

Lista_01

Marcelo Neves Lira

Carregando bibliotecas

```
library(quantmod)
library(fBasics)
library(tidyverse)
```

1.

a)

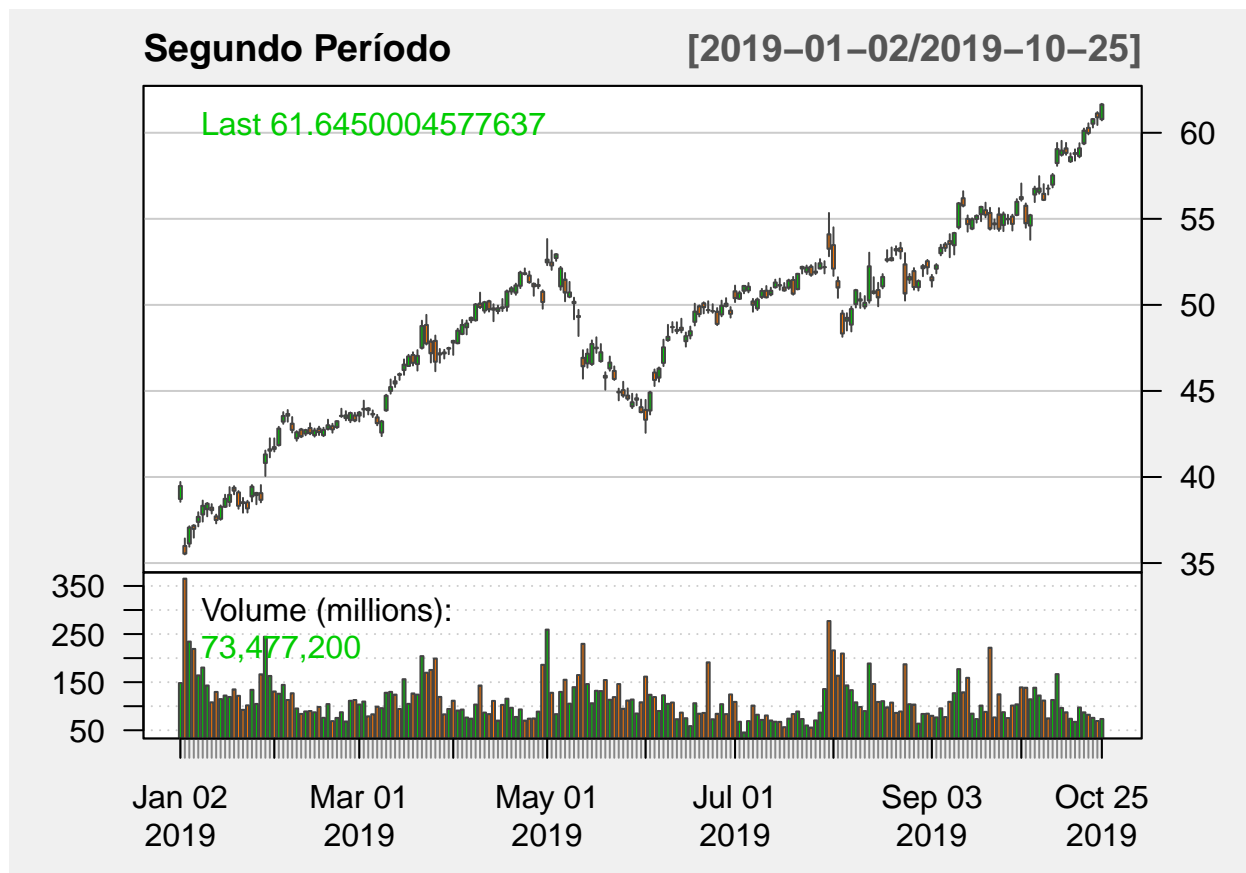
```
getSymbols("AAPL", from = "2018-01-01", to = "2018-12-31")
t1 <- AAPL # Período referente ao ano de 2018
getSymbols("AAPL", from = "2019-01-01", to = "2019-10-28")
t2 <- AAPL # Período referente ao ano de 2019
```

Ação escolhida foi a Apple. Ticker “AAPL”.

t1 e t2 são os dois períodos solicitados

b)

```
chartSeries(t2, theme = "white", name = "Segundo Período")
```



Na parte superior do gráfico de *candlestick*, estão as informações sobre as cotações (máximo, mínimo, fechamento, abertura) da Apple de janeiro a outubro de 2019. Na parte inferior, estão as informações sobre o volume de negociação diário.

c)

Primeiro período (janeiro a dezembro de 2018)

```
plot(Cl(t1), main = "Preço Fechamento")
```

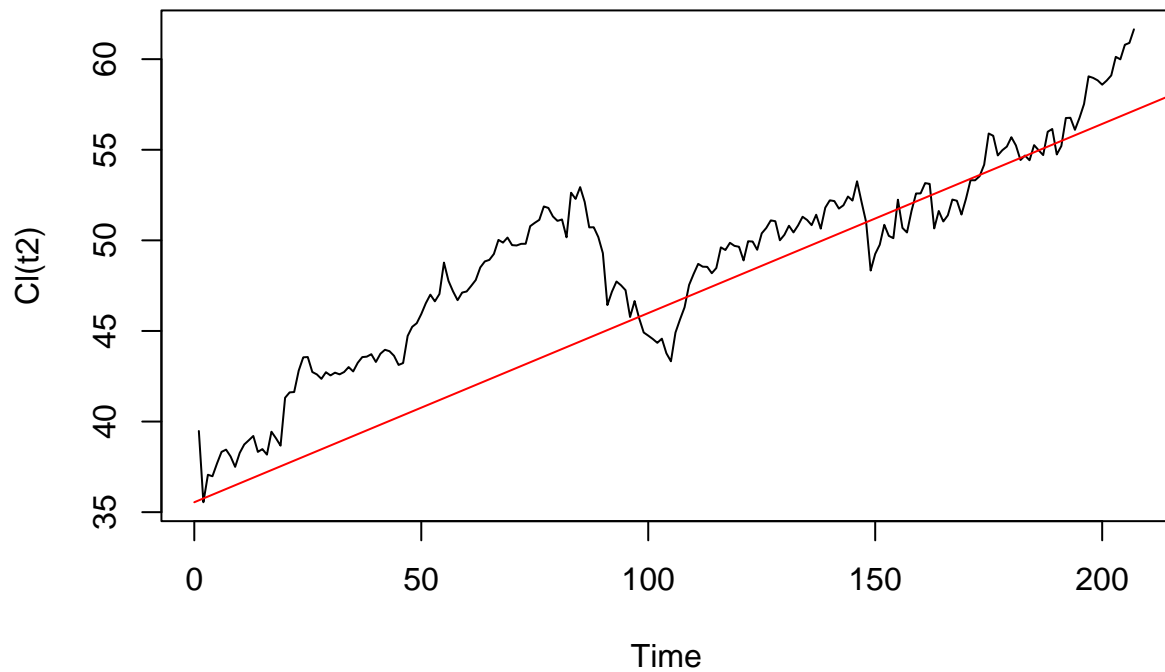


No início do período, as condições de estacionariedade fraca (i) e (ii) parecem valer. Depois disso, não parecem razoáveis de serem aceitas.

