Finanças Quantitativas: Lista 5

Lucas Moschen

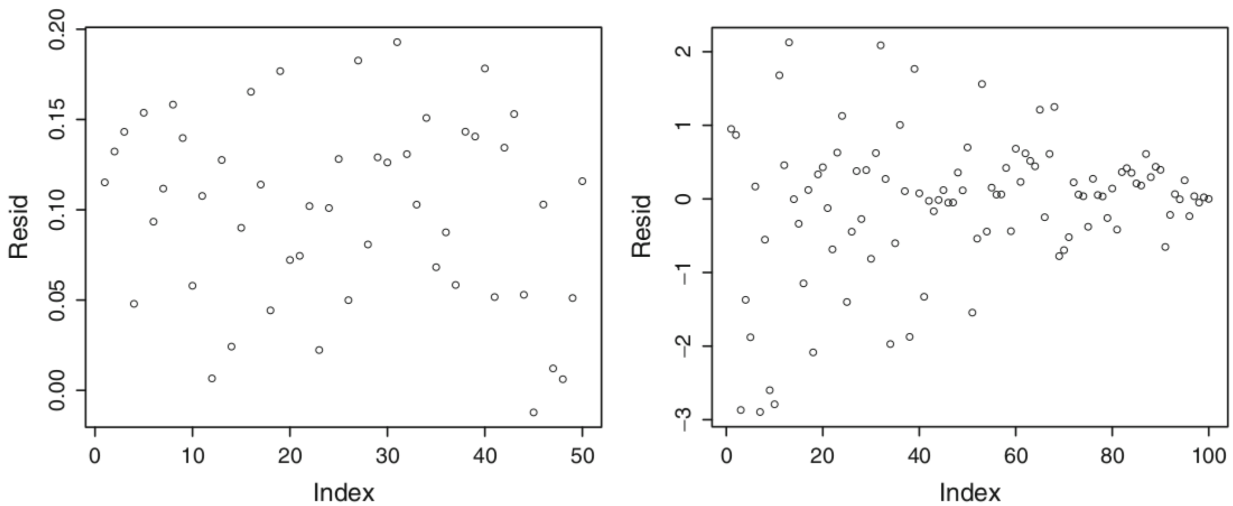
24 de maio de 2020

Exercício 1

Problema 4.1 (Carmona)

Item 1

Considere a seguinte imagem do livro.



Podemos observar que o plote da esquerda tem o eixo X marcado pelo índice de determinadas observações e o eixo Y valores, quase todos, estritamente positivos (com excessão aparente de um). Isso não é a característica de resíduos brutos, onde $\hat{\epsilon}_i = y - \hat{y}_i$, dado que a média desse plot é de longe 0. Como temos uma quantidade relativamente grande de observações e esperamos que os erros tenham média 0 ao modelarmos o erro quadrático, vemos que não é possível que esse plote represente isso.

Item 2

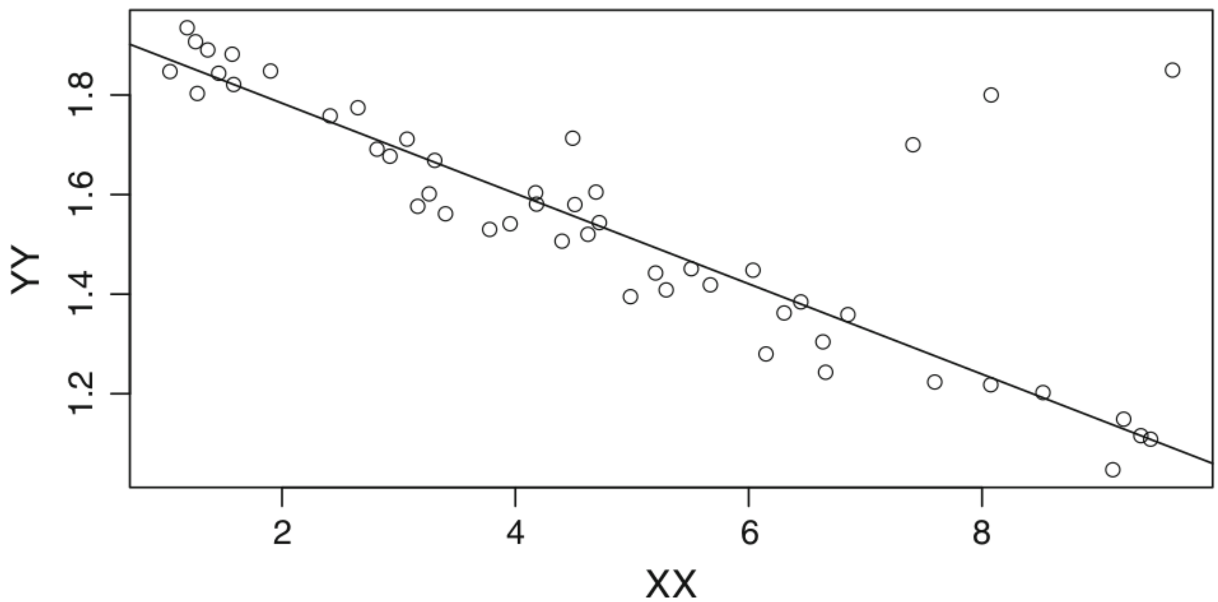
Ainda sobre a figura acima, obseve que quanto maior o índice, mais os resíduos se concentram em torno de 0. Como eles se concentram, podemos dizer, com base no gráfico que quanto maior o índice i, menor a variância em torno da média dos resíduos. Conhecemos nesse capítulo que os resíduos tem desvio padrão

$$\sigma_{\epsilon_i} = \sigma\sqrt{1 - h_{i,i}}$$

Assim, $h_{i,i}$ tem um comportamento crescente em relação ao índice, isto é, a diagonal tem termos em uma sequência crescente.

Item 3

Agora considere essa imagem:



Parece-me que essa linha é uma **regressão linear com desvios absolutos**, visto que ela não se influencia tanto com os três *outliers* na parte de cima do gráfico. Isto acontece, pois a regressão $\mathcal{L}_1$ é menos sensível a *outliers*.

