Lista 6
Finanças Quantitativas
Diogo Wolff Surdi

June 7, 2020

Questão 6.6

6.6.1

Para o modelo (i) temos:

$$X_t - BX_t = W_t - 1.5BW_t \implies X_t = (1-B)^{-1}(1-1.5B)W_t$$

Os polinômios são então $\phi(z) = 1 - z$ e $\theta(z) = 1 - 1.5z$.

Para o modelo (ii) temos:

$$X_t - 0.8BX_t = W_t - 0.5BW_t \implies X_t = (1-0.8B)^{-1}(1-0.5B)W_t$$

E os polinômios são $\phi(z) = 1 - 0.8z$ e $\theta(z) = 1 - 0.5z$.

6.6.2

Seguindo o critério do livro para verificar estabilidade e estacionaridade de processos ARMA, verificamos que a raiz de ϕ do processo (i) é igual a 1, enquanto que a raiz de ϕ do processo (ii) é maior que 1, logo apenas o processo (ii) é estacionário. Do mesmo modo, temos que θ de (i) é menor que 1 (em módulo) e θ de (ii) é maior do que 1, logo apenas o processo (ii) é invertível.

Questão 6.11

6.11.1

```
sample <- rnorm(5000)

process <- c(1:5000)
```

```
process[1] <- sample[1]
for (i in (2:5000)){
process[i] <- process[i-1]+sample[i]
}
```

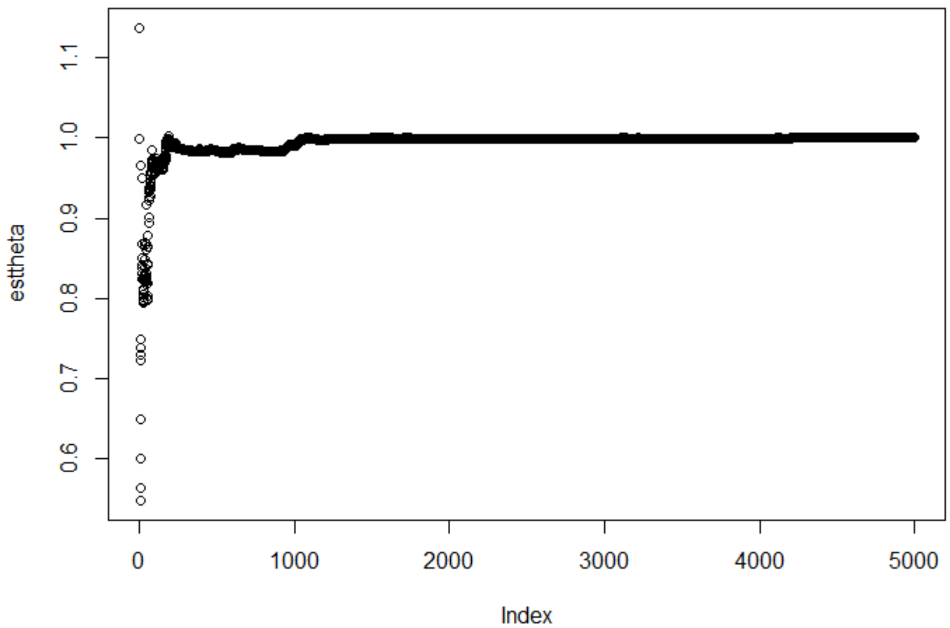
Nesse código, temos que process é a amostra da variável X_t.

