Lista_01

Marcelo Neves Lira

Parte teórica

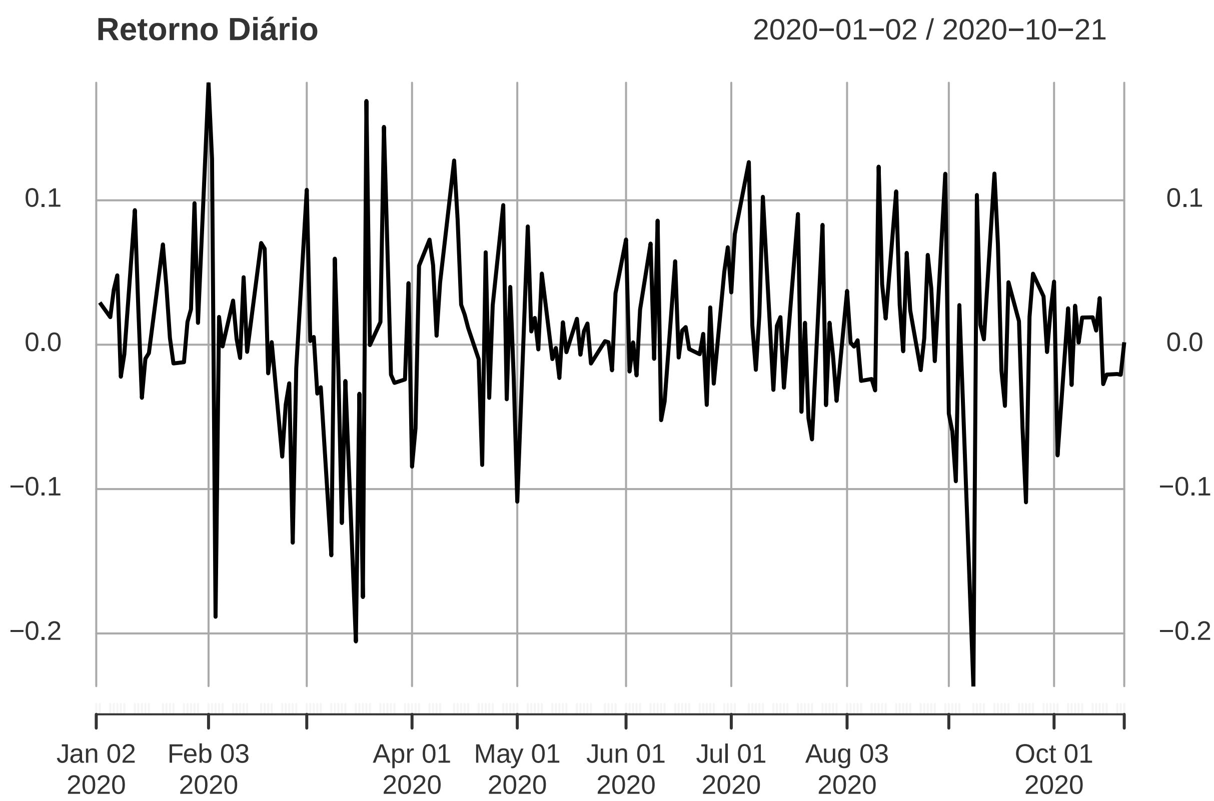
1.

a)

```r
library(quantmod)
getSymbols("TSLA", from = "2020-01-02", to = "2020-10-22")
```

```
## [1] "TSLA"
```

```r
TSLA.return <- diff(log(TSLA$TSLA.Close))
plot(TSLA.return, main = "Retorno Diário")
```



O gráfico com o retorno diário da Tesla é uma Série Temporal, ou seja, uma sequência numérica que se desenvolve em função do tempo, espaço, volume, ou dimensões de outra natureza. Já um Processo Estocástico $\{x_t\}_{t \in T} = \{x(t, w); t \in T, w \in \Omega\}$ é uma família de variáveis aleatórias indexadas em $t \in T$. Ou seja, é um conjunto de funções definidas em um mesmo Espaço de Probabilidade $\{\Omega, A, P\}$. São modelos utilizados para representar séries temporais.

b)

A natureza dos fenômenos econômicos está relacionada a comportamentos humanos, interações sociais e decisões tomadas por indivíduos, empresas e governos. Por outro lado, os fenômenos físicos são regidos por leis naturais e princípios científicos estabelecidos. Em fenômenos físicos, muitas vezes é possível identificar relações de causa e efeito bem definidas. Por exemplo, a lei da gravidade estabelece uma relação clara entre a massa dos objetos e a força gravitacional exercida sobre eles. No entanto, nos fenômenos econômicos, a causalidade muitas vezes é complexa e multifacetada, envolvendo uma variedade de fatores interconectados.

Desse modo, os fenômenos econômicos geralmente exibem um alto grau de aleatoriedade e incerteza. As decisões tomadas pelos agentes econômicos são influenciadas por uma variedade de fatores, incluindo mudanças nas condições econômicas, expectativas e comportamentos imprevisíveis. Por outro lado, os fenômenos físicos muitas vezes são mais determinísticos e previsíveis, uma vez que são regidos por leis naturais consistentes.

c)

A análise de séries temporais permite identificar padrões e tendências nos dados econômicos ao longo do tempo. Isso ajuda os economistas a compreenderem melhor os movimentos cíclicos, sazonais e de longo prazo da economia. A detecção de padrões pode levar a insights importantes sobre a dinâmica econômica e fornecer base empírica para o desenvolvimento de teorias econômicas. Através da análise de séries temporais, é possível desenvolver modelos e técnicas de previsão econômica que ajudam na tomada de decisões e na análise de políticas econômicas. Os modelos de séries temporais podem fornecer projeções e estimativas para variáveis econômicas importantes, permitindo aos formuladores de políticas antecipar e responder a mudanças econômicas.

Em suma, a análise de séries temporais univariadas desempenha um papel importante no desenvolvimento da Teoria Econômica, fornecendo insights empíricos, testando hipóteses, fornecendo ferramentas de previsão e modelando fenômenos econômicos complexos.

d)

Um processo estacionário de série temporal é aquele em que as distribuições de probabilidades são estáveis no decorrer do tempo, ou seja, se pegarmos qualquer coleção de variáveis aleatórias na sequência e depois deslocarmos essa sequência para diante em h períodos de tempo, a distribuição de probabilidade conjunta deve permanecer inalterada.

e)

Para que uma série temporal apresente estacionariedade é preciso que sua média e variância sejam invariantes no tempo, ou seja, $E(y_t) = \mu, Var(y_t) < +\infty, \forall t \in T$. Além disso, $Cov(y_t, y_{t-k}) = \gamma(k)$, ou seja, uma série fracamente estacionária, depende apenas de k, e mede a dependência linear de y_t sobre seu passado y_{t-k}.

f)

$$x_t = \beta_0 + \beta_1 t + \varepsilon_t$$

1ª condição de estacionariedade fraca $E(x_t) = \mu$

$$E(x_t) = E(\beta_0 + \beta_1 t + \varepsilon_t) \Rightarrow$$

$$E(x_t) = E(\beta_0) + E(\beta_1 t) + E(\varepsilon_t) \Rightarrow$$

$$E(x_t) = \beta_0 + \beta_1 \cdot E(t) + E(\varepsilon_t) \Rightarrow$$

Como $\varepsilon_t \sim R.B(0, \sigma^2)$, então $E(\varepsilon_t) = 0$

$$E(x_t) = \beta_0 + \beta_1 \cdot E(t)$$

Como $E(x_t) \neq cte$, a 1ª condição não é cumprida, e x_t não é fracamente estacionário

.

