2]] / length(diff(logest2))
# </code r> ============================================================== #
```

```
[1] 0.2562066
```

Já o valor 1 é obtido quando o log-retorno do dia x é 0 e do dia $x + 1$ é 1, i.e., no dia x a tomada de decisão indicada é a venda e no dia seguinte é a compra.

```r
# <code r> ================================================================= #
table(diff(logest2) == 1)
# </code r> ================================================================ #
```

```

FALSE  TRUE
  750   257
```

257 é o número de transições do estado 0 (venda) para o estado 1 (compra).

```r
# <code r> ================================================================= #
table(diff(logest2) == 1)[[2]] / length(diff(logest2))
# </code r> ================================================================ #
```

```
[1] 0.2552135
```

Em ambas situações observa-se que o número de mudanças de tomada de decisão de um dia para o outro é praticamente a metade do número de ocorrências de cada estado. Isso nos faz pensar que a matriz de probabilidades de transição que melhor representa essa série temporal é dada por

$$P = \begin{array}{c c} & \begin{array}{cc} \text{Venda} & \text{Compra} \end{array} \\ \begin{array}{c} \text{Venda} \\ \text{Compra} \end{array} & \begin{pmatrix} 0.50 & 0.50 \\ 0.50 & 0.50 \end{pmatrix} \end{array}.$$

Isto é, probabilidade de 50% de transitar e de permanecer em cada estado, para ambos os estados.

Para a série temporal é considerada uma distribuição Normal, e para cada estado o chute inicial dos parâmetros de sua respectiva Normal é a própria média e desvio padrão observados.

```r
# <code r> ================================================================= #
model_e2 <- BaumWelch(dthmm(logretorno
                            , Pi = matrix(rep(.5, 4), byrow = TRUE, nrow = 2)
                            , delta = c(0, 1)
                            , "norm"
                            , list(
                              mean = c(mean(logretorno[logest2 == 0])
                                       , mean(logretorno[logest2 == 1]))
                              , sd = c(sd(logretorno[logest2 == 0])
                                       , sd(logretorno[logest2 == 1]))))
                      , control = list(prt = FALSE
                                       , maxiter = 500
                                       , tol = 1e-05
                                       , posdiff = TRUE
```

```
                                              , converge = expression(diff < tol)))

# </code r> ================================================================== #
```

Antes de sair tirando conclusões verificamos a qualidade do ajuste na Figura 3:

```
# <code r> =================================================================== #
par(mfrow = c(1, 2)) ; plot(resid(model_e2)
                            , ylim = c(-4.25, 4.25)
                            , ylab = "Pseudo-resíduos"
                            , xlab = "Dias"
                            , las = 1
                            , main = "(a)") ; grid(col = 1) ; abline(
                              h = c(0, -1.96, 1.96, -2.58, 2.58)
                              , lty = c(1, 2, 2, 4, 4))
library(car)