Segundo período (janeiro a outubro de 2019)

```
plot.ts(Cl(t2), main = "Preço Fechamento")
lines(c(0, 250), c(min(Cl(t2)), max(Cl(t2))), col = "red")
```

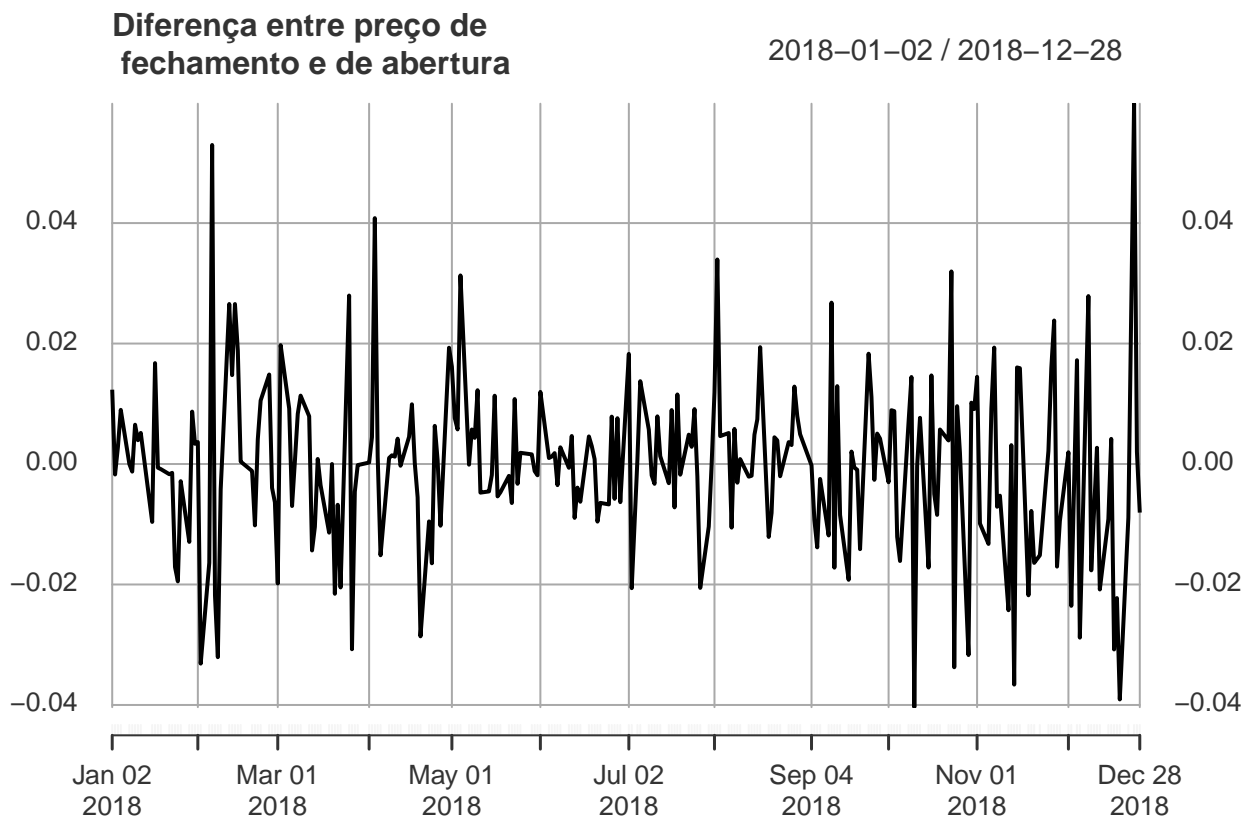
Preço Fechamento



Essa série temporal apresenta um comportamento mais determinístico, que poderia ser estimado por uma reta. Definitivamente, não apresenta média e variância invariantes no tempo.

d)

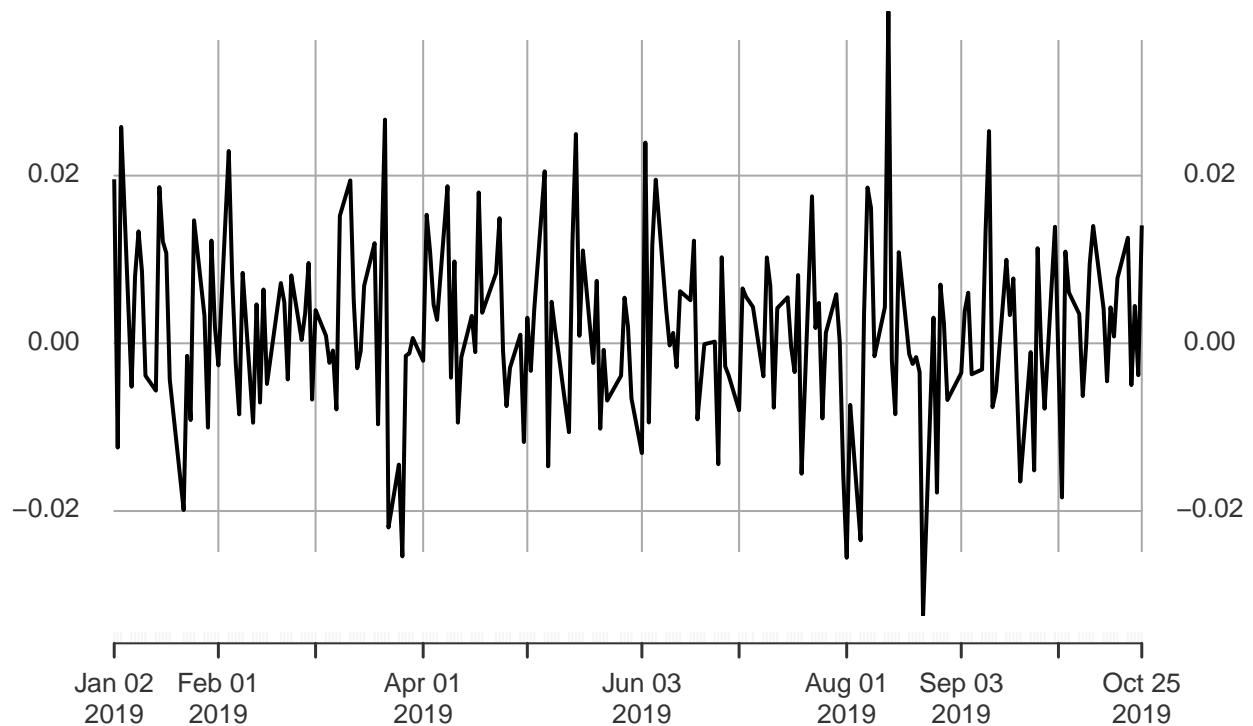
```
plot(OpCl(t1),  
      main = "Diferença entre preço de \n fechamento e de abertura")
```



```
plot(OpCl(t2),
      main = "Diferença entre preço de \n fechamento e de abertura")
```

Diferença entre preço de fechamento e de abertura

2019-01-02 / 2019-10-25



Ambas as séries agora apresentam características de um processo estacionário faco, com $E(x_t) = \mu$ e $Var(x_t) < +\infty$. Seria necessário observar o comportamento de decaimento dos lags pela FACV para confirmar.