2.

Dada uma série temporal fracamente estacionária x_t, a Função de Autocovariância (FACV) mede a covariância entre observações separadas por um determinado intervalo de tempo. A FACV é definida como:

$$\gamma(h) = Cov(x_t, x_{t\pm h}),$$

onde h é o intervalo de tempo entre as observações x_t e x_{t+h}. A FACV mede a covariância entre os valores em diferentes momentos, sendo positiva se eles tendem a variar juntos e negativa se variam em direções opostas.

A Função de Autocorrelação é a coleção de correlações dada por

$$\rho_h = \frac{Cov(x_t, x_{t\pm h})}{sqrt(Var(x_t)Var(x_{t\pm h}))}$$

É uma versão normalizada da FACV. A FAC normalizada varia entre -1 e 1, indicando a força e a direção da dependência entre os valores separados por um determinado intervalo de tempo. Um valor próximo de 1 indica uma forte autocorrelação positiva, enquanto um valor próximo de -1 indica uma forte autocorrelação negativa. Um valor próximo de zero indica que não há autocorrelação significativa.

A Função de Autocovariância (FACV) e a Função de Autocorrelação (FAC) são ferramentas estatísticas que ajudam a identificar e compreender a dependência temporal em séries temporais. Elas fornecem insights sobre padrões, auxiliam na escolha de modelos adequados.

3.

A Função de Autocorrelação Parcial (FACP) é uma extensão da Função de Autocorrelação (FAC) que ajuda a identificar a relação de dependência temporal entre observações em uma série temporal, controlando os efeitos de outras defasagens. A FACP pode auxiliar na identificação da ordem adequada de um modelo AR, indicando um número apropriado de diferenciações necessárias para tornar a série temporal estacionária.

4.

Parte Prática

1.

a)

```r
getSymbols("AAPL", from = "2018-01-01", to = "2018-12-31")
```

```
## [1] "AAPL"
```

```r
t1 <- AAPL # Perídodo refente ao ano de 2018
getSymbols("AAPL", from = "2019-01-01", to = "2019-10-28")
```

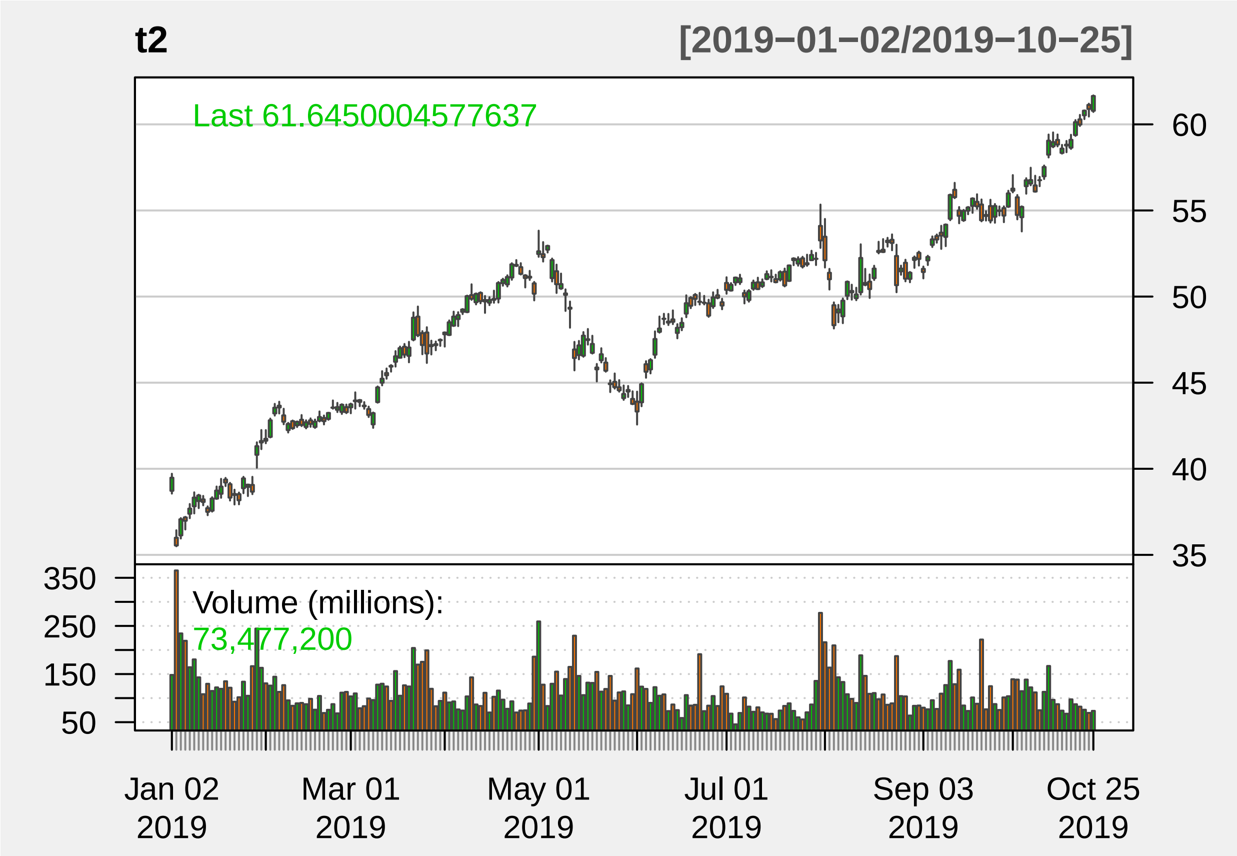
```
## [1] "AAPL"
```

```r
t2 <- AAPL # Perídodo refente ao ano de 2019
```

b)

Na parte superior do ráfico 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.

```r
chartSeries(t2, theme = "white")
```



c)

Primeiro período (janeiro a dezembro de 2018).