Problema 4.11 (Carmona)

Neste exercício, pretendemos analizar um exemplo de regressão não linear. Os dados são de uma droga *Puromycin* e possui uma tabela com três variáveis de um experimento biomédico em células, tratadas ou não. Denotamos y como a velocidade inicial da reação, enquanto x é a concentração da enzima. Pela relação de Michaelis-Menten, onde V_a é a velocidade assintótica e K uma constante:

$$y = \phi(x) = V_a \frac{x}{x + K}$$

Itens 1 e 2

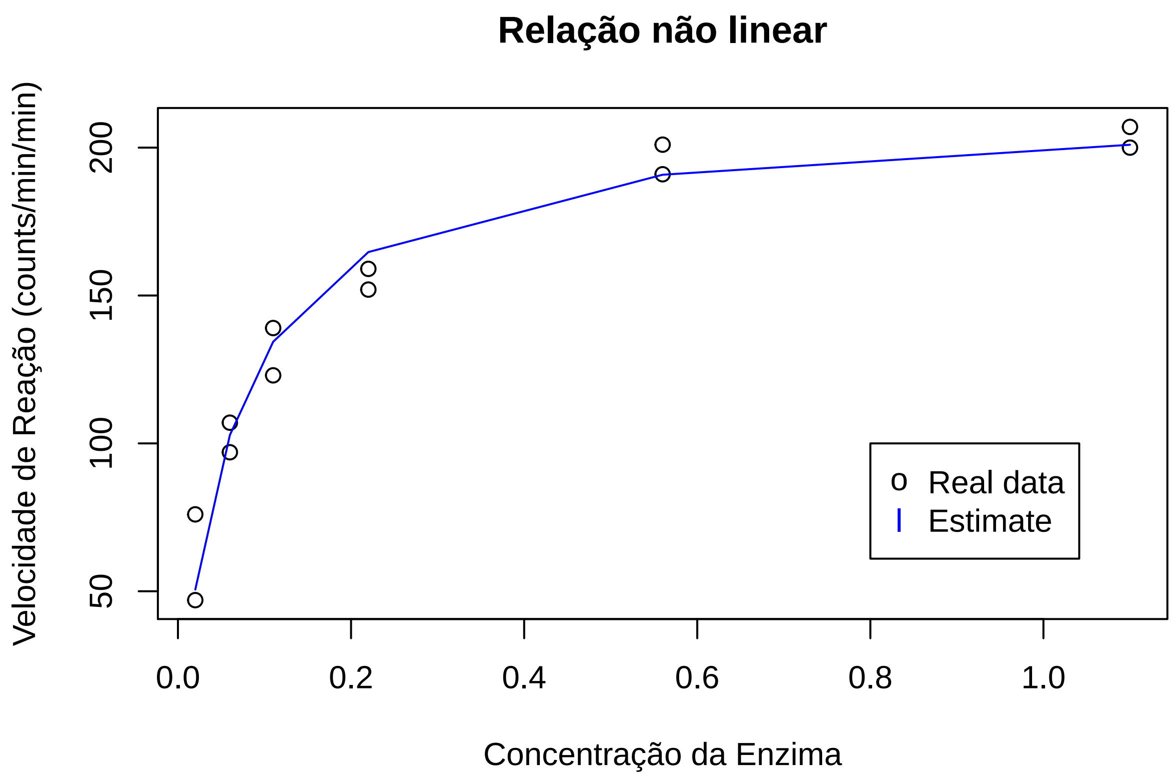
```
data("Puromycin", Puromycin)
attach(Puromycin)
y = Puromycin[state == "treated",]$rate
x = Puromycin[state == "treated",]$conc
plot(x,y, xlab = "Concentração da Enzima", ylab = "Velocidade de Reação (counts/min/min)",
    main = "Relação não linear")

y_fit = nls(y ~ Va*(x/(x + K)), start = c(Va = 200, K = 0.1))
print("These are the estimated parameters:")
```

```
## [1] "These are the estimated parameters:"
```

```
print(coef(y_fit))
```

```
##          Va           K
## 212.6836296   0.0641211
```

```
lines(x, fitted(y_fit), col = "blue")

legend(0.8, 100, legend = c("Real data","Estimate"),
        pch = c("o","l"), col = c("black", "blue"))
```



Problema 4.12 (Carmona)

Nesse exercício, usadmos a família generalizada Vasicek para parametrizar o termo estrutural da taxa de juros. Definimos $Y_{GV}(x,\theta)$ como:

$$Y_{GC}(x,\theta) = \theta_1 - \theta_2\theta_4\frac{1-e^{-x/\theta_4}}{x} + \theta_3\theta_4\frac{(1-e^{-x/\theta_4})^2}{4x}$$

Item 1

Sabemos que $f(t,T) = -\frac{d}{dT}\log P(t,T)$ (4.38) e $Y(t,T) = -\frac{1}{T-t}\log P(t,T)$. Juntando ambas as equações, obtemos que $f(t,T) = \frac{d}{dT}(T-t)Y(t,T)$. Observe que no livro, a equação 4.40 está com sinal trocado, por essa observação. Outra forma de visualizar essa troca de sinais é tirando o logaritmo da equação 4.39, onde $\tau = T - t$. Também uso como referência, *esse artigo*.

Nesse caso,

$$\begin{aligned}
f(x,\theta) &= \frac{d}{dx}xY(x,\theta)\\
&= Y(x,\theta) + x\frac{d}{dx}Y(x,\theta)\\
&= \theta_1 - \theta_2\theta_4\frac{1-e^{-x/\theta_4}}{x} + \theta_3\theta_4\frac{(1-e^{-x/\theta_4})^2}{4x} + \\
&+ x\left(-\theta_2\theta_4\frac{\frac{x}{\theta_4}e^{-x/\theta_4} - (1-e^{-x/\theta_4})}{x^2} + \theta_3\theta_4\frac{8\frac{x}{\theta_4}(1-e^{-x/\theta_4})e^{-x/\theta_4} - 4(1-e^{-x/\theta_4})^2}{16x^2}\right)\\
&= \theta_1 - \theta_2\theta_4\frac{1-e^{-x/\theta_4}}{x} + \theta_3\theta_4\frac{(1-e^{-x/\theta_4})^2}{4x} - \theta_2\theta_4\frac{\frac{x}{\theta_4}e^{-x/\theta_4} - (1-e^{-x/\theta_4})}{x} + \\
&+ \theta_3\theta_4\frac{2\frac{x}{\theta_4}(1-e^{-x/\theta_4})e^{-x/\theta_4} - (1-e^{-x/\theta_4})^2}{4x}\\
&= \theta_1 - \theta_2\frac{xe^{-x/\theta_4}}{x} + \theta_3\frac{2x(1-e^{-x/\theta_4})e^{-x/\theta_4}}{4x}\\
&= \theta_1 - \theta_2 e^{-x/\theta_4} + \frac{1}{2}\theta_3(1-e^{-x/\theta_4})e^{-x/\theta_4}\\
&= \theta_1 - \theta_2 e^{-x/\theta_4} + \frac{1}{2}\theta_3(1-e^{-x/\theta_4})e^{-x/\theta_4}\\
&= \theta_1 - (\theta_2 - \frac{1}{2}\theta_3(1-e^{-x/\theta_4}))e^{-x/\theta_4}
\end{aligned}$$

Agora considere essa função escrita em R:

```r
fgv <- function(x, theta){
  value <- theta[1] - (theta[2] - (1/2)*theta[3]*(1 -  exp(-x/theta[4])))*exp(-x/theta[4])
  return(value)
}
```

Interpretação dos parâmetros:

Assumimos que $\theta_4 > 0$. Nesse caso, observamos que quando $x \to \infty$, $e^{-x/\theta_4} \to 0$, logo θ_1, como no caso da família Nelson-Siegel, representa o valor assintótico da função f. Quando $x = 0$ e obtemos o valor da função no ponto inicial é dado por $\theta_1 - \theta_2$. Logo $\theta_1 > \max\{\theta_2, 0\}$. Quando derivamos essa expressão, obtemos:

$$f'(x,\theta) = (\theta_2 + \theta_3(e^{-x/\theta_4} - \frac{1}{2}))\frac{e^{-x/\theta_4}}{\theta_4} \tag{1}$$