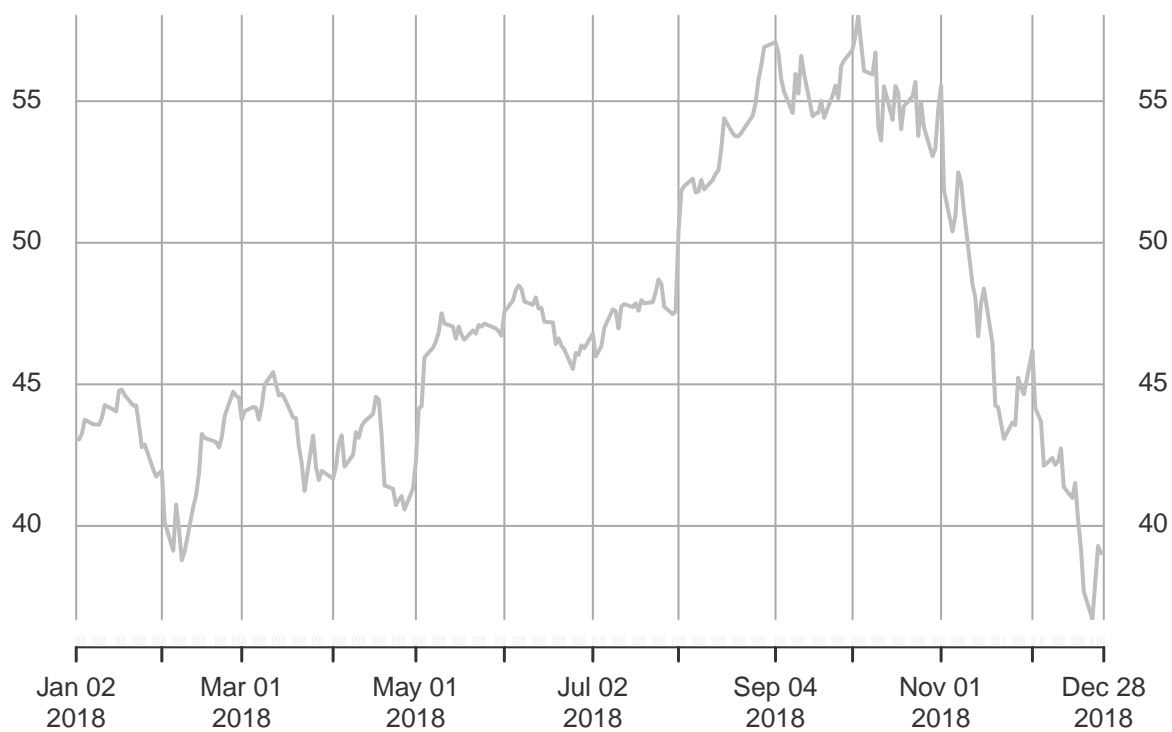
e)

Primeiro período

```
mm_aapl_5 <- rollmean(Cl(t1), k = 5, align = c("right"))
mm_aapl_15 <- rollmean(Cl(t1), k = 15, align = c("right"))
mm_aapl_30 <- rollmean(Cl(t1), k = 30, align = c("right"))
plot(Cl(t1), col = "gray", main = "Médias Móveis")
```

Médias Móveis

2018-01-02 / 2018-12-28



```
lines(mm_aapl_5, col = 5)
```

Médias Móveis

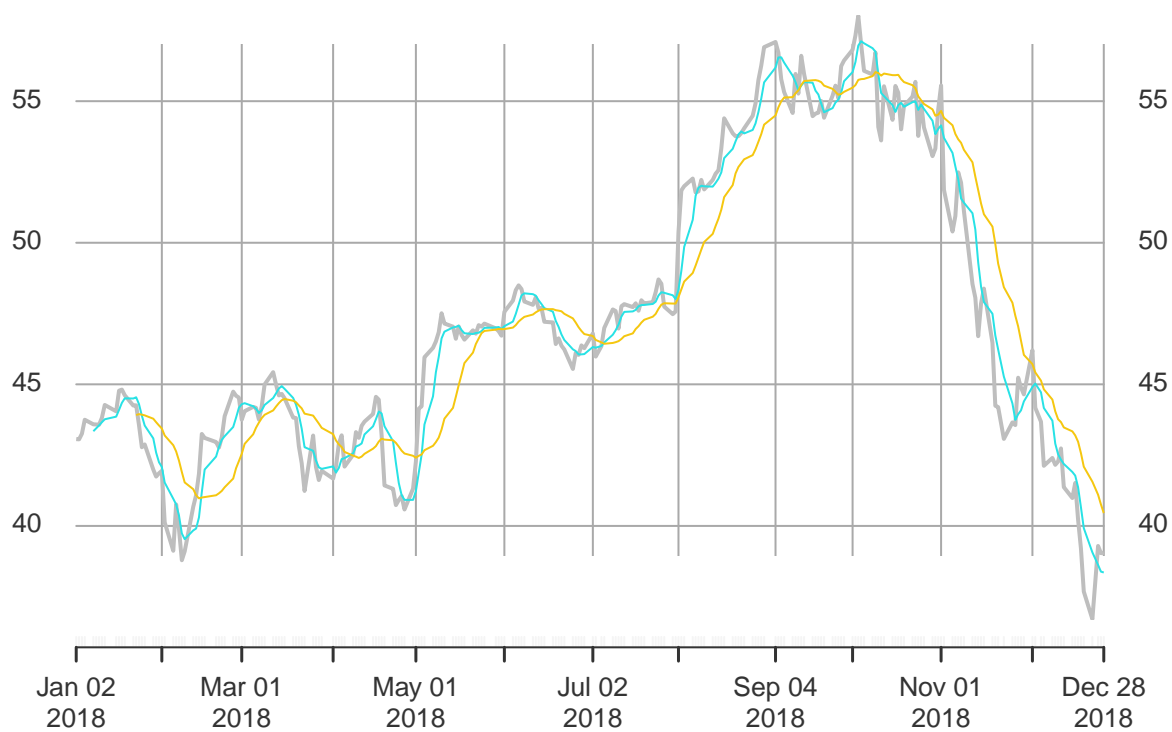
2018-01-02 / 2018-12-28



```
lines(mm_aapl_15, col = 15)
```


Médias Móveis

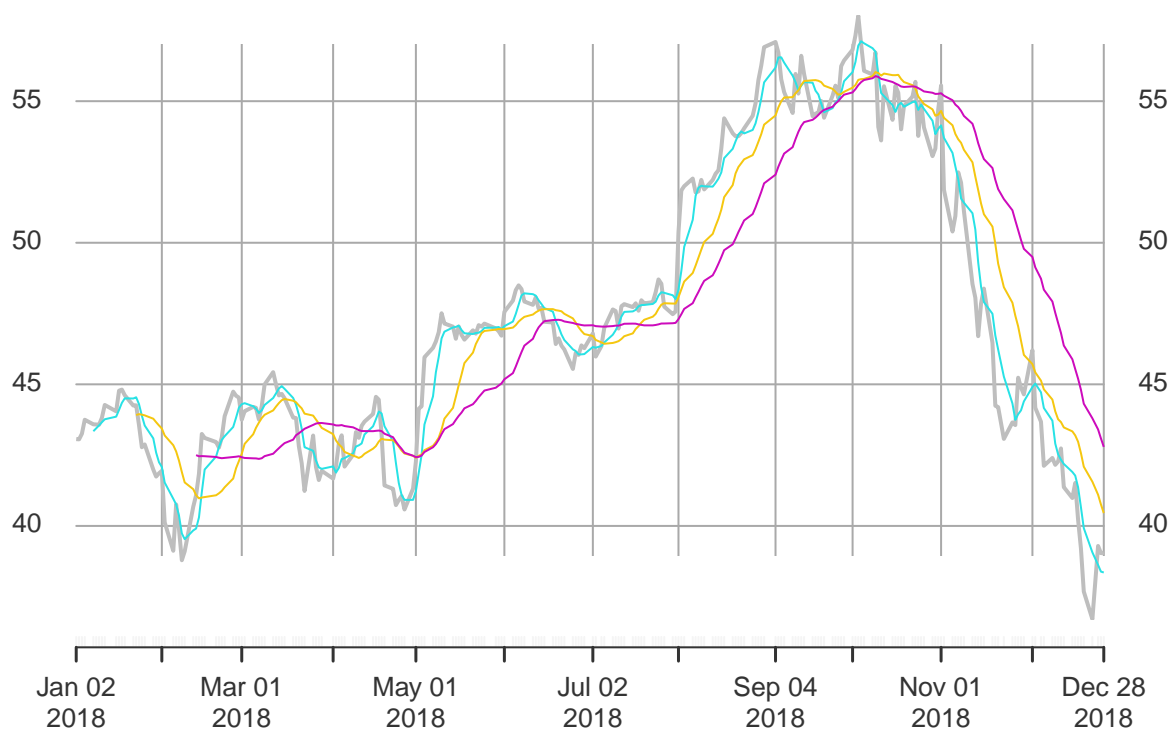
2018-01-02 / 2018-12-28



```
lines(mm_aapl_30, col = 30)
```