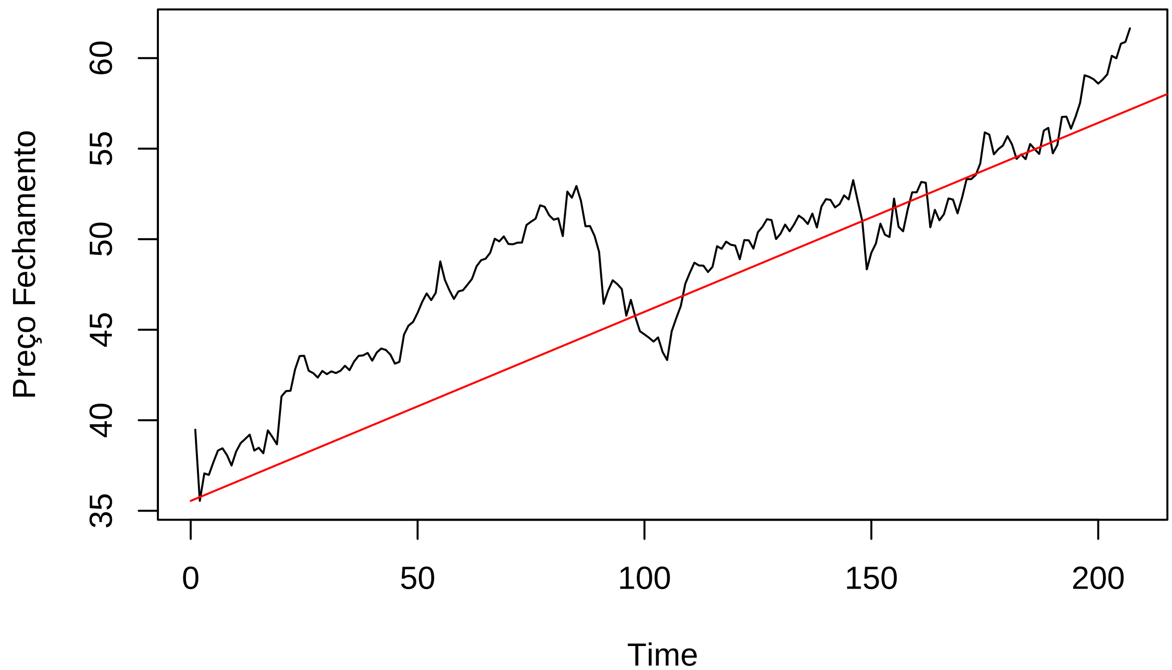
No início do período, as condições de estacionariedade fraca (i) e (ii) parecem valer. Depois disso, não parecem razoáveis de serem aceitas .

```
plot(t1$AAPL.Close)
```

Primeiro período (janeiro a outrubro de 2019)

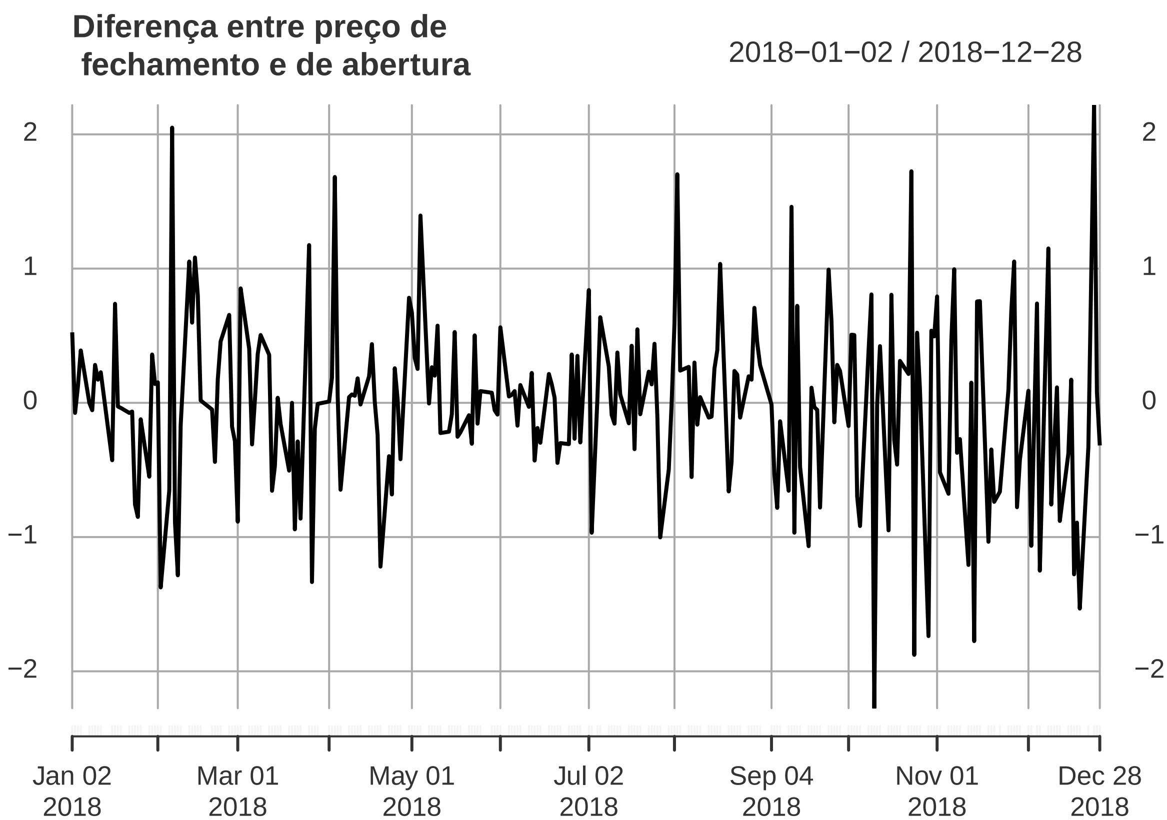
Essa série temporal apresenta um comportamento mais determinístico, que poderia ser estimado por uma reta. Definitivamente, não apresenta média e variância invariantes no tempo.

```r
# plot(t2$AAPL.Close)
plot.ts(t2$AAPL.Close, ylab = "Preço Fechamento")
lines(c(0, 250), c(min(t2$AAPL.Close), max(t2$AAPL.Close)), col = "red")
```