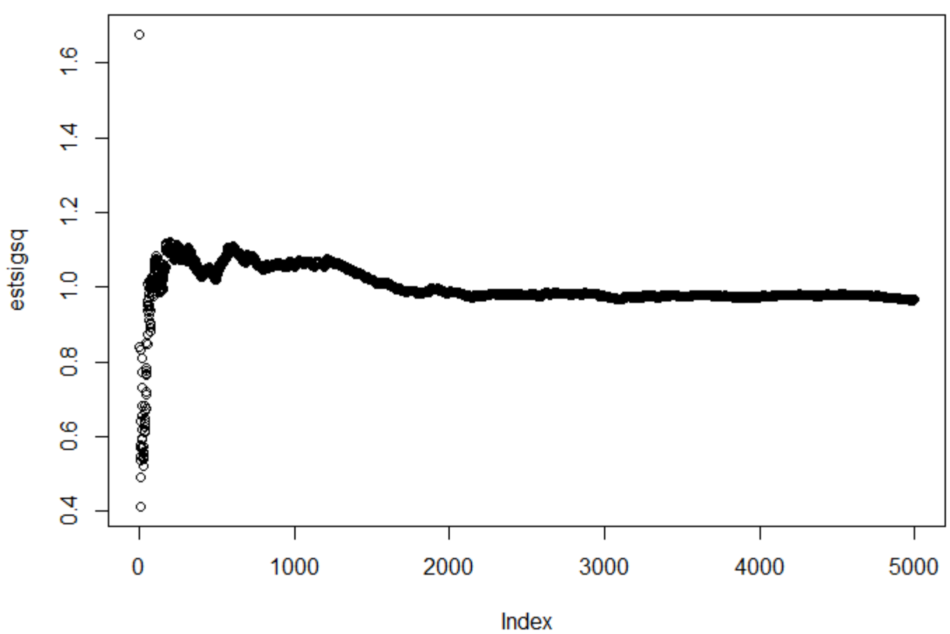
6.11.2

Devo notar que, como tomamos $x_0 = 0$, o primeiro termo acaba se anulando e devo começar a partir do segundo, para evitar dividir 0 por 0.

```
esttheta <- c(1:4999)
estsigsq <- c(1:4999)
sum <- 0
sum2 <- 0
sum3 <- 0

for (i in 2:5000){
for (j in 2:i){
sum <- sum + process[j]*process[j-1]
sum2 <- sum2 + process[j-1]^2
}
esttheta[i-1] <- sum/sum2
sum <- 0
sum2 <- 0
}

plot(esttheta)

for (i in 2:5000){
for (j in 2:i){
sum3 <- sum3 + (process[j]-esttheta[j-1]*process[j-1])^2
}
print(sum3)
estsigsq[i-1] <- sum3/(j-1)
sum3 <- 0
}

plot(estsigsq)

DF <- c(1:4999)

for (i in 1:4999){
DF[i] <- (esttheta[i]-1)/sqrt(estsigsq)
}
```

O código é bem lento (cheio de fors), mas não encontrei uma solução que fosse (muito) mais eficiente. A estatística de teste converge para 0. O plot das estimativas é:





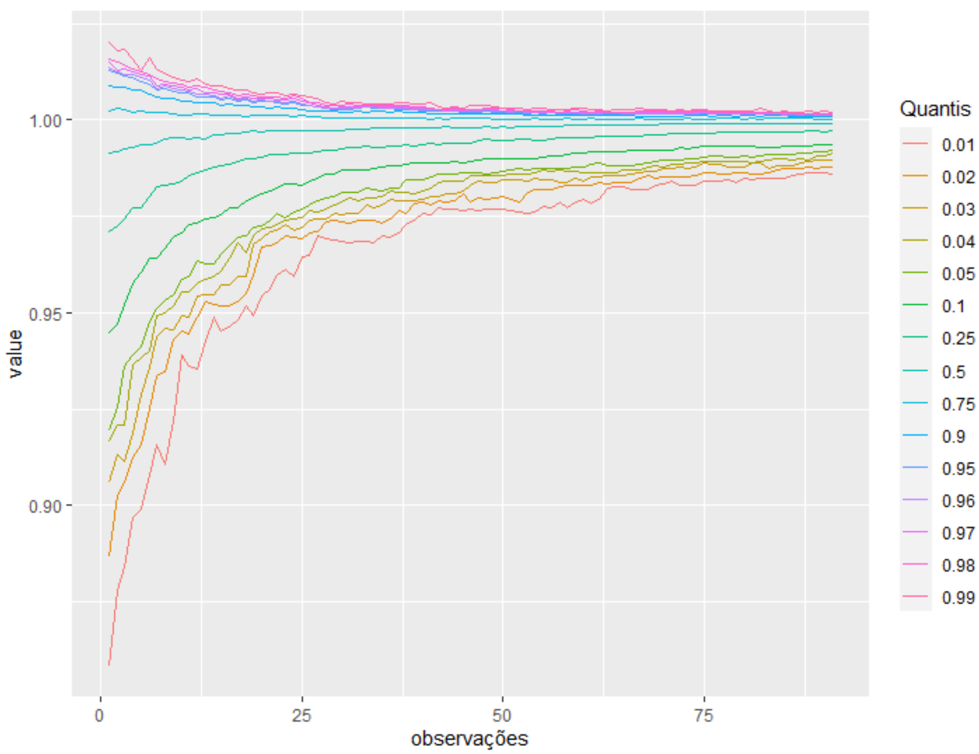
6.11.3

```r
itemc <- matrix(nrow=1000, ncol=500)*0
dummy <- c(1:1000)

for (i in 1:500){
dummy <- rnorm(1000)
itemc[1, i] <- dummy[1]
for(j in 2:1000){
itemc[j, i] <- itemc[j-1, i]+dummy[j]
}
}

XX <- matrix(nrow=999, ncol=500)*0

for (k in 1:500){
for (i in 1:999){
for (l in 2:1000){
for (j in 2:l){
sum <- sum + itemc[j,k]*itemc[j-1,k]
sum2 <- sum2 + itemc[j-1,k]^2
}
XX[l-1, k] <- sum/sum2
sum <- 0
sum2 <- 0
}
}
}

QQ <- matrix(nrow=91, ncol = 15)*0

parcial <- XX[, 1:18]

for (i in 0:89){
QQ[i+1,]<- quantile(XX[99+10*i,], probs = c(0.01, 0.02, 0.03, 0.04, 0.05,
0.1, 0.25, 0.5, 0.75, 0.9,
0.95, 0.96, 0.97, 0.98, 0.99),
na.rm=TRUE)
}

library(ggplot2)
library(reshape2)

suporte <- data.frame(matrix(nrow=91,ncol=16))
suporte[,2:16]<-QQ
suporte[,1]<-c(1:91)
colnames(suporte) <- c('observacoes', '0.01', '0.02', '0.03', '0.04',
```

```
'0.05', '0.1', '0.25', '0.5', '0.75',
'0.9', '0.95', '0.96', '0.97', '0.98', '0.99')

suporte <- melt(suporte, id.vars='observacoes', variable.name='Quantis')

ggplot(suporte, aes(observacoes,value)) + geom_line(aes(colour = Quantis))
```

Como pode ver, esse código é extremamente ineficiente. Após umas 11 horas rodando, consegui gerar a matriz XX, e o resultado está no gráfico do item seguinte. Novamente,tive problemas no primeiro índice, e meu código é entre 99 e 999.

6.11.4



6.11.5

```
cpn <- read.csv('(...)/CPN Historical Data.csv')
cpn$lret <- NA
cpn$lret[1] <- 0
for (i in 2:1258){
cpn$lret[i] <- log(cpn$Price[i]/cpn$Price[i-1])
}

library(tseries)

> adf.test(cpn$Price)
```

```
Augmented Dickey-Fuller Test

data:  cpn$Price
Dickey-Fuller = -4.6105, Lag order = 10, p-value = 0.01
alternative hypothesis: stationary

> adf.test(cpn$lret)

Augmented Dickey-Fuller Test

data:  cpn$lret
Dickey-Fuller = -11.647, Lag order = 10, p-value = 0.01
alternative hypothesis: stationary
```

Como pode-se ver, o p-valor é 0.01 em ambas, então o correto seria rejeitar a hipótese nesse nível (porém não sei se esse valor é arredondado, ele pode ser algo como 0.009, caso onde não rejeitaríamos, por exemplo). A questão fala que os quantis seriam proxy para a estatística, porém não vi onde utilizá-los na questão.

Questão 6.13

6.13.1

Sendo μ o drift, temos $X_{t-1} = \mu + X_{t-2} + W_{t-1}$. Então, seguindo indutivamente, encontramos que:

$$\begin{aligned} X_t &= \mu + (\mu + X_{t-2} + W_{t-1}) + W_t \\ &= 2\mu + (\mu + X_{t-3} + W + t - 2) + W_{t-1} + W_t \\ &\vdots \\ &= t\mu + X_{t-t} + W_1 + ... + W_t \\ &= t\mu + x_0 + \sum_{i=1}^{t} W_i \end{aligned}$$

6.13.2

Temos:

$$\mathbb{E}[X_t] = \mathbb{E}[x_0 + \mu t + \sum_{i=1}^{t} W_i] = \mu t + n\mathbb{E}[W_t] = \mu t$$