Médias Móveis

2018-01-02 / 2018-12-28

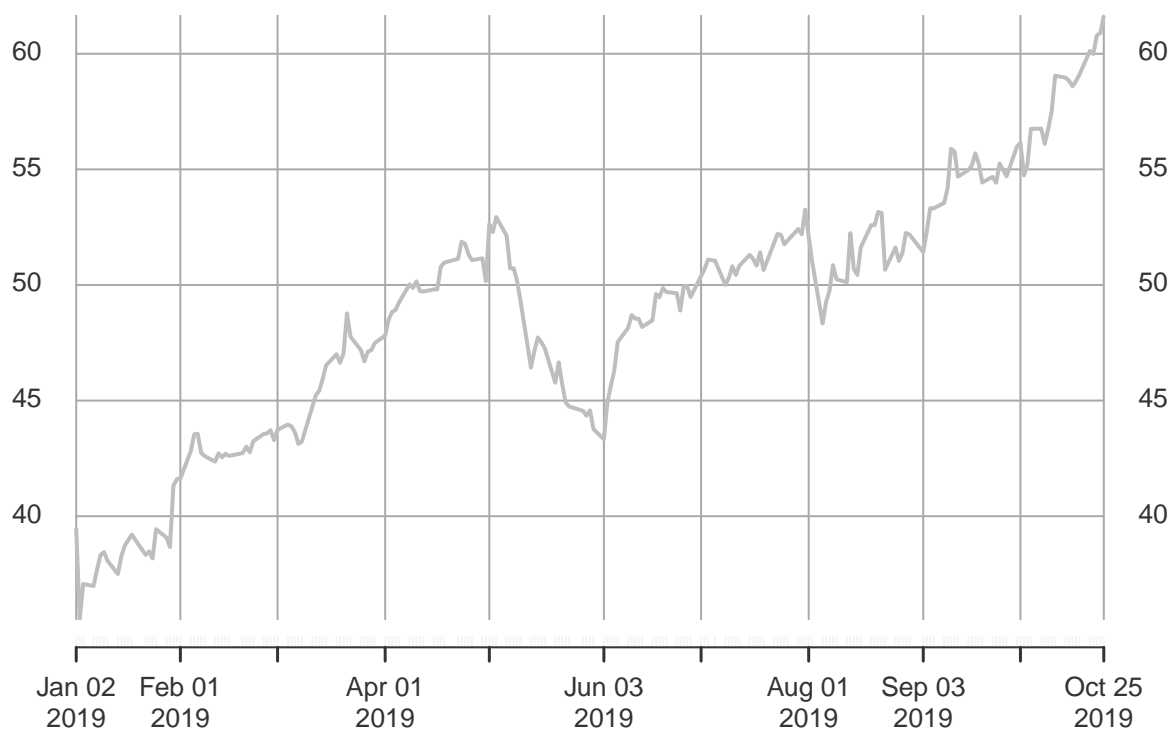


Segundo período

```
mm_aapl_5 <- rollmean(C1(t2), k = 5, align = c("right"))
mm_aapl_15 <- rollmean(C1(t2), k = 15, align = c("right"))
mm_aapl_30 <- rollmean(C1(t2), k = 30, align = c("right"))
plot(C1(t2), col = "gray", main = "Médias Móveis")
```

Médias Móveis

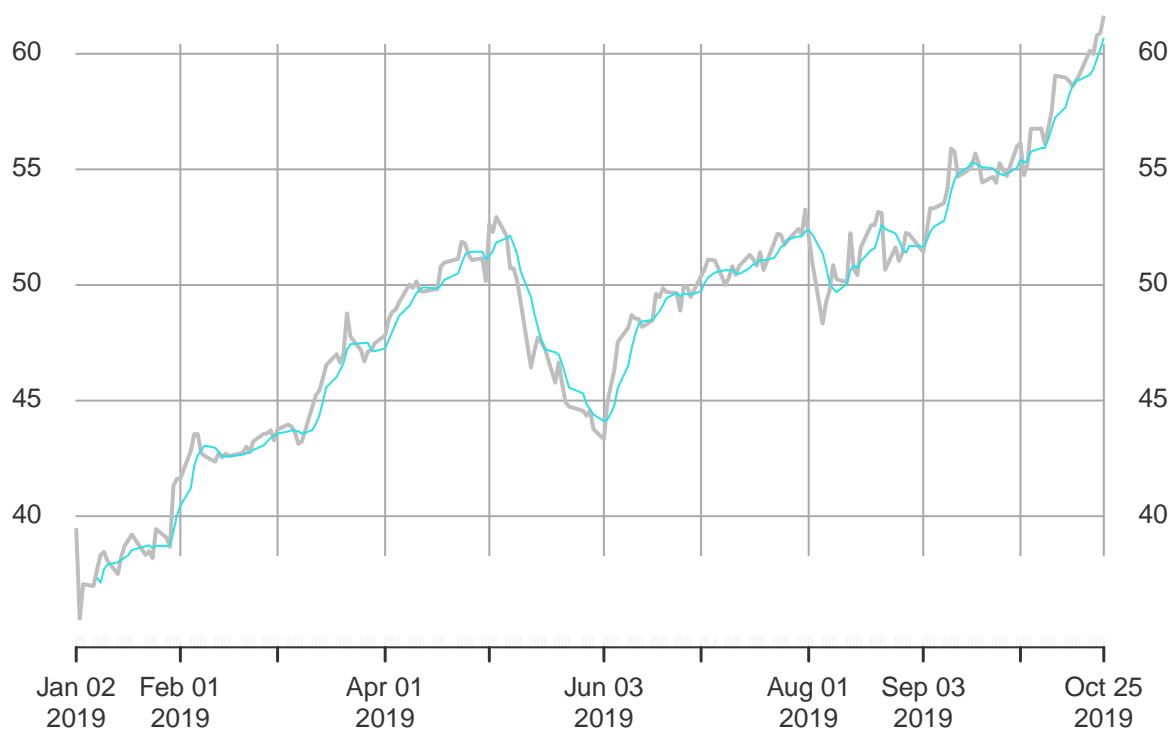
2019-01-02 / 2019-10-25



```
lines(mm_aapl_5, col = 5)
```