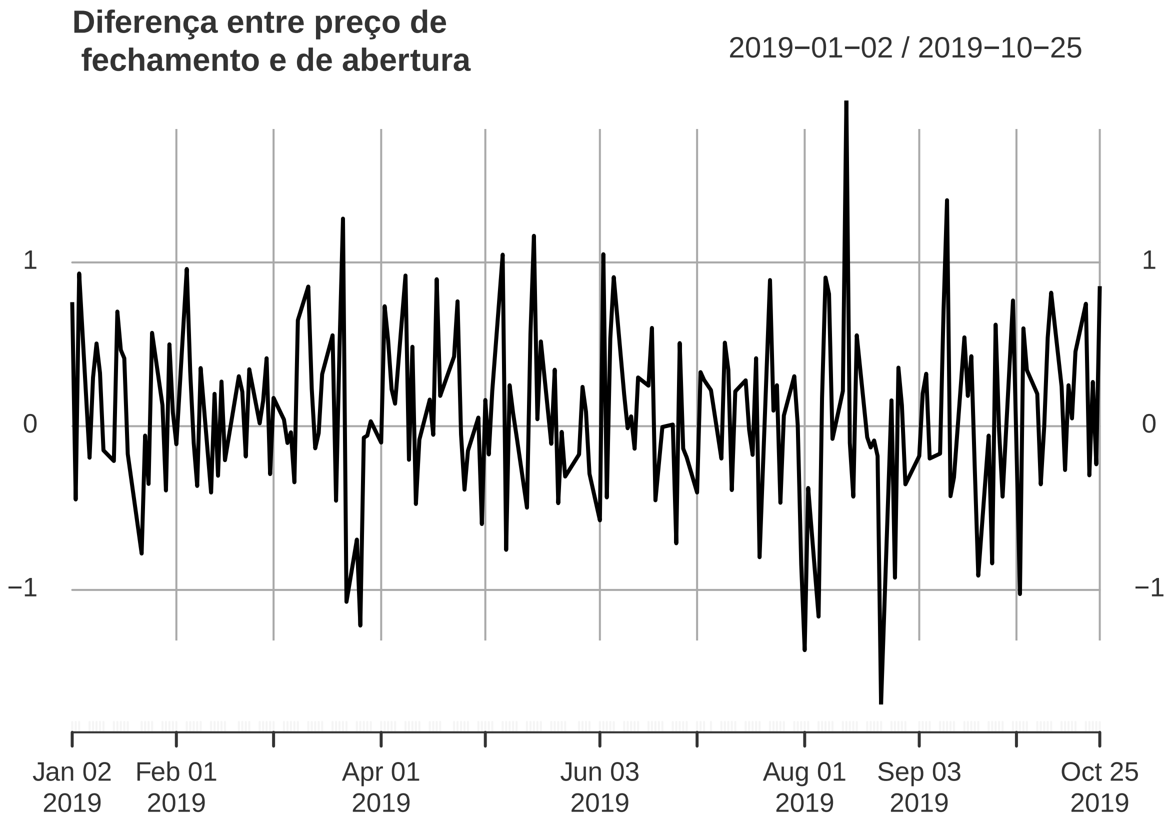


d)

```
delta_abertura_fechamento_t1 <- t1$AAPL.Close - t1$AAPL.Open
delta_abertura_fechamento_t2 <- t2$AAPL.Close - t2$AAPL.Open

plot(delta_abertura_fechamento_t1,
     main = "Diferença entre preço de \n fechamento e de abertura")
```




```
plot(delta_abertura_fechamento_t2,
     main = "Diferença entre preço de \n fechamento e de abertura")
```

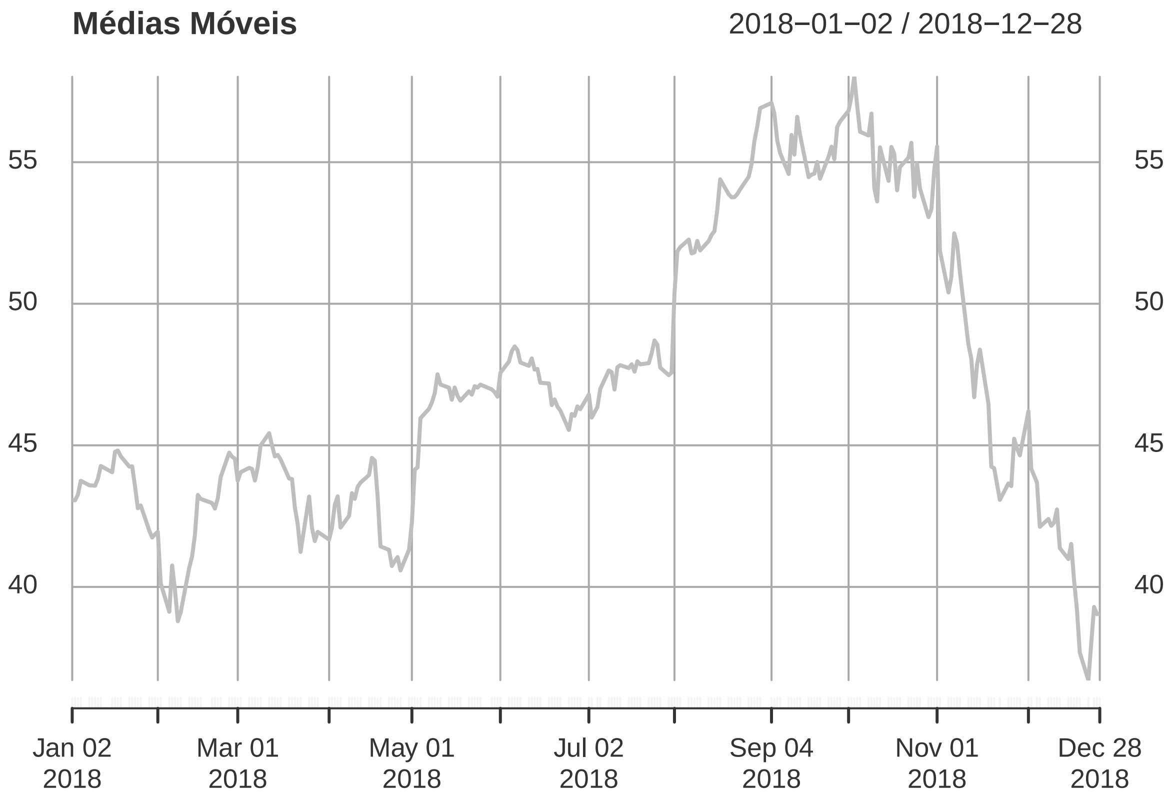


Ambas as séries agora apresentam características de um processo estacionário faco, com $E(x_t) = \mu$ e $Var(x_t) < +\infty$. Seria necessário observar o comportamento de decaimento dos lags pela FACV.

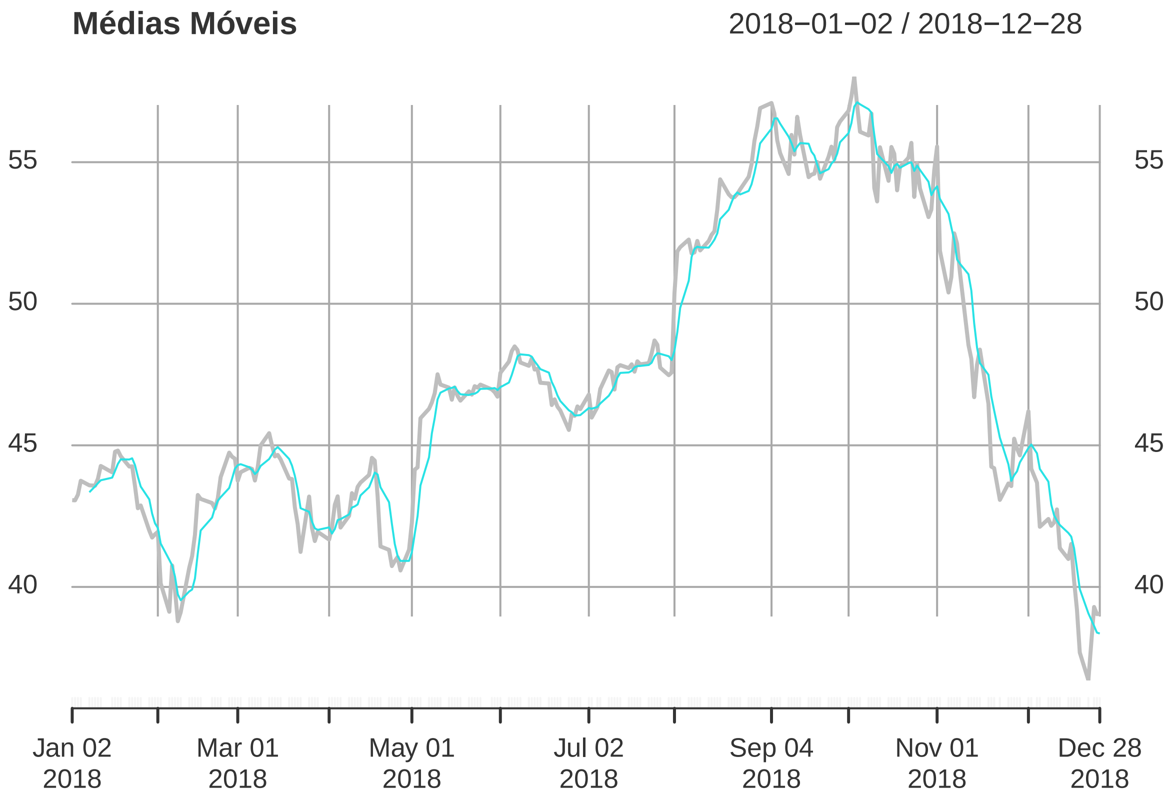
e)

Primeiro período .