qqPlot(residuals(model_e2)
       , xlab = "Quantis teóricos"
       , ylab = "Pseudo-resíduos"
       , las = 1
       , main = "(b)") ; layout(1)
# </code r> ================================================================== #
```

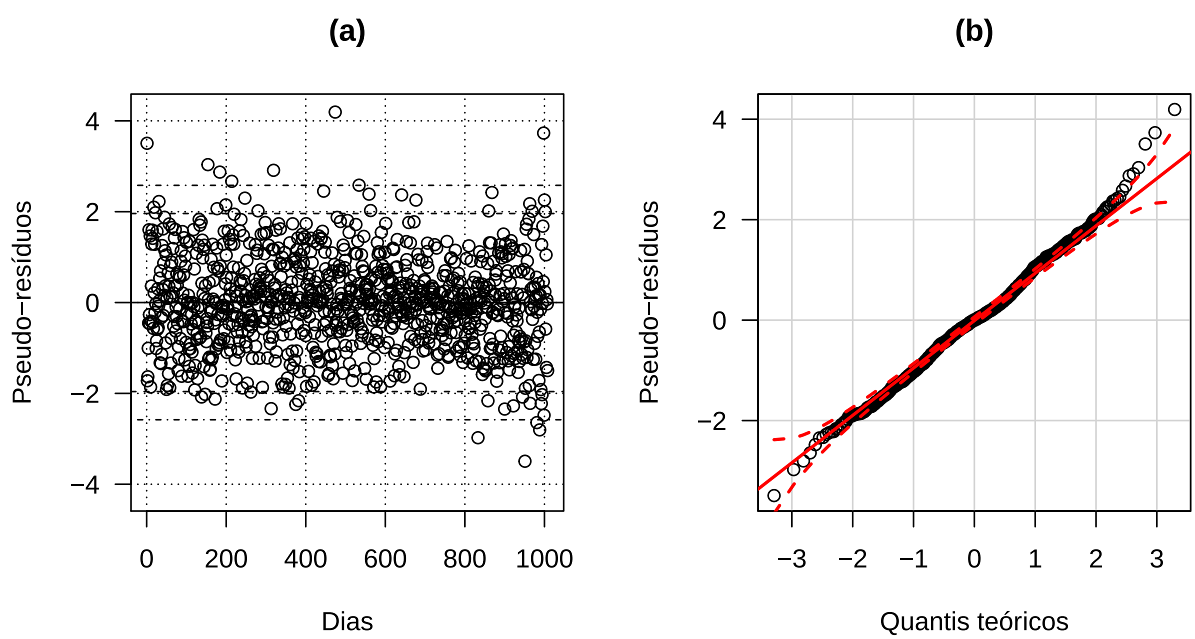

Figura 3: Análise gráfica dos (pseudo-) resíduos. (a): resíduos segundo a ordem das observações com acéscimo de linhas em zero, 1.96 (mais e menos) e 2.58 (mais e menos), seguindo a 'regra' dos três desvios padrão; (b): envelope simulado para os resíduos

Sem grandes problemas, não rejeitamos a presença de um comportamento aleatório e a normalidade dos pseudo-resíduos.

Abaixo observa-se a matriz de probabilidades de transição estimada

```
# <code r> ================================================================ #
model_e2$Pi
# </code r> =============================================================== #
```

```
          [,1]      [,2]
[1,] 0.7104082 0.2895918
[2,] 0.2242944 0.7757056
```

As probabilidades estimadas diferem bastante das utilizadas como chutes iniciais.

Estima-se maiores probabilidades de permanência, com destaque para o segundo estado (compra), que apresenta uma menor probabilidade de transição, i.e., em dias consecutivos é mais próvavel que a tomada de decisão seja a mesma, e se for pra ela mudar, é mais próvavel que ela mude pra tomada de decição venda $\rightarrow$ compra.

Para o estado venda foi atribuído como chute inicial uma média de -0.00348 e um desvio padrão de 0.00346, e foi estimada uma média de -5.6×10^{-4} e um desvio padrão de 0.00663.

Para o estado compra foi atribuído como chute inicial uma média de 0.00324 e um desvio padrão de 0.00335, e foi estimada uma média de 1.2×10^{-4} e um desvio padrão de 0.00254.

Uma vez que o modelo foi escolhido é possível fazer a predição dos estados da Cadeia de Markov.

Observa-se com a Figura 4 que para retornos próximos de zero a Cadeia está associando a tomada de decisão compra, e para valores mais distantes, venda. Contudo, pela lógica, para a tomada de decisão compra se deve ter log-retornos maiores que zero, e para a tomada de decisão venda, log-retornos menores que zero.