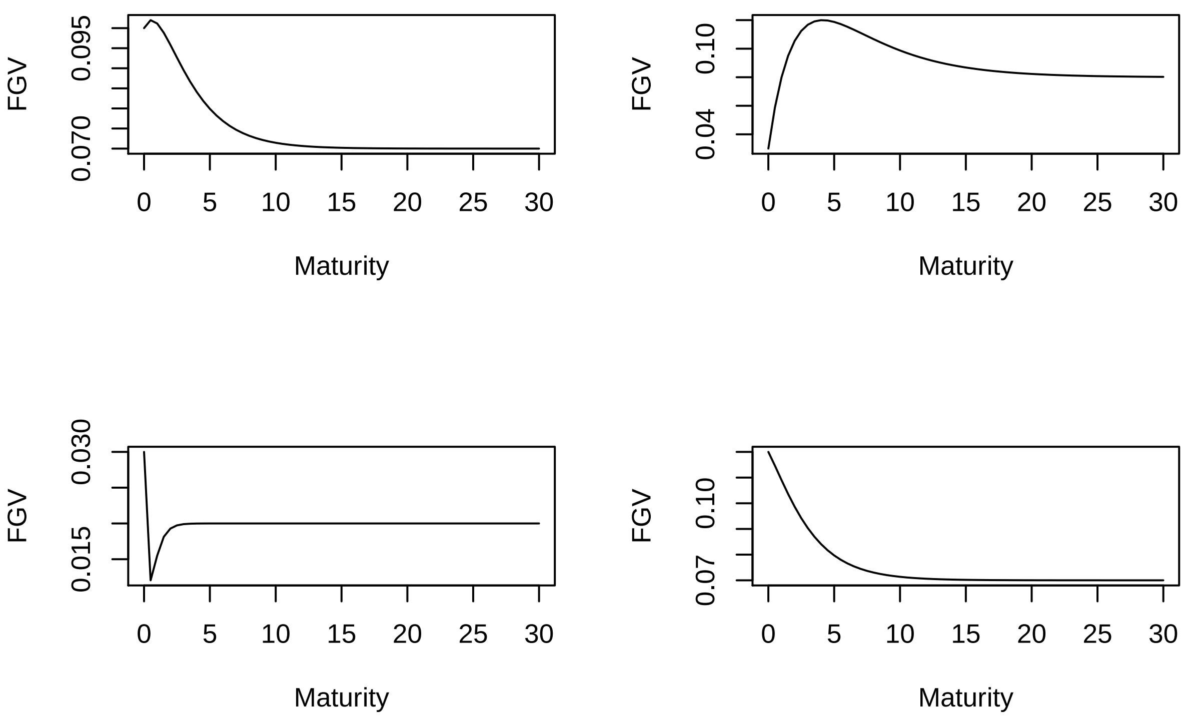
$$f''(x,\theta) = -(\theta_2 + \theta_3(2e^{-x/\theta_4} - \frac{1}{2}))\frac{e^{-x/\theta_4}}{\theta_4^2} \tag{2}$$

Nesse caso, se $-\frac{\theta_2}{\theta_3} + \frac{1}{2} > 0,\ i.e. \frac{1}{2} > \frac{\theta_2}{\theta_3}$ há um pico que pode ser *hump* ou *dip* dependendo de θ_3. Observando a equação 2, vemos que no ponto crítico, se $\theta_3 > 0$, a expressão é negativa, logo temos um minimo local(*dip*). Caso contrário (excetuando o caso em que $\theta_3 = 0$), temos um máximo local(*hump*). Além disso, θ_4 está relacionado com a localização do pico.

A magnitude do pico também está relacionada com esses parâmetros. Em especial, quanto maior $theta_4$, menor a função derivada é e menor será o pico (em magnitude).

Plots

```r
par(mfrow = c(2,2))
x <- seq(from = 0, to = 30, by = 0.5)
# Exemplo 1
theta = c(0.07, -0.03, 0.1, 2.5)
plot(x,fgv(x, theta), type = "l", xlab = "Maturity", ylab = "FGV")
# Exemplo 2: Observe o aumento dos parâmetros e o pico da curva
theta = c(0.08, 0.05, 0.5, 4.5)
plot(x,fgv(x, theta), type = "l", xlab = "Maturity", ylab = "FGV")
# Exemplo 3: Agora theta3 é negativo e theta4 bem pequeno
theta = c(0.02, -0.01, -0.1, 0.5)
plot(x,fgv(x, theta), type = "l", xlab = "Maturity", ylab = "FGV")
# Exemplo 4: Nesse caso, como não respeitamos a relação citada, não há pico.
theta = c(0.07, -0.05, 0.05, 2.5)
plot(x,fgv(x, theta), type = "l", xlab = "Maturity", ylab = "FGV")
```



Item 2

Podemos extrair essa fórmula por $P(x) = e^{-xY(x,\theta)}$.

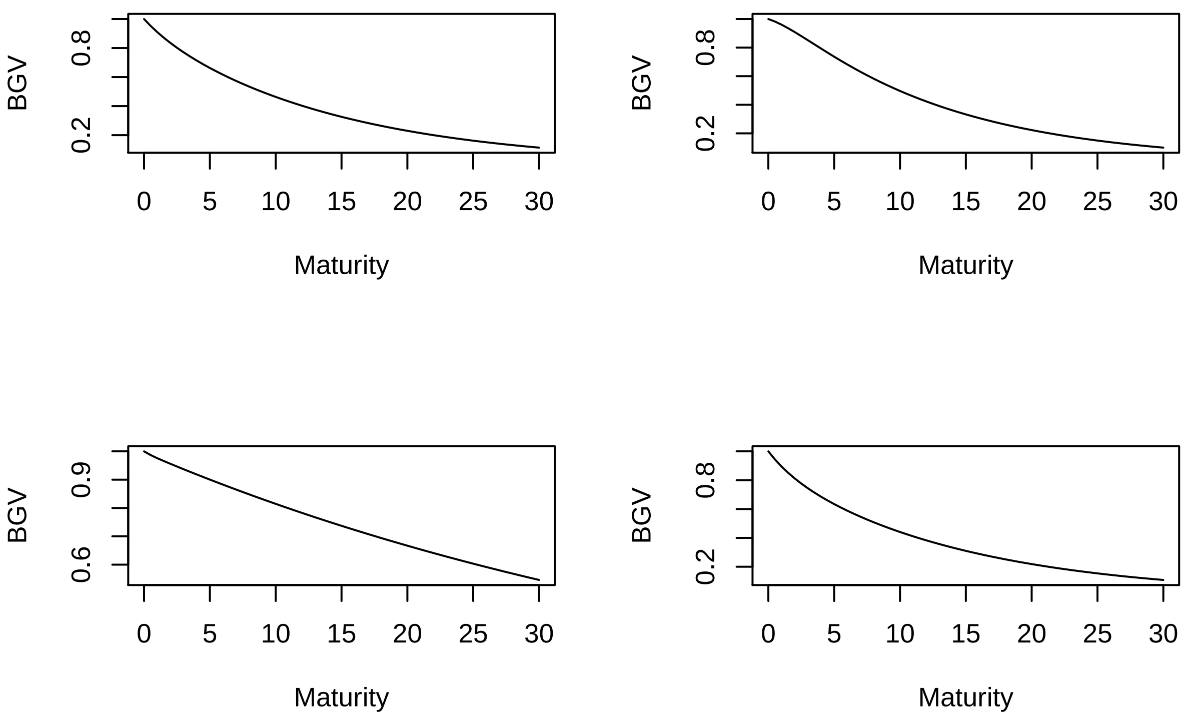
$$P(x,\theta) = \exp\{-\theta_1 x + \theta_2\theta_4(1 - e^{-x/\theta_4}) - \theta_3\theta_4\frac{(1 - e^{-x/\theta_4})^2}{4}\}$$