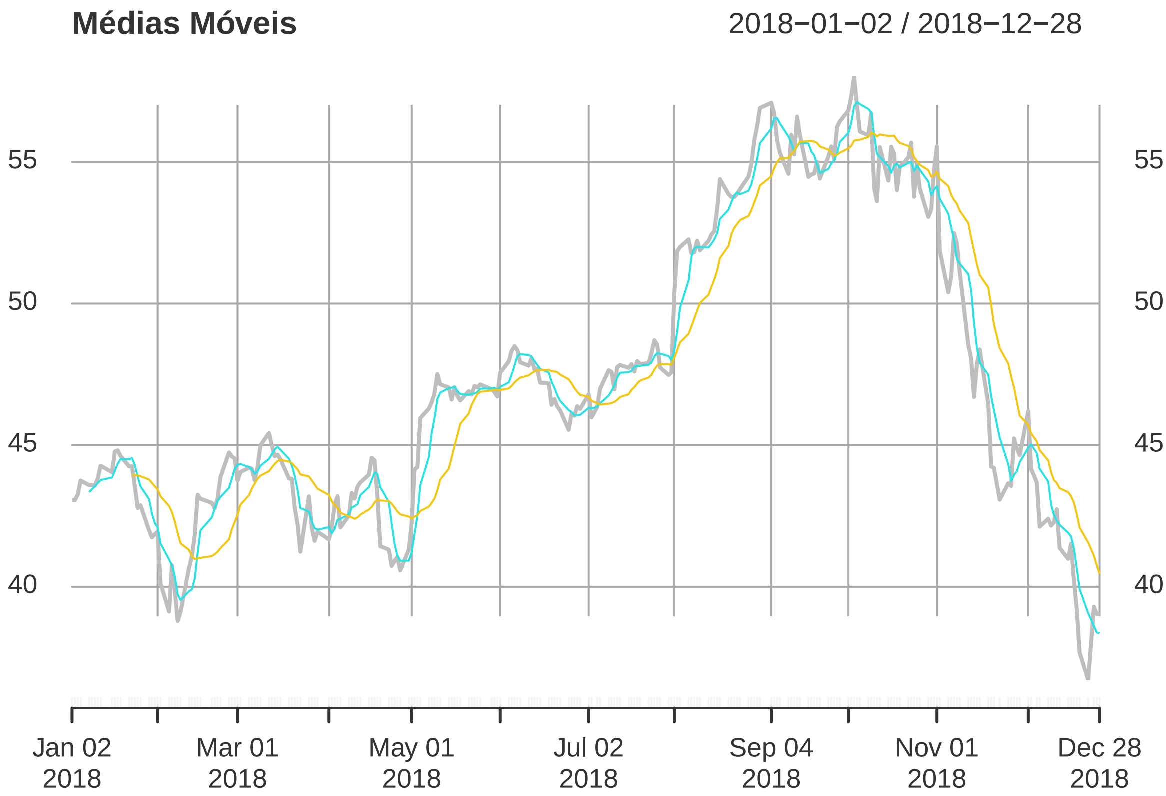
```
mm_aapl_5 <- rollmean(Cl(t1), k = 5, align = c("right"))
mm_aapl_15 <- rollmean(Cl(t1), k = 15, align = c("right"))
mm_aapl_30 <- rollmean(Cl(t1), k = 30, align = c("right"))
plot(Cl(t1), col = "gray", main = "Médias Móveis")
```