```
# <code r> ================================================================ #
xyplot(
  logretorno ~ 1:length(logretorno) | factor(
    Viterbi(model_e2)
    , labels = c("Vendo dólar", "Compro dólar"))
  , type = c("p", "g")
  , col = 1
  , strip = strip.custom(bg = "white")
  , scales = list(alternating = 1)
  , xlab = "Dias"
  , ylab = "log-retorno")
# </code r> =============================================================== #
```

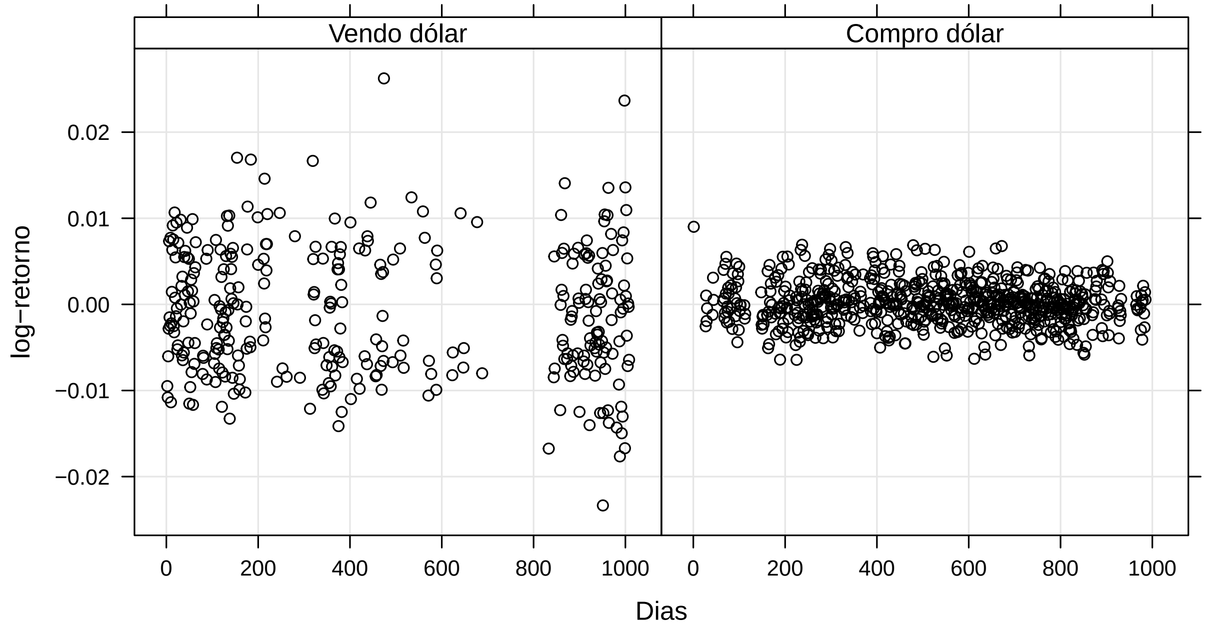

Figura 4: Predição dos dois estados da Cadeia de Markov

Diferentes chutes iniciais para a matriz de probabilidades de transição foram utilizados, entretanto as probabilidades estimadas diferiram muito pouco, levando aos mesmos resultados de predição.

Uma alternativa é aumentar o número de estados da Cadeia de Markov oculta.

Modelo de três estados

O novo estado é a tomada de decisão espera, i.e., não mexer na carteira de Euros.

```
# <code r> ================================================================ #
logest3 <- ifelse(
  logretorno >= -.0025 & logretorno <= .0025, 0, ifelse(
    logretorno < -.0025, -1, 1))

table(logest3)
# </code r> =============================================================== #
```

```
logest3
 -1   0   1
247 534 227
```

Foi usado como delimitadores pra esse novo estado log-retornos no intervalo ±0.0025.

Observa-se que dos 1008 log-retornos, 534 (53%) se encaixam nesse estado (0: espera). Com um número bem equilibrado acima e abaixo, estados venda (-1) e compra (1), respectivamente.

Com base no aprendizado obtido com o Modelo Oculto de Markov de dois estados, foi atribuído como chute inicial iguais probabilidades de transição para os estados.

```
# <code r> ================================================================ #
model_e3 <- BaumWelch(dthmm(logretorno
                            , Pi = matrix(rep(1/3, 9), byrow = TRUE, nrow = 3)
                            , delta = c(0, 0, 1)
                            , "norm"
                            , list(
                              mean = c(mean(logretorno[logest3 == 0])
                                       , mean(logretorno[logest3 == -1])
                                       , mean(logretorno[logest3 == 1]))
                              , sd = c(sd(logretorno[logest3 == 0])
                                       , sd(logretorno[logest3 == -1])
                                       , sd(logretorno[logest3 == 1]))))
                      , control = list(prt = FALSE
                                       , maxiter = 500
                                       , tol = 1e-05
                                       , posdiff = TRUE
                                       , converge = expression(diff < tol)))
# </code r> =============================================================== #
```

Com a Figura 5 observa-se um ajuste satisfatório do modelo e em seguida a matriz de probabilidades de transição estimada.