Agora considere a função R:

```
bgv <- function(x,theta){
  value <- (-theta[1])*x + theta[2]*theta[4]*(1 - exp(-x/theta[4]) - (1/4)*theta[3]*theta[4]*(1 - exp(-
  return(exp(value))
}
```

Plots

```
par(mfrow = c(2,2))
x <- seq(from = 0, to = 30, by = 0.5)
# Exemplo 1
theta = c(0.07, -0.03, 0.1, 2.5)
plot(x,bgv(x, theta), type = "l", xlab = "Maturity", ylab = "BGV")
# Exemplo 2: Observe o aumento dos parâmetros e o pico da curva
theta = c(0.08, 0.05, 0.5, 4.5)
plot(x,bgv(x, theta), type = "l", xlab = "Maturity", ylab = "BGV")
# Exemplo 3: Agora theta3 é negativo e theta4 bem pequeno
theta = c(0.02, -0.01, -0.1, 0.5)
plot(x,bgv(x, theta), type = "l", xlab = "Maturity", ylab = "BGV")
# Exemplo 4: Nesse caso, como não respeitamos a relação citada, não há pico.
theta = c(0.07, -0.05, 0.05, 2.5)
plot(x,bgv(x, theta), type = "l", xlab = "Maturity", ylab = "BGV")
```



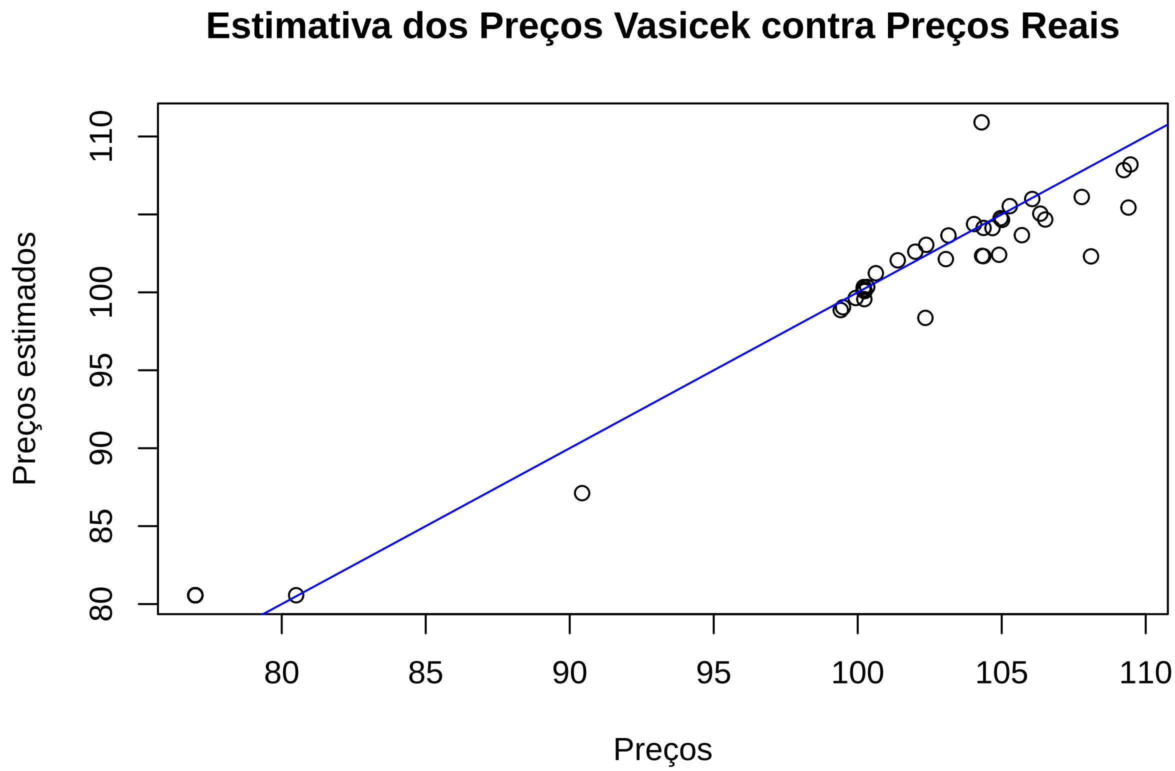
Item 3

```
library(Rsafd)
# Importando os dados
data("GermanB041700", GermanB041700, package = "Rsafd")
```

```r
# Precisamos calcular Bj(theta)
BondPrice <- function(theta){
  Coupon <- GermanB041700$Coupon
  AI <- GermanB041700$Accrud.Intrst
  Maturity <- GermanB041700$Life
  LL <- floor(1 + Maturity)

  PriceEstimated <- rep(0, length(Coupon))
  for(i in 1:length(Coupon)){
    x <- seq(to=Maturity[i], by = 1, length = LL[i])
    Discount <- bgv(x, theta = theta)
    CF <- rep(Coupon[i], LL[i])
    CF[LL[i]] <- 100 + CF[LL[i]]
    PriceEstimated[i] <- sum(CF*Discount)
  }
  return(PriceEstimated - AI)
}

# Função de perda
BondVf <- function(theta){

  PriceReal <- GermanB041700$Price
  error = sum((PriceReal - BondPrice(theta))^2)
  return(error)

}

Vf.fit <- optim(par = c(0.05, -0.03, 0.1, 2.0),
                fn = BondVf, gr = NULL, method = "L-BFGS-B",
                lower = c(0, -Inf, -Inf, 0), upper = c(Inf, Inf, Inf, Inf))

print(Vf.fit)
```

```
## $par
## [1] 0.05847472 0.01455796 0.10061879 1.99987815
##
## $value
## [1] 281.3055
##
## $counts
## function gradient
##       24       24
##
## $convergence
## [1] 0
##
## $message
## [1] "CONVERGENCE: REL_REDUCTION_OF_F <= FACTR*EPSMCH"
```

```r
plot(GermanB041700$Price, BondPrice(Vf.fit$par),
     main = "Estimativa dos Preços Vasicek contra Preços Reais",
     xlab = "Preços", ylab = "Preços estimados")
abline(a = 0,b = 1, col = "blue")
```


Estimativa dos Preços Vasicek contra Preços Reais
Preços estimados
80
85
90
95
100
110
80
85
90
95
100
105
110
Preços