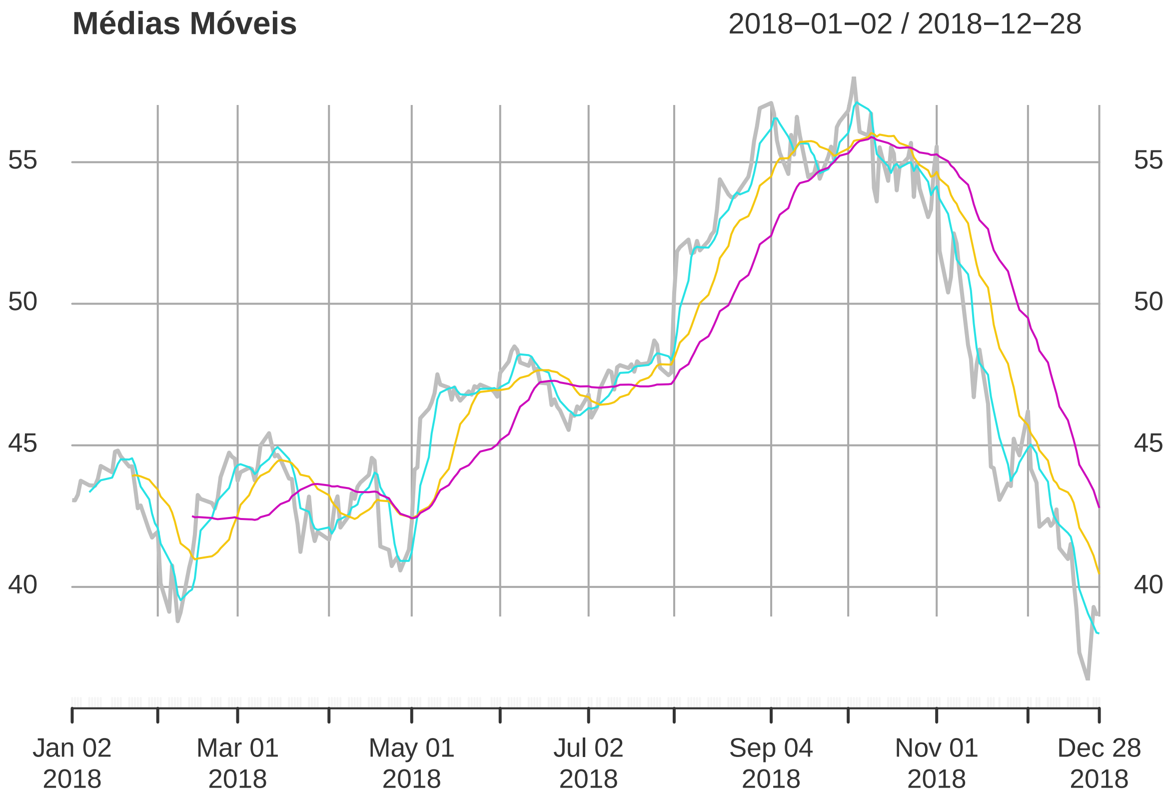

```
lines(mm_aapl_5, col = 5)
```




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

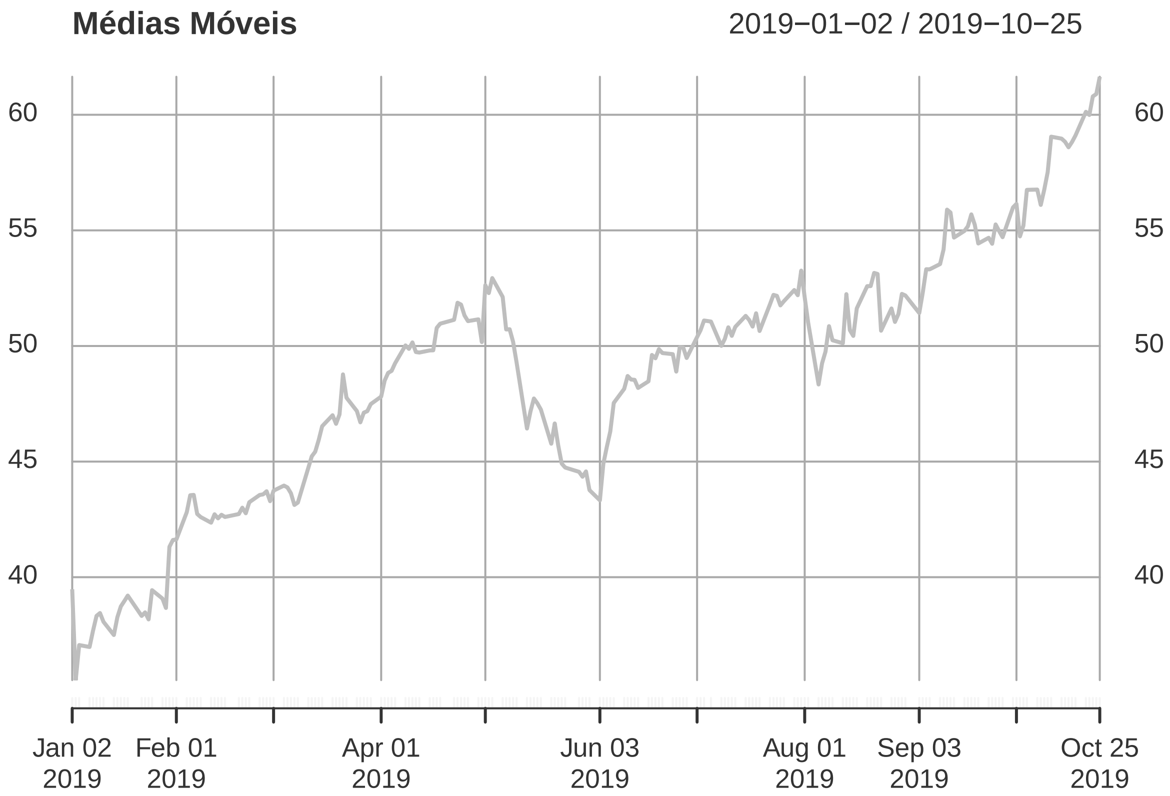



```r
lines(mm_aapl_30, col = 30)
```

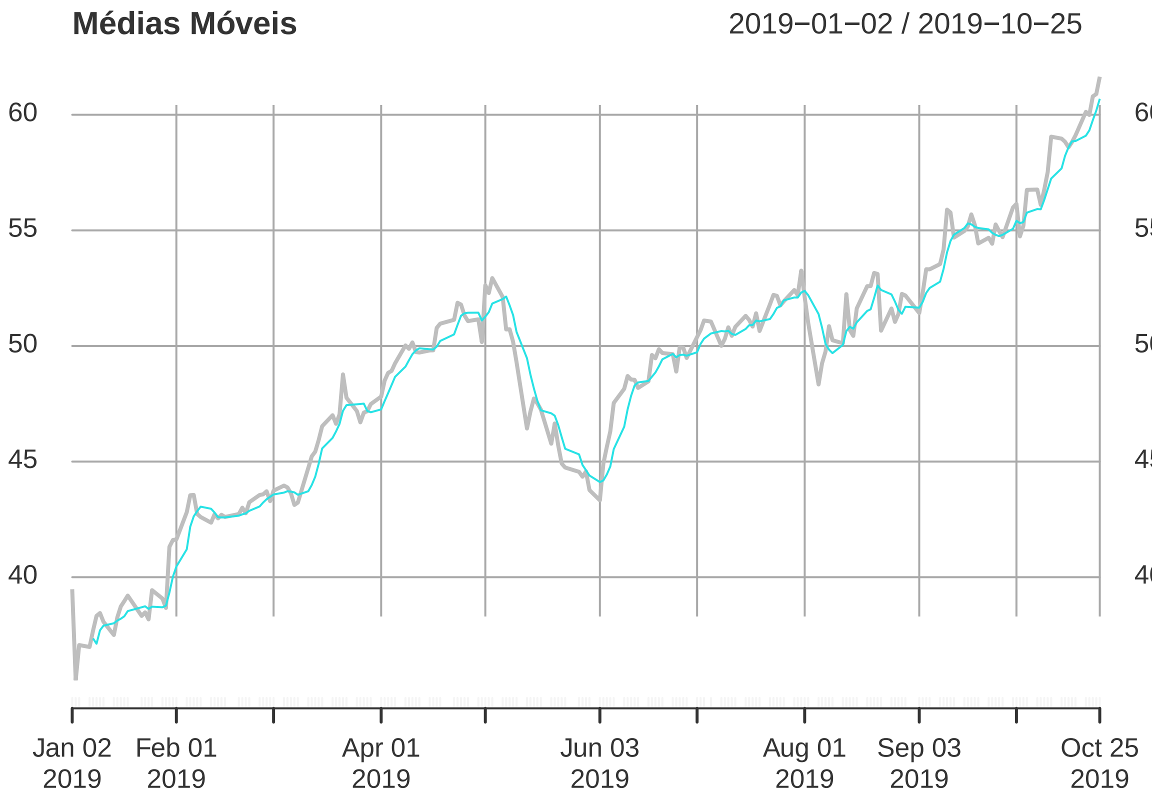


Segundo período .