E também:

$$Var(X_t) = \sum_{i=1}^{t} Var(W_i) = t\sigma^2$$

Seja então $\phi_X(t_1, t_2) = cov(X_{t1}, X_{t2})$. Temos que:

$$\phi(t, t) = \mathbb{E}[(X_t - \mu_t)^2] = Var(X_t) = t\sigma^2$$

E também:

$$\begin{aligned}\phi(t_1,t_2) &= \mathbb{E}[X_{t1},X_{t2}] - \mathbb{E}[X_{t1}]\mathbb{E}[X_{t2}] \\ &= \mathbb{E}[X_{t1},X_{t2}] - \mu t_1 \mu t_2 \\ &= \mathbb{E}\left[\left(x_0 + \mu t_1 + \sum_{i=1}^{t_1} W_i\right)\left(x_0 + \mu t_2 + \sum_{i=1}^{t_2} W_i\right)\right] - \mu^2 t_1 t_2\end{aligned}$$

6.13.3

O processo X_t tem raízes complexas de módulo menor igual a 1, logo não é estacionário.

6.13.4

Temos que:

$$X_t = x_0 + \mu t + \sum_{i=1}^{t} W_i$$

E:

$$X_{t-1} = x_0 + \mu(t-1) + \sum_{i=1}^{t-1} W_i$$

Logo ao subtrair uma da outra encontramos:

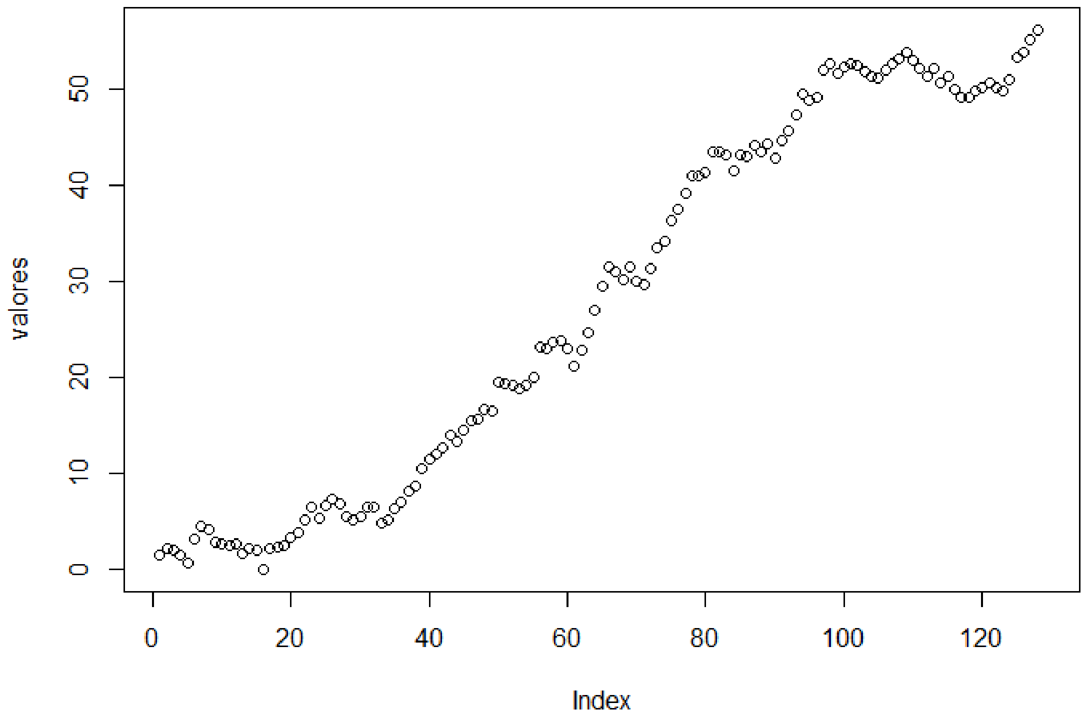
$$X_t - X_{t-1} = \mu + W_t$$

6.13.5

O processo tem drift, logo não é estacionário.

6.13.6

```
x0 <- 1.5
mu <- 0.5
n <- 128

noise <- rnorm(n)

valores <- c(1:129)
valores[1] <- x0
for (i in 2:n+1){
valores[i] <- valores[i-1]+mu+noise[i]
}
```


valores
0
10
20
30
40
50
0
20
40
60
80
100
120
Index