Exercício 2

Considere o modelo de regressão linear múltipla:

$$\boldsymbol{Y} = \boldsymbol{X}\beta + \epsilon$$

E o custo associado a β, que queremos minimizar, é:

$$\begin{aligned}\mathcal{L}_2(\beta) &= ||\boldsymbol{Y} - \boldsymbol{X}\beta||^2 = \langle \boldsymbol{Y} - \boldsymbol{X}\beta, \boldsymbol{Y} - \boldsymbol{X}\beta \rangle \\ &= \langle \boldsymbol{Y}, \boldsymbol{Y} \rangle - 2\langle \boldsymbol{Y}, \boldsymbol{X}\beta \rangle + \langle \boldsymbol{X}\beta, \boldsymbol{X}\beta \rangle \\ &= \langle \boldsymbol{Y}, \boldsymbol{Y} \rangle - 2\langle \boldsymbol{X}\beta, \boldsymbol{Y} \rangle + \langle \boldsymbol{X}\beta, \boldsymbol{X}\beta \rangle \\ &= \boldsymbol{Y}^T\boldsymbol{Y} - 2\beta^T\boldsymbol{X}^T\boldsymbol{Y} + \beta^T\boldsymbol{X}^T\boldsymbol{X}\beta\end{aligned}$$

Para minimizar esse valor, primeiro procuramos os pontos críticos. Nesse caso, veja que a primeira expressão independe do vetor β e, portanto, sua derivada será 0. A segunda expressão, temos uma combinação dos elementos do vetor β, logo, ao derivar parcialmente em relação a cada valor, obtemos a expressão equivalente e, por isso, a derivação é linear. Na última expressão, veja que $\boldsymbol{X}^T\boldsymbol{X}$ é uma expressão com os valores de β quadráticos e, portanto, a expressão se segue:

$$\frac{d}{d\beta}\mathcal{L}_2(\hat{\beta}) = -2\boldsymbol{X}^T\boldsymbol{Y} + 2\boldsymbol{X}^T\boldsymbol{X}\hat{\beta} = 0$$

Desta forma, como o posto de $\boldsymbol{X}$ é completo $\boldsymbol{X}^T\boldsymbol{X}$ tem colunas linearmente independentes e é, portanto, invertível. Desta maneita, temos uma solução única e:

$$\boldsymbol{X}^T\boldsymbol{X}\hat{\beta} = \boldsymbol{X}^T\boldsymbol{Y} \implies \hat{\beta} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{Y}$$

Nesse obtemos que $\hat{\beta}$ é um ponto crítico. Para averiguar se é argumento mínimo, façamos:

$$\frac{d^2}{d\beta^2}\mathcal{L}_2(\hat{\beta}) = 2\boldsymbol{X}^T\boldsymbol{X}$$

A matriz $\boldsymbol{X}^T\boldsymbol{X}$ é simétrica e, $\forall x \in \mathbb{R}^{p+1}, x^T\boldsymbol{X}^T\boldsymbol{X}x = \langle \boldsymbol{X}x, \boldsymbol{X}x \rangle \geq 0$ e será igual a 0 somente se $\boldsymbol{X}x = 0$. Como X tem posto completo, ele tem espaço anulado com dimensão 0 e, poranto, $anul(X) = \{0\}$ e, se $\boldsymbol{X}x = 0, x = 0$. Concluo, então que essa matriz é estritamente positiva e, desta maneira, $\hat{\beta}$ é de fato um mínimo da expressão.

Exercício 3

```r
library(BatchGetSymbols) # get financial data
```

Item a

Esses são os *trading symbols* das componentes que integram a Ibovespa, segundo a página oficial.

```r
df.ibov <- GetIbovStocks()
print(df.ibov$tickers)
```

```
##  [1] "ABEV3"  "AZUL4"  "B3SA3"  "BBAS3"  "BBDC3"  "BBDC4"  "BBSE3"  "BEEF3"
##  [9] "BPAC11" "BRAP4"  "BRDT3"  "BRFS3"  "BRKM5"  "BRML3"  "BTOW3"  "CCRO3"
## [17] "CIEL3"  "CMIG4"  "COGN3"  "CPFE3"  "CRFB3"  "CSAN3"  "CSNA3"  "CVCB3"
## [25] "CYRE3"  "ECOR3"  "EGIE3"  "ELET3"  "ELET6"  "EMBR3"  "ENBR3"  "ENGI11"
## [33] "EQTL3"  "FLRY3"  "GGBR4"  "GNDI3"  "GOAU4"  "GOLL4"  "HAPV3"  "HGTX3"
## [41] "HYPE3"  "IGTA3"  "IRBR3"  "ITSA4"  "ITUB4"  "JBSS3"  "KLBN11" "LAME4"
## [49] "LREN3"  "MGLU3"  "MRFG3"  "MRVE3"  "MULT3"  "NTCO3"  "PCAR3"  "PETR3"
## [57] "PETR4"  "QUAL3"  "RADL3"  "RAIL3"  "RENT3"  "SANB11" "SBSP3"  "SULA11"
## [65] "SUZB3"  "TAEE11" "TIMP3"  "TOTS3"  "UGPA3"  "USIM5"  "VALE3"  "VIVT4"
## [73] "VVAR3"  "WEGE3"  "YDUQ3"
```

Item b

Tomo os dados históricos do período de 01/01/2019 até o dia de 31/12/2019 cada uma das 74 ações da Ibovespa. Daquelas que printei acima e integram atualmente a bolsa, as ações da "NTCO3.SA" não possuiam atividade nesse período. Por simplicidade e para analisar um tempo mais complexo, eu a retiro das analisadas. Além disso, eu integro as ações em uma *dataframe* que conterá também a informação dos dados do índice da Ibovespa.

```r
first.date <- "2019-01-01"
last.date <- "2019-12-31"

df <- BatchGetSymbols("^BVSP", first.date = first.date, last.date = last.date)
stocks <- data.frame(BVSP = df$df.tickers$price.close)

for (ticker in df.ibov$tickers) {
  tickerSA = paste(ticker, ".SA", sep = "")
  df <- BatchGetSymbols(tickerSA, first.date =   first.date, last.date = last.date)
  if(length(stocks$BVSP) != length(df$df.tickers$price.close)){
    next
  }
  stocks <- cbind(stocks, df$df.tickers$price.close)
}

# Removing NA values
stocks <- stocks[complete.cases(stocks),]
```

Item c

Suponho que o portfolio de mercado seja dado pelo índice da Ibovespa e considero a taxa de juros uma média anual das taxas diárias da taxa SELIC do ano de 2019. É aproximadamente a taxa ao longo do ano dividida pelo número de dias. Assim:

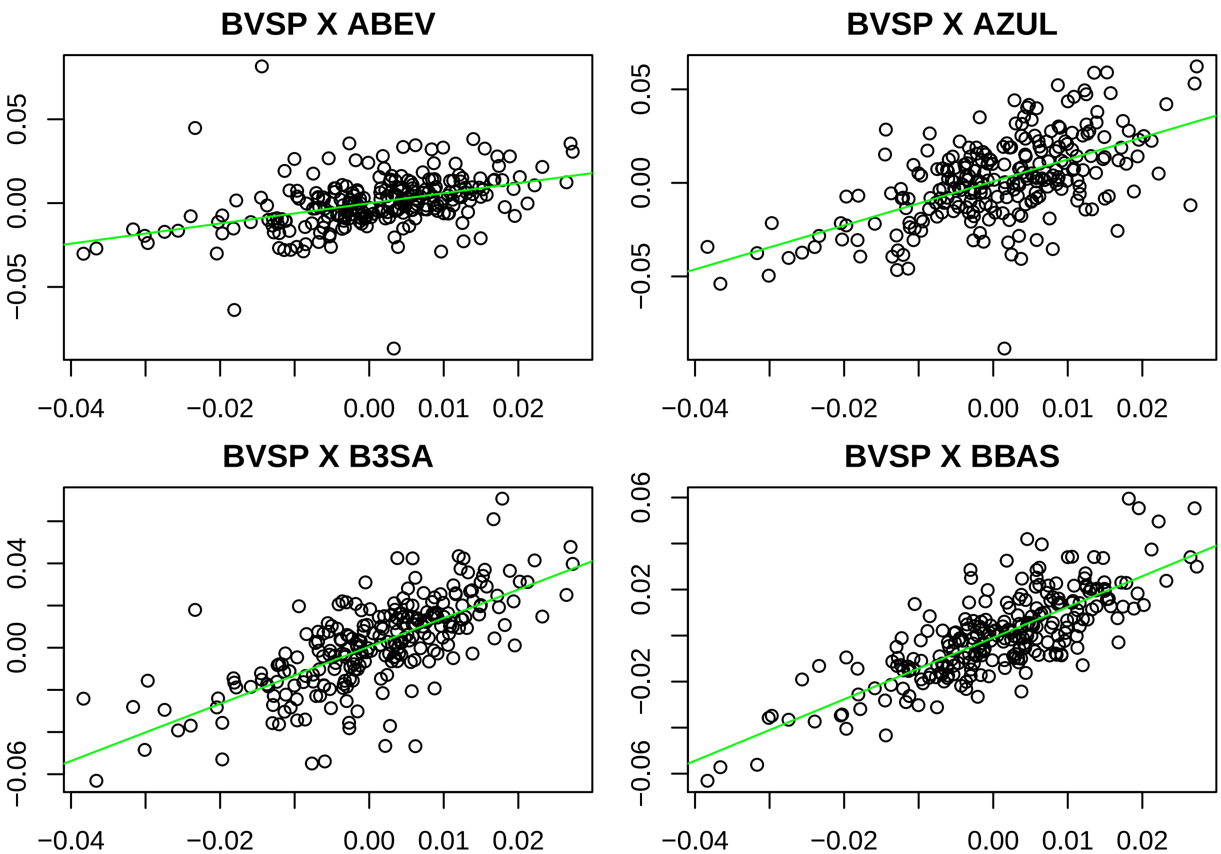
```r
# Primeiro, vamos calcular os log-retornos de cada ação.
logstocks <- data.frame(diff(log(as.matrix(stocks))))
```

```r
# Taxa de juros diária
r = 0.0002288414
logstocks <- logstocks - r

# Renomear corretamente
names(logstocks) <- append(c("BVSP"), df.ibov$tickers[-c(54)])
```

```r
# Regressão Linear de Mínimos Quadrados
coefficients <- data.frame(BVSP = c("Intercept","Slope"))

# Avaliações para o item d
r2 <- rep(0, length(names(logstocks)))
p.value.intercept <- rep(0, length((names(logstocks))))
mean.residual <- rep(0,length(logstocks))
median.residual <- rep(0,length(logstocks))

for(i in 2:length(names(logstocks))){
  model <- lsfit(logstocks$BVSP, logstocks[names(logstocks)[i]])
  r2[i] = as.numeric(ls.print(model, print.it = FALSE)$summary[2])
  p.value.intercept[i] <- data.frame(ls.print(model, print.it = FALSE)$coef.table)$Pr...t..[1]
  mean.residual[i] = mean(model$residuals)
  median.residual[i] = median(model$residuals)
  coefficients[names(logstocks)[i]] = model$coefficients
}

# Destaco os coeficientes estimados para as 4 primeiras ações:
print(coefficients[,1:6])
```

```
##        BVSP         ABEV3       AZUL4        B3SA3         BBAS3         BBDC3
## 1 Intercept -9.413385e-05 0.000803734 0.0005182878 -0.0009010482 -0.0005850791
## 2     Slope  6.014182e-01 1.175318353 1.3550593394  1.3367636766  1.1896241611
```

```r
# Plotting dessas quatro primeiras ações, cada uma em seu respectivo eixo y contra o retorno de mercado
par(mfrow = c(2,2), mar = c(2,2,2,1))

# Plotting dos gráficos
plot(logstocks$BVSP, logstocks$ABEV3, main = "BVSP X ABEV")
abline(a = coefficients[1,2], b = coefficients[2,2], col = "green")
plot(logstocks$BVSP, logstocks$AZUL4, main = "BVSP X AZUL")
abline(a = coefficients[1,3], b = coefficients[2,3], col = "green")
plot(logstocks$BVSP, logstocks$B3SA3, main = "BVSP X B3SA")
abline(a = coefficients[1,4], b = coefficients[2,4], col = "green")
plot(logstocks$BVSP, logstocks$BBAS3, main = "BVSP X BBAS")
abline(a = coefficients[1,5], b = coefficients[2,5], col = "green")
```