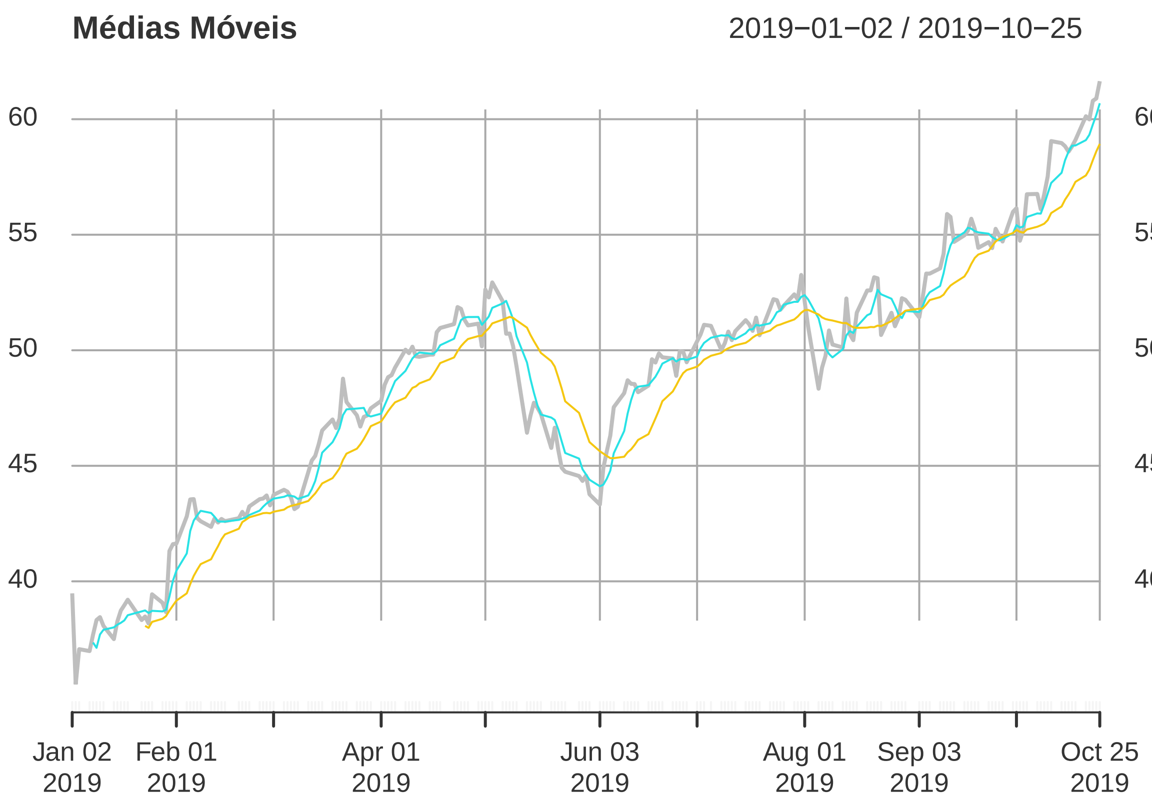
```
mm_aapl_5 <- rollmean(Cl(t2), k = 5, align = c("right"))
mm_aapl_15 <- rollmean(Cl(t2), k = 15, align = c("right"))
mm_aapl_30 <- rollmean(Cl(t2), k = 30, align = c("right"))
plot(Cl(t2), col = "gray", main = "Médias Móveis")
```



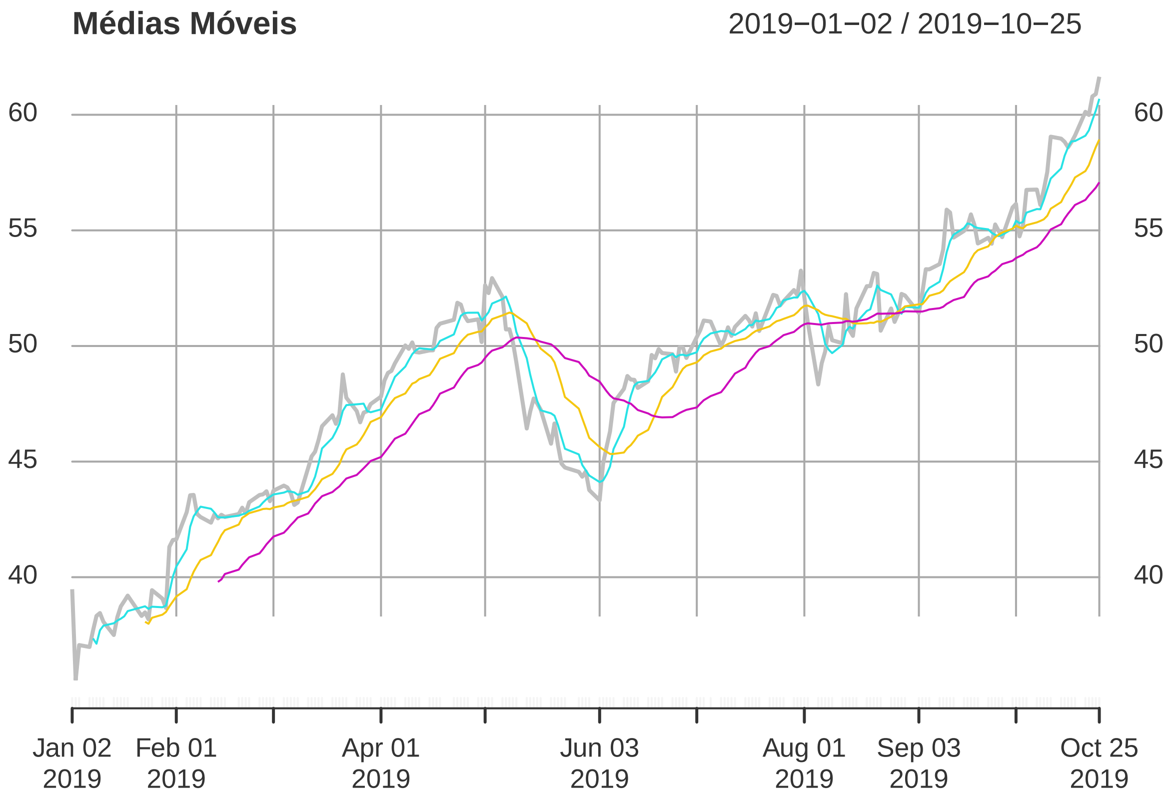

```r
lines(mm_aapl_5, col = 5)
```




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




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



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)

```
n_1 <- length(as.ts(Cl(t1$AAPL.Close))) - 1
ret_simples_t1 <- diff(Cl(t1$AAPL.Close))/Cl(t1$AAPL.Close)[1:(n_1)]

n_2 <- length(as.ts(Cl(t2$AAPL.Close))) - 1
ret_simples_t2 <- diff(Cl(t2$AAPL.Close))/Cl(t2$AAPL.Close)[1:(n_2)]
```

Calculando os retornos simples dos dois períodos ####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)

```
paste("Média do retorno simples primeiro período",
      round(100*mean(ret_simples_t1[2:n_1]),4), "%")
```

Calculando as média dois períodos

```
## [1] "Média do retorno simples primeiro período -0.0561 %"
```

```
paste("Média do retorno simples segundo período",
      round(100*mean(ret_simples_t2[2:n_2]),4), "%")
```

```
## [1] "Média do retorno simples segundo período 0.1955 %"
```

```
paste("Média do log-retorno primeiro período",
      round(100*mean(log_ret_t1[2:n_1]),4), "%")
```

```
## [1] "Média do log-retorno primeiro período -0.0393 %"
```

```
paste("Média do log-retorno segundo período",
      round(100*mean(log_ret_t2[2:n_2]),4), "%")
```

```
## [1] "Média do log-retorno segundo período 0.2685 %"
```

```


paste("Desvio-padrão retorno simples do primeiro período",
      round(sd(ret_simples_t1[2:n_1]),4))
```