Médias Móveis

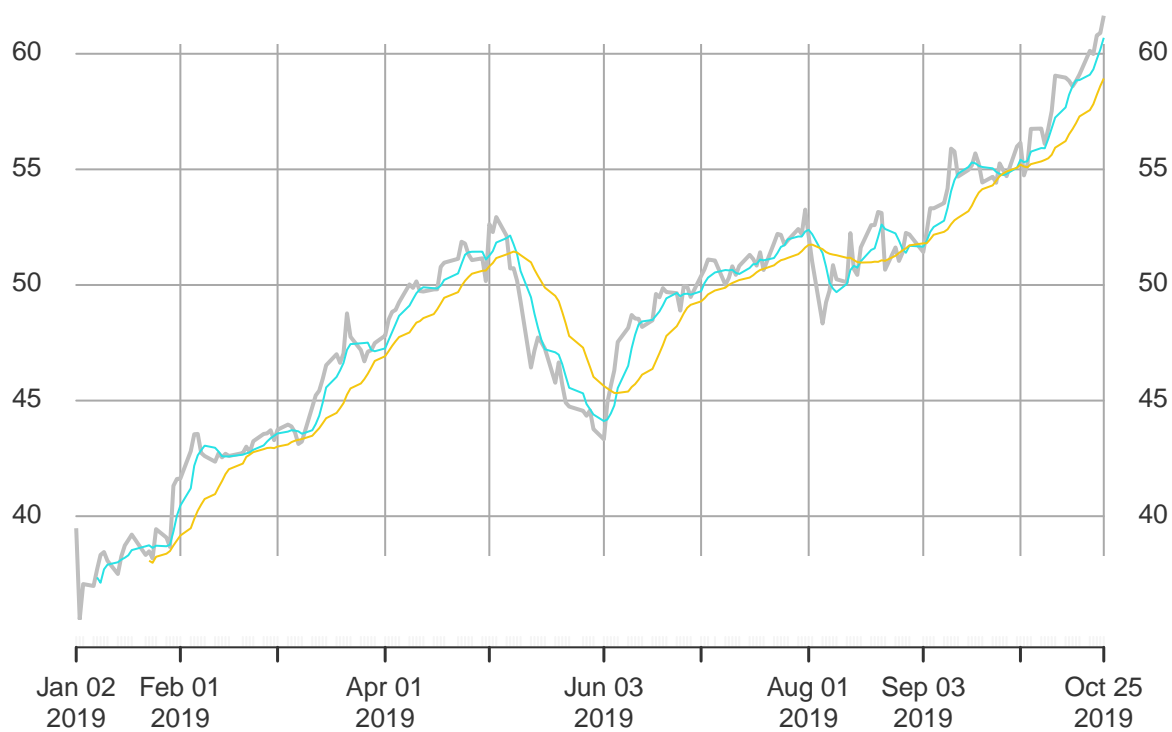
2019-01-02 / 2019-10-25



```
lines(mm_aapl_15, col = 15)
```

Médias Móveis

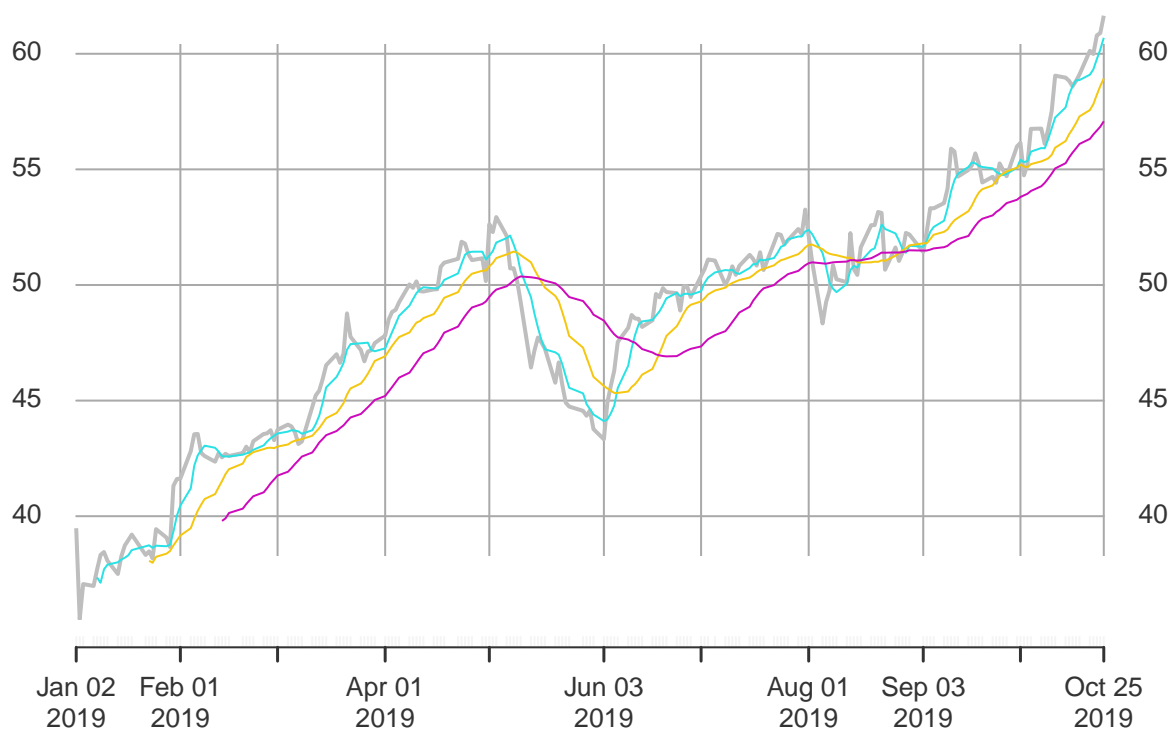
2019-01-02 / 2019-10-25



```
lines(mm_aapl_30, col = 30)
```

Médias Móveis

2019-01-02 / 2019-10-25



Médias móveis suavizam dados e identificar tendências ao longo do tempo. Conforme maior é a janela, maior o grau de suavização das médias móveis.

f)

Calculando os retornos simples dos dois períodos

```
n1 <- length(as.ts(Cl(t1)))
ret_simples_t1 <- as.numeric( diff(Cl(t1)) / Cl(t1) )[2:n1]

n2 <- length((as.ts(Cl(t2))))
ret_simples_t2 <- as.numeric( diff(Cl(t2)) / Cl(t2) )[2:n2]
```

Calculando os log-retornos dos dois períodos

```
log_ret_t1 <- diff(log(as.numeric(Cl(t1))))
log_ret_t2 <- diff(log(as.numeric(Cl(t2))))
```

g)

Calculando as média dois períodos

```
paste("Média doo retorno simples primeiro período",  
      basicStats(ret_simples_t1)["Mean",1], "%")
```

```
## [1] "Média doo retorno simples primeiro período -0.000557 %"
```

```
paste("Média do retorno simples segundo período",  
      basicStats(ret_simples_t2)["Mean",1], "%")
```

```
## [1] "Média do retorno simples segundo período 0.002005 %"
```

```
paste("Média do log-retorno primeiro período",  
      basicStats(log_ret_t1)["Mean",1], "%")
```

```
## [1] "Média do log-retorno primeiro período -0.000392 %"
```

```
paste("Média do log-retorno segundo período",  
      basicStats(log_ret_t2)["Mean",1], "%")
```

```
## [1] "Média do log-retorno segundo período 0.002163 %"
```