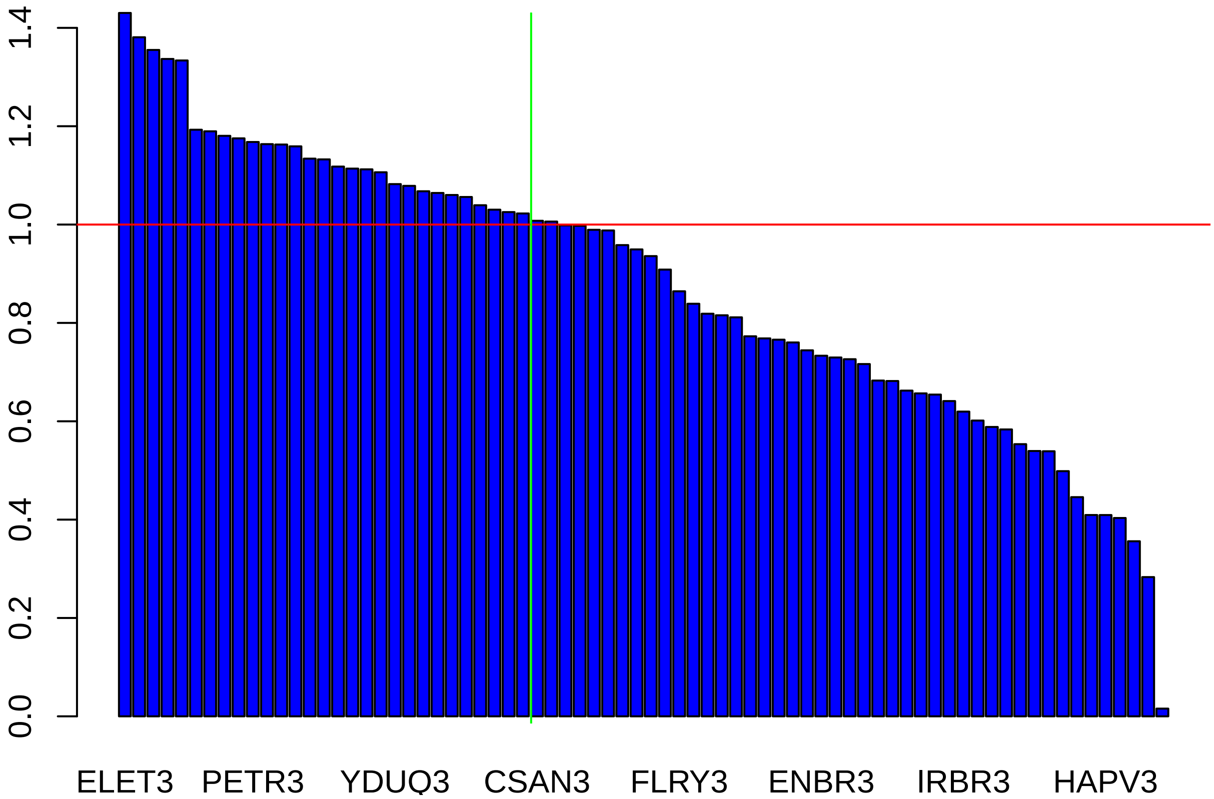


É interessante que visualmente já percebemos que os interceptos são muito próximos a 0, como esperávamos. De fato, a média do valor absoluto é mostrada na figura. Também vemos como de fato os dados tem um comportamento linear. Além disso, aprendemos que a estimativa da inclinação da reta ($\hat{\beta}_j$) para cada ativo é conhecida como investimento beta e mede a sensitividade do retorno com as variações dos retornos do portifólio de mercado. Assim, quando esse valor é maior do que 1, temos ativos com mais risco. Para visualizar, considere o seguinte gráfico, ele indica as diferenças de cada dessa estimativa para cada ativo.

```r
# Valor médio absoluto do intercepto
print(mean(abs(as.matrix(coefficients[1,-1]))))
```

```
## [1] 0.001023665
```

```r
par(mar = c(2,2,2,1))
barplot(as.matrix(sort(coefficients[2,-1], decreasing = TRUE)), col = "blue")
abline(h = 1.0, col = "red")
abline(v = 35.0, col = "green")
```



Nesse gráfico, podemos perceber que aproximadamente 40% dos ativos tem investimento beta maior do que 1. Além disso, um ativo demonstrou a estimativa próxima a 0, o que indica possível atuação de *outliers*.

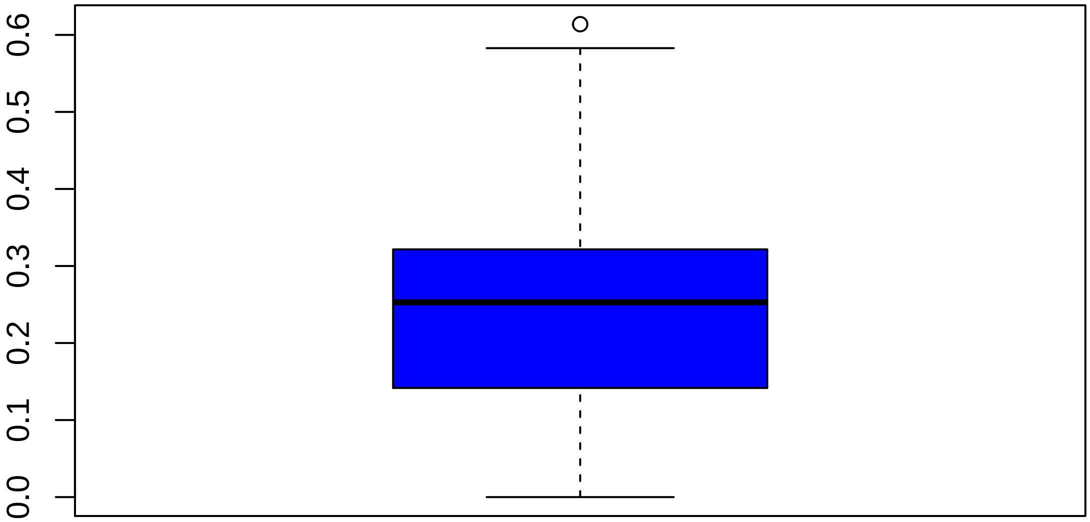
Item d

Hipóteses

1. Existência significativa de uma relação linear.

Observe o BoxPLot gerado com os valores do R_2. Apesar de nas figuras, haver uma tendência linear dos dados, existe muito ruído e a mediana dos dados é em torno de 0.25. Nesse caso, não conseguimos extrair completamente a relação linear dos dados com o nosso modelo.

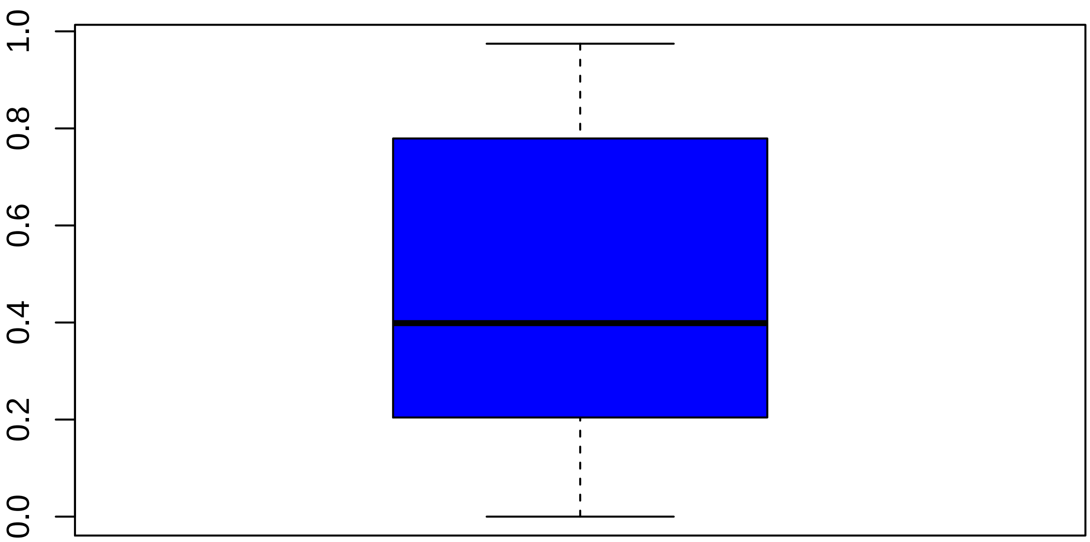
```
boxplot(r2, col = "blue")
```



2. Um intercepto 0.

Observe boxplot do t-teste realizado e o comportamento do p-valor. Apenas 8 ativos tiveram o p-valor inferior a 0.05. Logo, na maioria dos casos, não é possível rejeitar a hipótese nula de que o intercepto é 0.