```
# <code r> ================================================================ #
par(mfrow = c(1, 2)) ; plot(resid(model_e3)
                            , ylim = c(-4.5, 4.5)
                            , ylab = "Pseudo-resíduos"
                            , xlab = "Dias"
                            , las = 1
                            , main = "(a)") ; grid(col = 1) ; abline(
                              h = c(0, -1.96, 1.96, -2.58, 2.58)
                              , lty = c(1, 2, 2, 4, 4))
qqPlot(residuals(model_e3)
       , xlab = "Quantis teóricos"
       , ylab = "Pseudo-resíduos"
       , las = 1
       , main = "(b)") ; layout(1)
# </code r> =============================================================== #
```

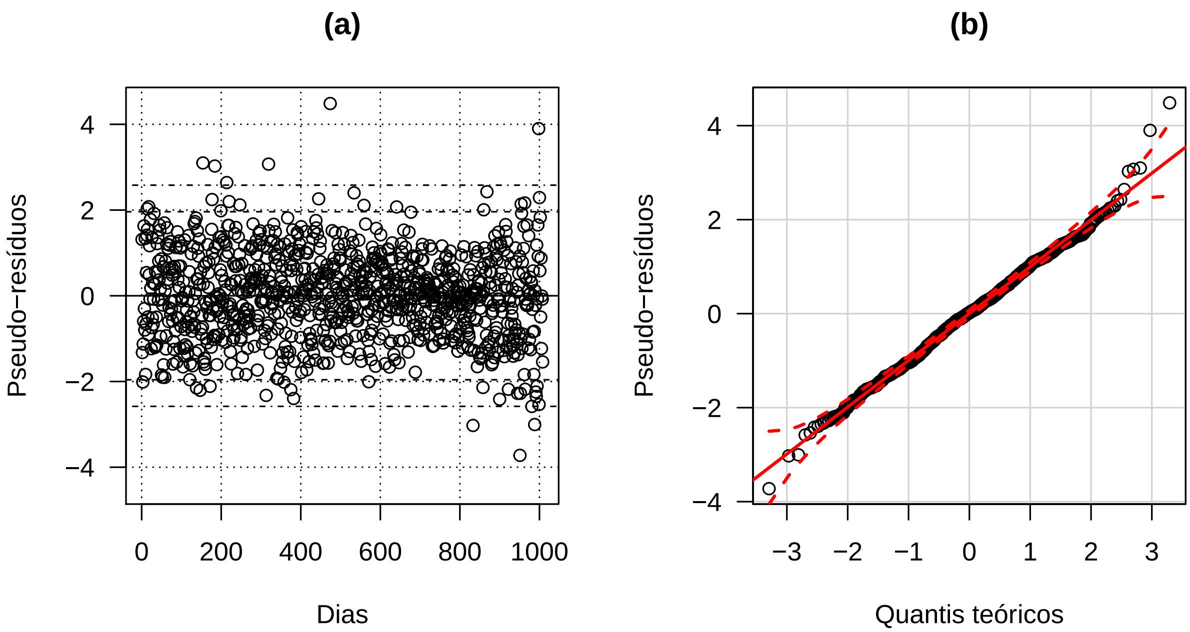

Figura 5: Análise gráfica dos (pseudo-) resíduos. (a): resíduos segundo a ordem das observações com acéscimo de linhas em zero, 1.96 (mais e menos) e 2.58 (mais e menos), seguindo a 'regra' dos três desvios padrão; (b): envelope simulado para os resíduos

```
# <code r> ================================================================= #
model_e3$Pi
# </code r> ================================================================ #
```

```
          [,1]       [,2]        [,3]
[1,] 0.5643763 0.01662133 0.419002383
[2,] 0.2861313 0.13290712 0.580961608
[3,] 0.3562025 0.63873048 0.005067058
```

A maior probabilidade de permanência é estimada para o primeiro estado, espera. A partir desse estado é mais provável transitar para o 3, compra (probabilidade 0.42).

O estado compra é o que apresenta a menor probabilidade de permanência, tendo 64% de chance de transitar para o estado venda.

Do estado venda a maior probabilidade (0.58) é de transitar para o estado compra.

Para o estado espera foi atribuído como chute inicial uma média de 0 e um desvio padrão de 0.00125, e foi estimada uma média de 10^{-4} e um desvio padrão de 0.00182.

Para o estado venda foi atribuído como chute inicial uma média de -0.00611 e um desvio padrão de 0.00332, e foi estimada uma média de -0.00251 e um desvio padrão de 0.00588.

Para o estado compra foi atribuído como chute inicial uma média de 0.00585 e um desvio padrão de 0.00335, e foi estimada uma média de 0.00149 e um desvio padrão de 0.00572.

Uma vez que o modelo foi escolhido é possível fazer a predição dos estados da Cadeia de Markov.