Calculando desvios dos dois períodos

```
paste("Desvio-padrão do primeiro período",  
      basicStats(ret_simples_t1)["Stdev",1])
```

```
## [1] "Desvio-padrão do primeiro período 0.01819"
```

```
paste("Desvio-padrão do segundo período",  
      basicStats(ret_simples_t2)["Stdev",1])
```

```
## [1] "Desvio-padrão do segundo período 0.0178"
```

```
paste("Desvio-padrão do primeiro período",  
      basicStats(log_ret_t1)["Stdev",1])
```

```
## [1] "Desvio-padrão do primeiro período 0.018155"
```

```
paste("Desvio-padrão do segundo período",  
      basicStats(log_ret_t2)["Stdev",1])
```

```
## [1] "Desvio-padrão do segundo período 0.017653"
```

Testando hipótese $H_0 : \mu_{log-retorno} = 0$

```
t.test(log_ret_t1)
```

```
##  
## One Sample t-test  
##  
## data: log_ret_t1  
## t = -0.34096, df = 248, p-value = 0.7334  
## alternative hypothesis: true mean is not equal to 0  
## 95 percent confidence interval:  
## -0.002658266 0.001873723  
## sample estimates:  
## mean of x  
## -0.0003922718
```

```
t.test(log_ret_t2)
```

```
##  
## One Sample t-test  
##  
## data: log_ret_t2  
## t = 1.7587, df = 205, p-value = 0.08012  
## alternative hypothesis: true mean is not equal to 0  
## 95 percent confidence interval:  
## -0.0002618869 0.0045880804  
## sample estimates:  
## mean of x  
## 0.002163097
```

h)

Primeiro período

```
x <- seq(min(log_ret_t1), max(log_ret_t1),.001)  
  
densidade <- density(log_ret_t1)  
  
hist(log_ret_t1,  
     prob = TRUE,  
     breaks = 50,  
     main = "Retorno Apple vs Gaussiana",  
     xlab = "log-retorno (janeiro-dezembro/2018)")  
  
curve(dnorm(x,  
           mean=mean(log_ret_t1),  
           sd=sd(log_ret_t1)),  
      col="darkblue",  
      lwd=2,  
      add=TRUE,  
      yaxt="n")
```

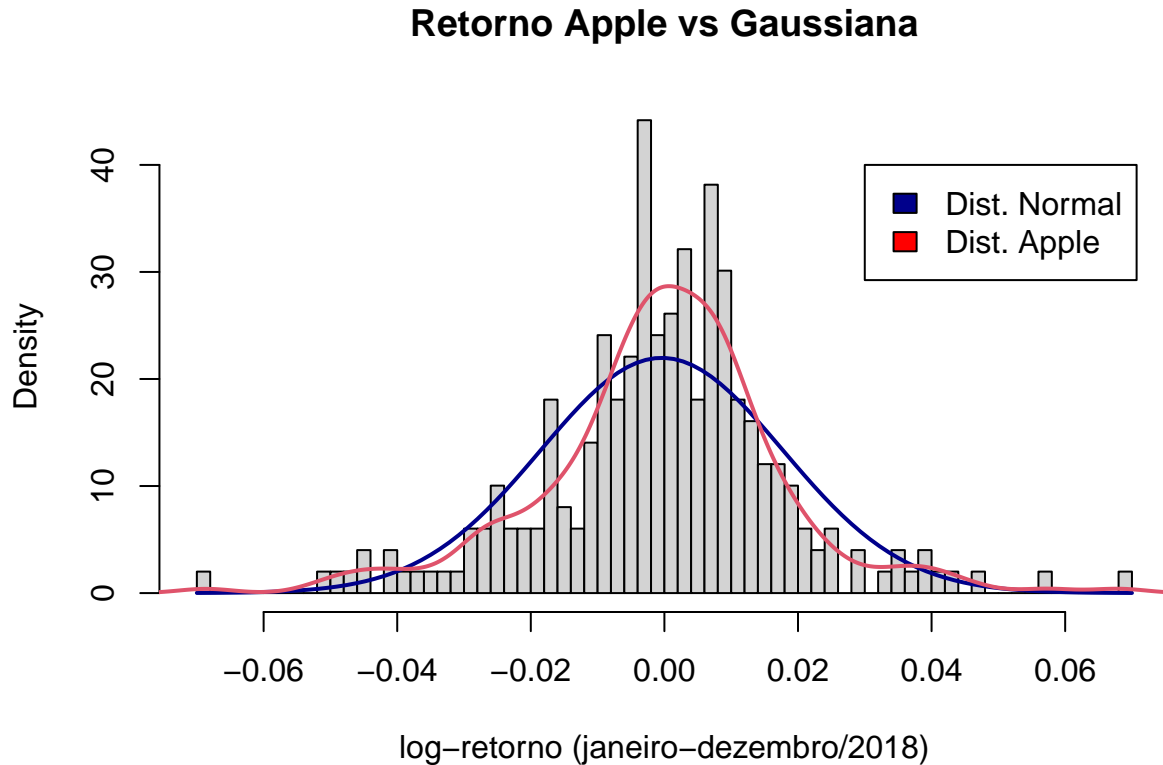


```

lines(densidade,
      lwd = 2,
      col = 2)

legend(x = 0.03, y = 40,
       legend = c("Dist. Normal", "Dist. Apple"),
       fill = c("darkblue", "red"))

```



Segundo período

```

x <- seq(min(log_ret_t2), max(log_ret_t2), .001)

densidade <- density(log_ret_t2)

hist(log_ret_t2,
     prob = TRUE,
     breaks = 50,
     main = "Retorno Apple vs Gaussiana",
     xlab = "log-retorno (janeiro-outubro/2019)")

curve(dnorm(x,
            mean=mean(log_ret_t2),
            sd=sd(log_ret_t2)),

```

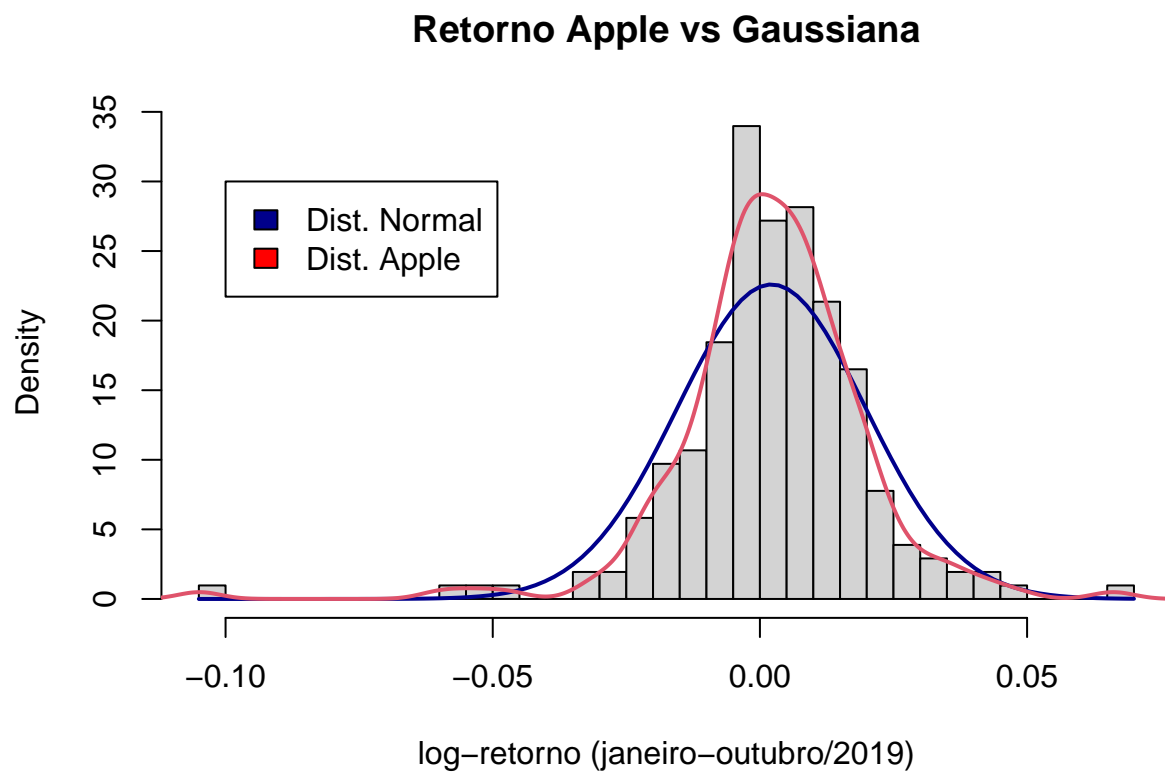
```

col="darkblue",
lwd=2,
add=TRUE,
yaxt="n")

lines(densidade,
      lwd = 2,
      col = 2)

legend(x = -0.1, y = 30,
       legend = c("Dist. Normal", "Dist. Apple"),
       fill = c("darkblue", "red"))