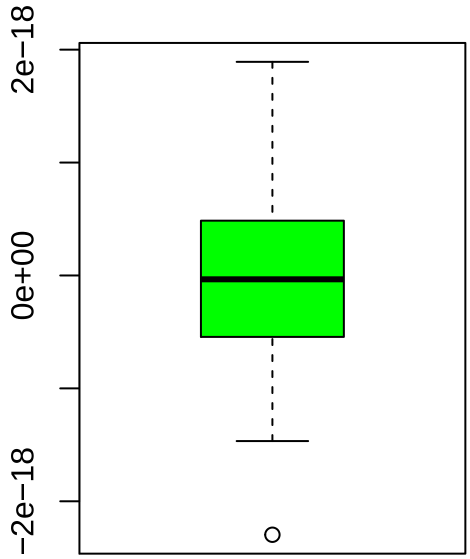
```
boxplot(p.value.intercept, col = "blue")
```

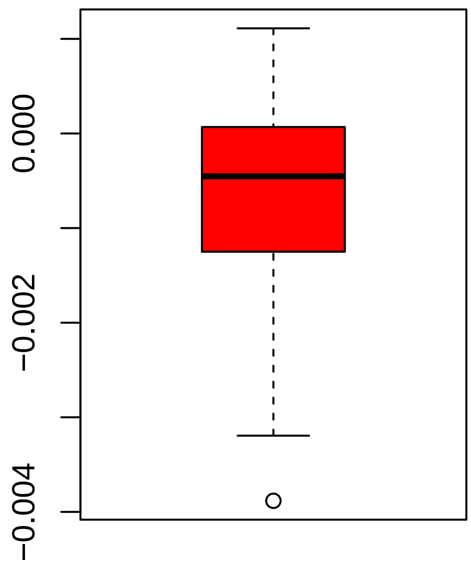


3. Erros normalmente distribuidos e não correlacionados.

Para isso, vamos aplicar dois conhecimentos do livro. O primeiro é analisar a média e mediana dos resíduos. Observe que a média das médias é 0, como esperávamos Além disso, a mediana também tem um comportamento interessante, próximo de 0. Isso já nos mostra que a média 0 é atingida.

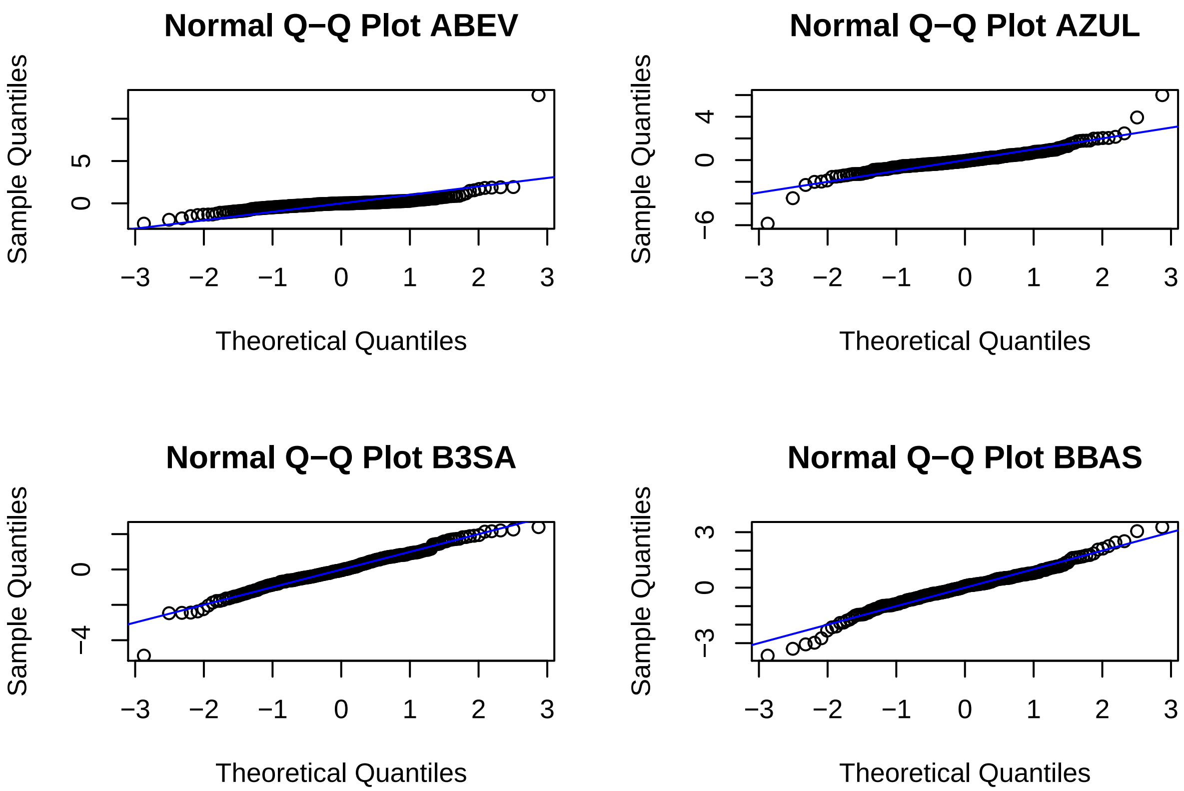
```
par(mfrow = c(1,2))
boxplot(mean.residual, col = "green")
boxplot(median.residual, col = "red")
```





Para a segunda análise, proposta pelo livro, façamos novamente os plots das quatro primeiras ações da bolsa que foram mostrados anteriormente. Mas agora, analisaremos os resíduos em um qqnorm. Com excessão de alguns poucos *outliers*, podemos inferir que a distribuição dos erros é de fato normal.

```
model1 <- lsfit(logstocks$BVSP, logstocks[names(logstocks)[1]])
model2 <- lsfit(logstocks$BVSP, logstocks[names(logstocks)[2]])
model3 <- lsfit(logstocks$BVSP, logstocks[names(logstocks)[3]])
model4 <- lsfit(logstocks$BVSP, logstocks[names(logstocks)[4]])

model1.diag <- ls.diag(model1)
model2.diag <- ls.diag(model2)
model3.diag <- ls.diag(model3)
model4.diag <- ls.diag(model4)

par(mfrow = c(2,2))
qqnorm(model1.diag$std.res, main = "Normal Q-Q Plot ABEV")
abline(0,1, col = "blue")
qqnorm(model2.diag$std.res, main = "Normal Q-Q Plot AZUL")
abline(0,1, col = "blue")
qqnorm(model3.diag$std.res, main = "Normal Q-Q Plot B3SA")
abline(0,1, col = "blue")
qqnorm(model4.diag$std.res, main = "Normal Q-Q Plot BBAS")
abline(0,1, col = "blue")
```



4. Estabilidade.

Para isso, vamos utilizar os dados do ano de 2018 para checar se há constância do parâmetro ao longo do tempo. Como notado pelo livro-texto, esse é um grande problema que será abordado mais para frent, em outros capítulos. Ademais, farei esse teste.

Item e