```
# <code r> ================================================================ #
xyplot(
  logretorno ~ 1:length(logretorno) | factor(
    Viterbi(model_e3)
    , labels = c("Espero", "Vendo dólar", "Compro dólar"))
  , type = c("p", "g")
  , col = 1
  , strip = strip.custom(bg = "white")
  , scales = list(alternating = 1)
  , xlab = "Dias"
  , ylab = "log retorno")
# </code r> =============================================================== #
```

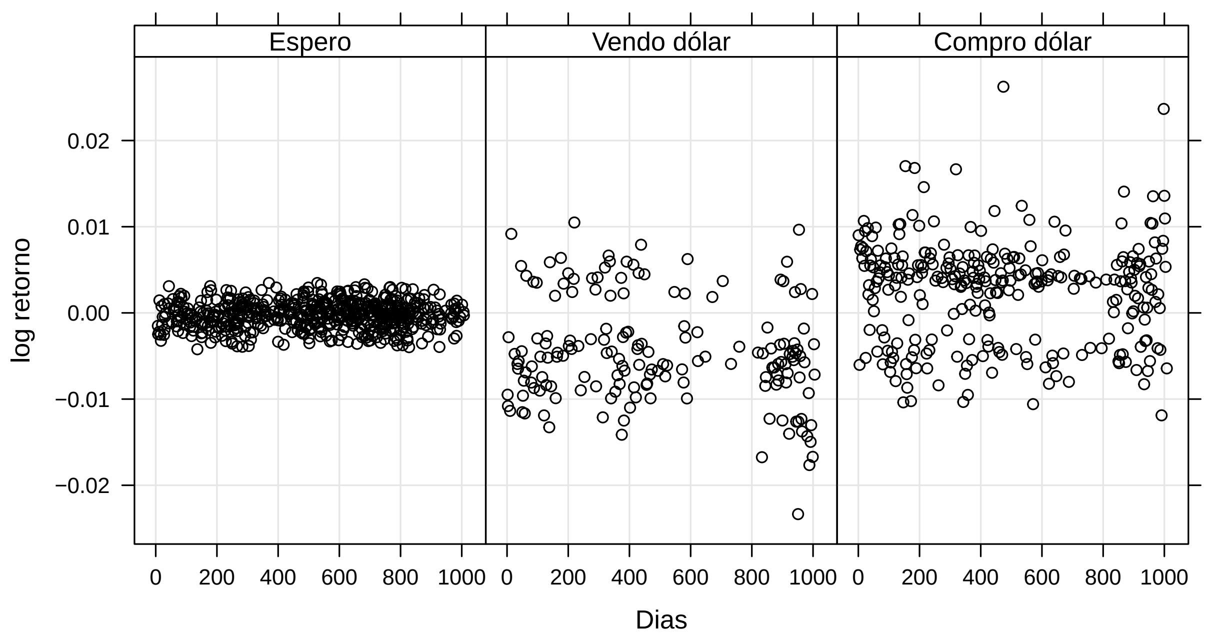

Figura 6: Predição dos três estados da Cadeia de Markov

Com a Figura 6 se observa uma maior coerência de resultados, se comparada com a Figura 4.

Aqui, para log-retornos muito próximos de zero a tomada de decisão é esperar (ok, como foi imaginado).

O ideal seria apenas log-retornos positivos para a tomada de decisão compra, e log-retornos negativos pra tomada de decisão venda, contudo, observa-se um maior número de log-retornos negativos e menores para o estado venda, e um maior número de log-retornos positivos e grandes para o estado compra.

Esses resultados não são excelentes, mas são bons e plausíveis. Talvez esse seja o melhor possível que a abordagem de Modelos Ocultos de Markov pode obter para essa série temporal.

Diferentes chutes iniciais para a matriz de probabilidades de transição foram utilizados, entretanto as probabilidades estimadas diferiram muito pouco, levando aos mesmos resultados de predição.