Calculando desvios dos dois períodos

```
## [1] "Desvio-padrão retorno simples do primeiro período 0.0182"
```

```
paste("Desvio-padrão retorno simples do segundo período",
      round(sd(ret_simples_t2[2:n_2]),4))
```

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

```
paste("Desvio-padrão log-retorno do primeiro período",
      round(sd(log_ret_t1[2:n_1]),4))
```

```
## [1] "Desvio-padrão log-retorno do primeiro período 0.0182"
```

```
paste("Desvio-padrão log-retorno do segundo período",
      round(sd(log_ret_t2[2:n_2]),4))
```

```
## [1] "Desvio-padrão log-retorno do segundo período 0.016"
```

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

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

```
##
##  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
```

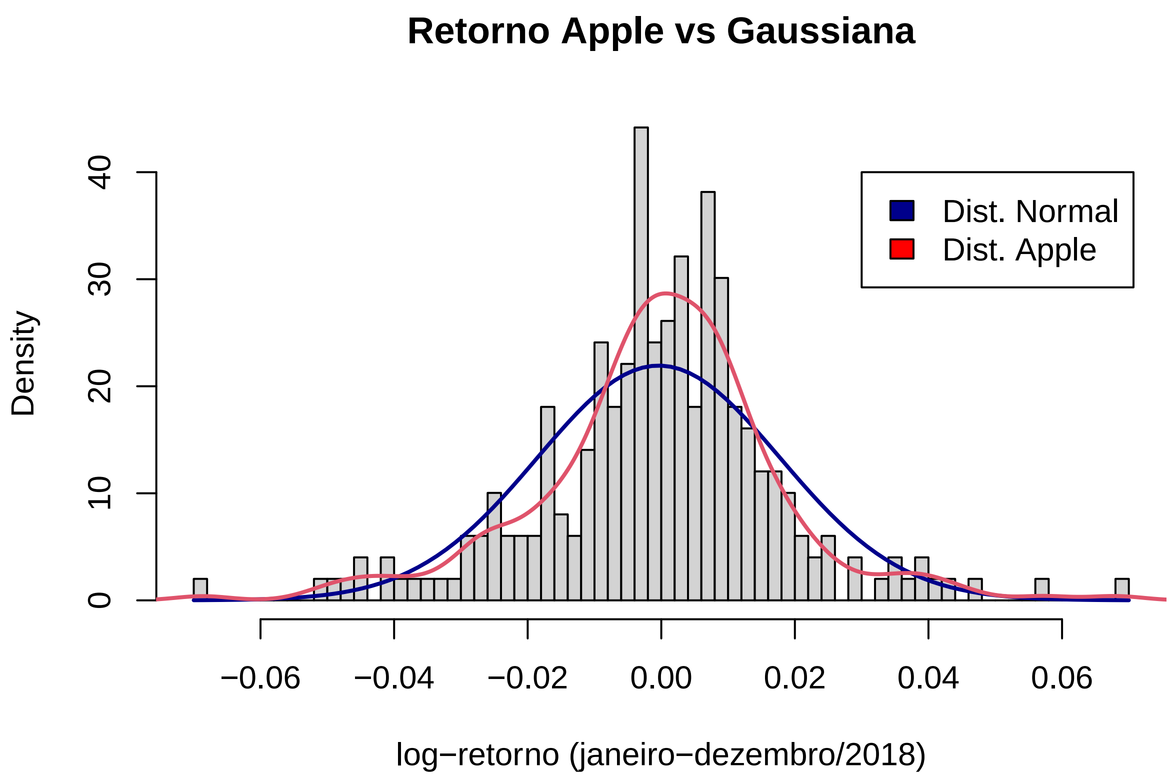
```r
t.test(log_ret_t2)
```

```
##
##  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)

```r
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[2:n_1]),
            sd=sd(log_ret_t1[2:n_1])),
      col="darkblue",
      lwd=2,
      add=TRUE,
```

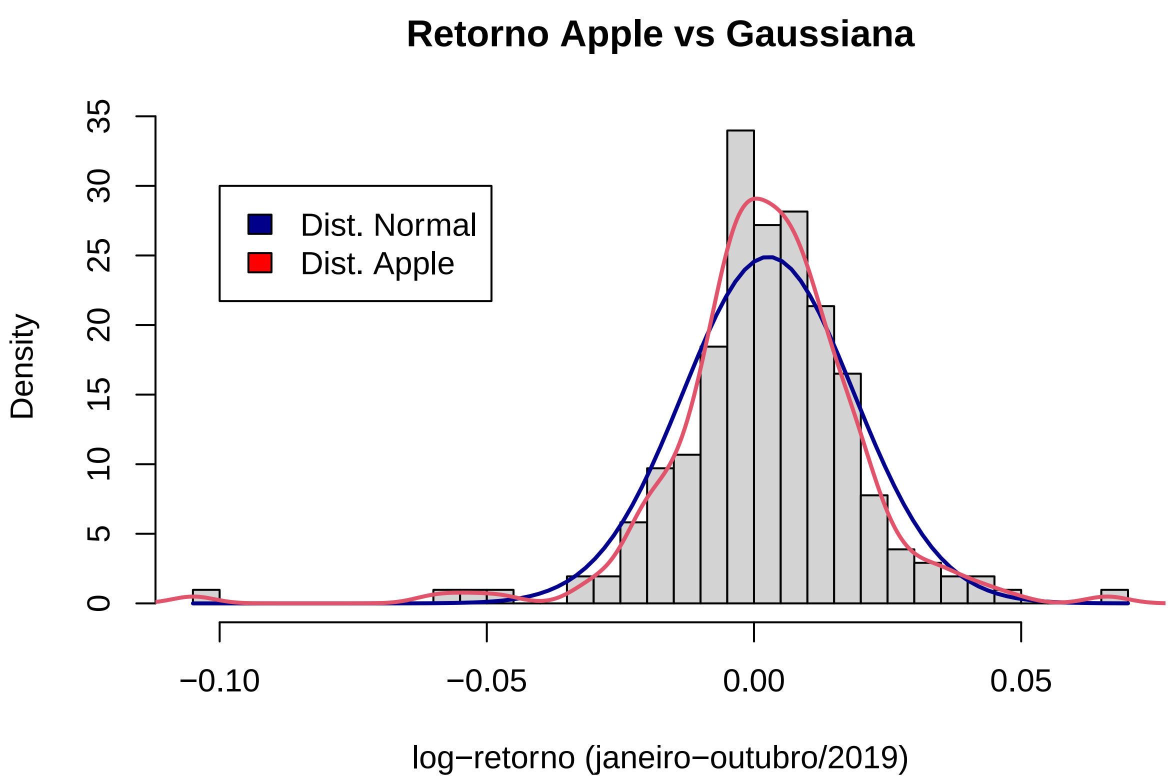
```
       yaxt="n")

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

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



Primeiro 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[2:n_2]),
```

```
            sd=sd(log_ret_t2[2:n_2])),
      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"))
```



Segundo período