```



2.

Carregando bibliotecas

```
library(tidyverse)
```

2_1.

```
dados <- read.table("./Dados/ch1data/d-axp3dx-0111.txt", header = T)
view(dados)
```

a)

```
infos <- c("Mean", "Stdev", "Skewness", "Kurtosis", "Minimum", "Maximum")

for (col in 2:length(dados)) {

  values <- ((basicStats(
    dados[col])
    [infos, 1]))

  print(colnames(dados[col]))
  print(infos)
  print(values)
}
```

```
## [1] "axp"
## [1] "Mean"      "Stdev"      "Skewness" "Kurtosis" "Minimum" "Maximum"
## [1] 0.000534 0.026368 0.459773 9.592053 -0.175949 0.206485
## [1] "vw"
## [1] "Mean"      "Stdev"      "Skewness" "Kurtosis" "Minimum" "Maximum"
## [1] 0.000224 0.013652 -0.098318 7.982134 -0.089762 0.114889
## [1] "ew"
## [1] "Mean"      "Stdev"      "Skewness" "Kurtosis" "Minimum" "Maximum"
## [1] 0.000626 0.012080 -0.247410 8.108428 -0.078240 0.107422
## [1] "sp"
## [1] "Mean"      "Stdev"      "Skewness" "Kurtosis" "Minimum" "Maximum"
## [1] 0.000094 0.013779 0.008152 8.532667 -0.090350 0.115800
```

b)

```
log_ret <- log(1 + dados[2:5])

for (col in 1:length(log_ret)) {

  values <- ((basicStats(
    log_ret[col])
    [infos, 1]))

  print(colnames(log_ret[col]))
  print(infos)
  print(values)
}
```

```
## [1] "axp"
## [1] "Mean"      "Stdev"      "Skewness" "Kurtosis" "Minimum" "Maximum"
## [1] 0.000188 0.026294 0.020992 9.020499 -0.193523 0.187711
## [1] "vw"
## [1] "Mean"      "Stdev"      "Skewness" "Kurtosis" "Minimum" "Maximum"
## [1] 0.000131 0.013670 -0.300352 7.880082 -0.094049 0.108755
## [1] "ew"
## [1] "Mean"      "Stdev"      "Skewness" "Kurtosis" "Minimum" "Maximum"
## [1] 0.000553 0.012100 -0.427315 8.017712 -0.081470 0.102035
## [1] "sp"
## [1] "Mean"      "Stdev"      "Skewness" "Kurtosis" "Minimum" "Maximum"
## [1] -0.000001 0.013790 -0.206357 8.322826 -0.094695 0.109572
```

c)

```
t.test(log_ret["axp"], mu = 0, alternative = "two.sided")
```

```
##
## One Sample t-test
##
## data: log_ret["axp"]
## t = 0.35999, df = 2534, p-value = 0.7189
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
## -0.0008360686 0.0012120714
## sample estimates:
## mean of x
## 0.0001880014
```

O p -valor = 0,7189 da hipótese nula $H_0 : \mu = 0$ está muito além do $\alpha = 5\%$, então não rejeitamos a hipótese nula de que a média do log-retorno de AXP é igual a zero.

2_2.

Um dos valores do SP é um caractere “.”, precisamos remover esse caractere para realizar os cálculos. Através do código: `dados[, "sp"][dados[, "sp"] == "."]` eu encontro o caractere e depois eu o substituo pela média dos valores de SP, desconsiderando o próprio caractere com: `mean(as.numeric(dados[, "sp"]), na.rm = TRUE)`

```
dados <- read.table("./Dados/ch1data/m-ge3dx-4011.txt", header = T)
view(dados)
dados[, "sp"][dados[, "sp"] == "."] <- mean(as.numeric(dados[, "sp"]),
, na.rm = TRUE)
dados[, "sp"] <- as.numeric(dados[, "sp"])
view(dados)
```

a)

```

infos <- c("Mean", "Stdev", "Skewness", "Kurtosis", "Minimum", "Maximum")

for (col in 2:5) {

  values <- ((basicStats(
    dados[col])
    [infos, 1]))

  print(colnames(dados[col]))
  print(infos)
  print(values)
}

```

```

## [1] "ge"
## [1] "Mean"      "Stdev"      "Skewness" "Kurtosis" "Minimum" "Maximum"
## [1] 0.010519 0.065998 0.051618 1.239488 -0.272877 0.251236
## [1] "vw"
## [1] "Mean"      "Stdev"      "Skewness" "Kurtosis" "Minimum" "Maximum"
## [1] 0.009316 0.043328 -0.660821 2.355320 -0.225363 0.165585
## [1] "ew"
## [1] "Mean"      "Stdev"      "Skewness" "Kurtosis" "Minimum" "Maximum"
## [1] 0.012179 0.054793 -0.306964 3.138812 -0.272248 0.299260
## [1] "sp"
## [1] "Mean"      "Stdev"      "Skewness" "Kurtosis" "Minimum" "Maximum"
## [1] 0.006178 0.042664 -0.589798 2.367521 -0.239541 0.163047

```

b)

```

log_ret <- log(1 + dados[2:5])

for (col in 1:length(log_ret)) {

  values <- ((basicStats(
    log_ret[col])
    [infos, 1]))

  print(colnames(log_ret[col]))
  print(infos)
  print(values)
}

```

```

## [1] "ge"
## [1] "Mean"      "Stdev"      "Skewness" "Kurtosis" "Minimum" "Maximum"
## [1] 0.008318 0.065735 -0.290782 1.778316 -0.318660 0.224132
## [1] "vw"
## [1] "Mean"      "Stdev"      "Skewness" "Kurtosis" "Minimum" "Maximum"
## [1] 0.008331 0.043740 -0.943052 3.517912 -0.255361 0.153223
## [1] "ew"
## [1] "Mean"      "Stdev"      "Skewness" "Kurtosis" "Minimum" "Maximum"
## [1] 0.010611 0.055043 -0.745712 4.169659 -0.317795 0.261795
## [1] "sp"

```

```
## [1] "Mean"      "Stdev"      "Skewness" "Kurtosis" "Minimum" "Maximum"
## [1] 0.005241 0.043130 -0.878415 3.612070 -0.273833 0.151043
```

c)

```
t.test(log_ret["ge"], mu = 0, alternative = "two.sided")
```

```
##
## One Sample t-test
##
## data: log_ret["ge"]
## t = 3.713, df = 860, p-value = 0.000218
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
## 0.003921037 0.012715061
## sample estimates:
## mean of x
## 0.008318049
```

O p -valor = 0,000218 da hipótese nula $H_0 : \mu = 0$ é muito menor do que $\alpha = 5\%$, então rejeitamos a hipótese nula de que a média do log-retorno de GE é igual a zero.

2_3.

a)

```
t.test(dados["sp"], mu = 0, alternative = "two.sided")
```

```
##
## One Sample t-test
##
## data: dados["sp"]
## t = 4.2488, df = 860, p-value = 2.384e-05
## alternative hypothesis: true mean is not equal to 0
## 95 percent confidence interval:
## 0.003323913 0.009031515
## sample estimates:
## mean of x
## 0.006177714
```

O p -valor obtido é praticamente 0, portanto, rejeitamos $H_0 : \mu = 0$

b)

```

skw <- skewness(dados[, "ge"])
t_obs <- skw / sqrt(6 / length(dados[, "ge"]))
p_valor <- 2 * (1 - pnorm(t_obs))

if (p_valor > 0.05) {
  cat("p-value = ", p_valor, " > 0.05, cannot reject the null hypothesis")
} else {
  cat("p-value = ", p_valor, " <= 0.05, reject the null hypothesis")
}

```

```
## p-value = 0.5363498 > 0.05, cannot reject the null hypothesis
```

c)

```

kt <- kurtosis(dados[, "ge"])
t_obs <- kt / sqrt(24 / length(dados[, "ge"]))
p_valor <- 2 * (1 - pnorm(t_obs))

if (p_valor > 0.05) {
  cat("p-value = ", p_valor, " > 0.05, cannot reject the null hypothesis")
} else {
  cat("p-value = ", p_valor, " <= 0.05, reject the null hypothesis")
}

```

```
## p-value = 1.136868e-13 <= 0.05, reject the null hypothesis
```

2_4.

a)

```

dados <- read.table("./Dados/ch1data/d-axp3dx-0111.txt", header = T)

skw <- skewness(log(dados["axp"] + 1))
t_obs <- skw / sqrt(6 / length(dados[, "axp"]))
p_valor <- 2 * (1 - pnorm(t_obs))

if (p_valor > 0.05) {
  cat("p-value = ", p_valor, " > 0.05, cannot reject the null hypothesis")
} else {
  cat("p-value = ", p_valor, " <= 0.05, reject the null hypothesis")
}

```

```
## p-value = 0.6661179 > 0.05, cannot reject the null hypothesis
```

b)

```

kt <- kurtosis(log(dados[, "axp"] + 1))
t_obs <- kt / sqrt(24 / length(dados[, "axp"]))
p_valor <- 2 * (1 - pnorm(t_obs))

if (p_valor > 0.05) {
  cat("p-value = ",p_valor," > 0.05, cannot reject the null hypothesis")
} else {
  cat("p-value = ",p_valor," <= 0.05, reject the null hypothesis")
}

```

```
## p-value = 0 <= 0.05, reject the null hypothesis
```

2_5.

```

dados_uk = read.table("./Dados/ch1data/d-fx-ukus-0711.txt", header=T)
dados_jp = read.table("./Dados/ch1data/d-fx-ukus-0711.txt", header=T)

```

a)

```

diff_uk = diff(log(dados_uk$rate))
diff_jp = diff(log(dados_jp$rate))

```

b)

```
paste(infos,basicStats(diff_uk)[infos, 1])
```

```

## [1] "Mean -0.000185"      "Stdev 0.007485"      "Skewness -0.359374"
## [4] "Kurtosis 5.467822"   "Minimum -0.049662"   "Maximum 0.044349"

```

```
paste(infos,basicStats(diff_jp)[infos, 1])
```

```

## [1] "Mean -0.000185"      "Stdev 0.007485"      "Skewness -0.359374"
## [4] "Kurtosis 5.467822"   "Minimum -0.049662"   "Maximum 0.044349"

```

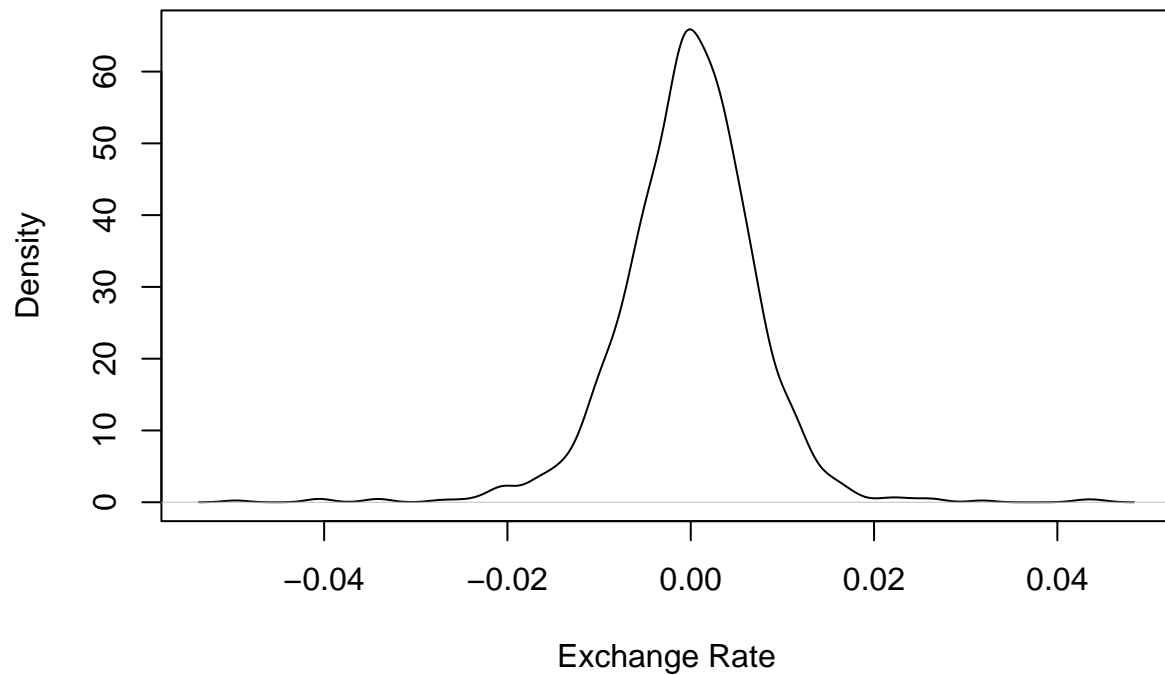
c)

```

plot(density(diff_jp), main = "Dollar-Yen Exchange Rate",
     xlab = "Exchange Rate")

```


Dollar-Yen Exchange Rate



d)

```
p_valor <- t.test(diff_jp, mu = 0, alternative = "two.sided")$p.value

if (p_valor > 0.05) {
  cat("p-value = ",p_valor," > 0.05, cannot reject the null hypothesis")
} else {
  cat("p-value = ",p_valor," <= 0.05, reject the null hypothesis")
}
```

```
## p-value = 0.3856454 > 0.05, cannot reject the null hypothesis
```