

# Séries Temporais: Trabalho I

Mariana Costa Freitas

2025-01-08

## Questão 9 (capítulo 4)

Primeiro vamos carregar os dados dos retornos diários do IBOVESPA, armazenando seus valores no vetor `retornos`.

```
library(dplyr)
```

```
##  
## Anexando pacote: 'dplyr'  
  
## Os seguintes objetos são mascarados por 'package:stats':  
##  
##   filter, lag  
  
## Os seguintes objetos são mascarados por 'package:base':  
##  
##   intersect, setdiff, setequal, union
```

```
library(janitor)
```

```
##  
## Anexando pacote: 'janitor'  
  
## Os seguintes objetos são mascarados por 'package:stats':  
##  
##   chisq.test, fisher.test
```

```
# Dados do IBOVESPA  
url <- "https://www.ime.usp.br/~pam/M-IBV-SP"  
  
# Ler o arquivo usando read.table()  
dados <- read.table(url, header = TRUE) |>  
  clean_names() |>  
  select(ibov)  
  
retornos <- dados$ibov
```

Para a variância condicionar, vamos usar um modelo de Suavização Exponencial Simples (SES), que segue a seguinte equação de recorrência:

$$\sigma_t^2 = \alpha \sigma_{t-1}^2 + (1 - \alpha) r_{t-1}^2$$

Assim, a variância condicional no instante  $t$  é uma combinação linear entre a variância condicional passada) e o quadrado do retorno mais recente. Vamos usar a função:

```
cond_var <- function(alpha, retornos) {
  n <- length(retornos)
  sigma2 <- numeric(n) # Vetor para armazenar a variância
  sigma2[1] <- var(retornos) # Variância inicial (estimativa empírica)

  for (t in 2:n) {
    sigma2[t] <- alpha * sigma2[t-1] + (1 - alpha) * retornos[t-1]^2
  }
  return(sigma2)
}
```

Já o parâmetro  $\alpha$  controla o peso atribuído à variância passada e ao retorno atual; logo quando  $\alpha$  está próximo de 1, a variância condicional atribui maior peso à variância passada, e quando  $\alpha$  está próximo de 0, maior peso é dado ao retorno recente. O valor ótimo de  $\alpha$  é geralmente estimado a partir dos dados históricos, minimizando uma métrica de erro, como o Erro Percentual Absoluto Médio (MAPE).

O Erro Percentual Absoluto Médio (MAPE) é uma métrica utilizada para avaliar a precisão de modelos de previsão, sendo calculado como:

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| \frac{Y_t - \hat{Y}_t}{Y_t} \right|$$

Nesse caso,  $Y_t$  representa o quadrado dos retornos observados e  $\hat{Y}_t$  a variância condicional estimada pelo modelo. Assim, é possível estimar o  $\alpha$  ótimo:

```
mape <- function(alpha, retornos) {
  sigma2 <- cond_var(alpha, retornos)
  error <- abs(retornos^2 - sigma2) / abs(retornos^2)
  return(mean(error, na.rm = TRUE))
}

# Otimização
optim_result <- optim(par = 0.5, fn = mape, retornos = retornos,
                     method = "Brent", lower = 0, upper = 1)
alpha_opt <- optim_result$par
cat("Alpha otimizado:", alpha_opt, "\n")
```

```
## Alpha otimizado: 0.0122113
```

Com  $\alpha$  otimizado, podemos encontrar as previsões das variâncias condicionais:

```
sigma2 <- cond_var(alpha_opt, retornos)
sigma2_t1 <- alpha_opt * sigma2[length(sigma2)] + (1 - alpha_opt) * retornos[length(retornos)]^2
sigma2_t2 <- alpha_opt * sigma2_t1 + (1 - alpha_opt) * retornos[length(retornos)]^2

cat("Variância um passo à frente:", sigma2_t1, "\n")
```

```
## Variância um passo à frente: 25629939
```

```
cat("Variância dois passos à frente:", sigma2_t2, "\n")
```

```
## Variância dois passos à frente: 25551541
```

## Questão 10 (capítulo 4)

### Item a)

Carregando os dados de umidade a serem utilizados:

```
library(readxl)

atmosfera <- read_excel("atmosfera.xls",
                        sheet = "Plan1") |>
  select(-c(`18.39999999999999`))

serie <- atmosfera$umidade
```

Devemos ajustar médias móveis para  $r = 6$ ,  $r = 9$  e  $r = 12$ . Temos a fórmula para a média móvel simples:

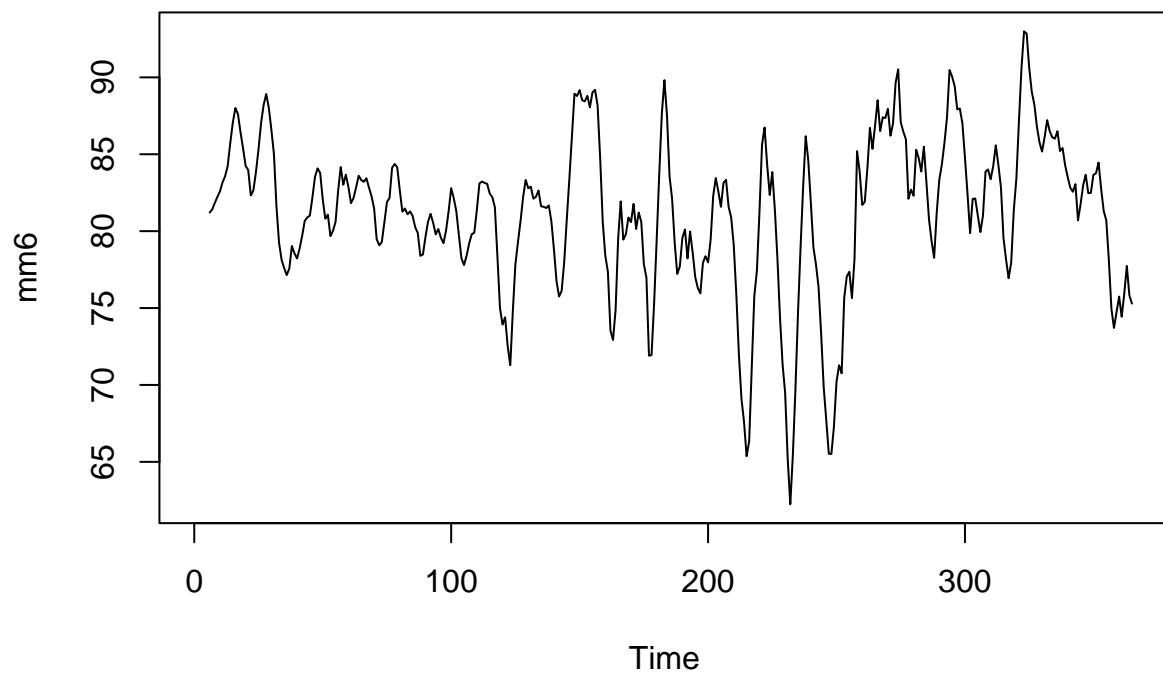
$$\text{Média móvel}_t = \frac{1}{r} \sum_{i=0}^{r-1} X_{t-i}$$

Assim, usamos:

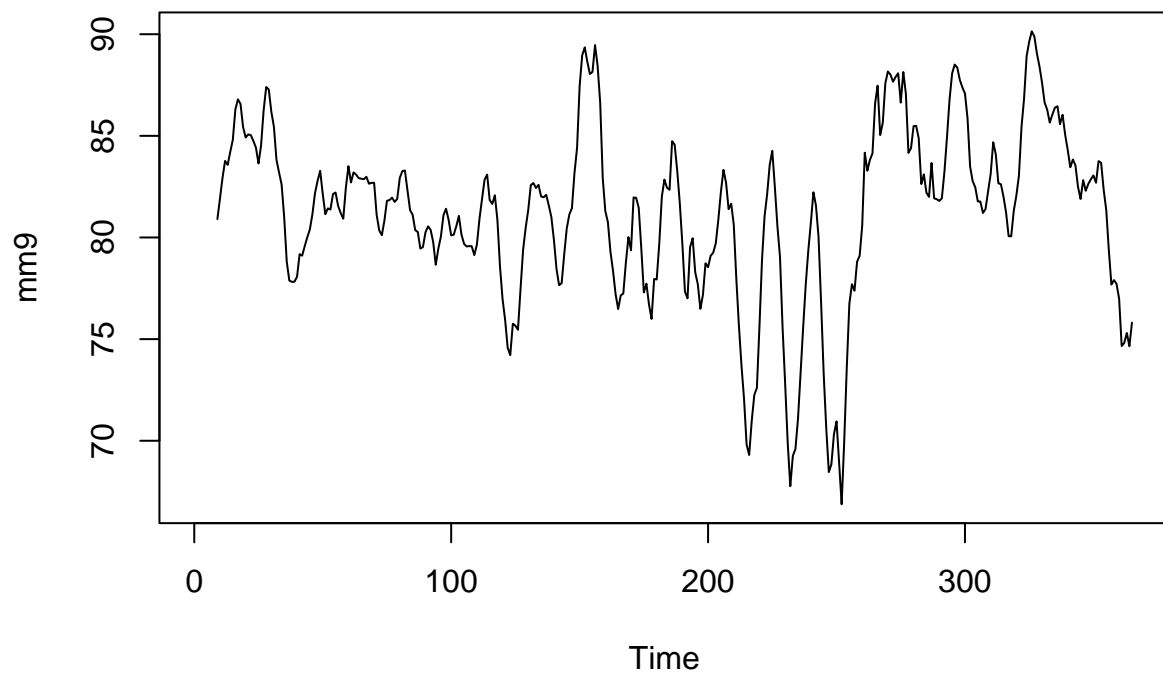
```
# Função para calcular médias móveis
calcular_mm <- function(serie, r) {
  stats::filter(serie, rep(1/r, r), sides = 1)
}

# Cálculo das médias móveis
mm6 <- calcular_mm(serie, 6)
mm9 <- calcular_mm(serie, 9)
mm12 <- calcular_mm(serie, 12)

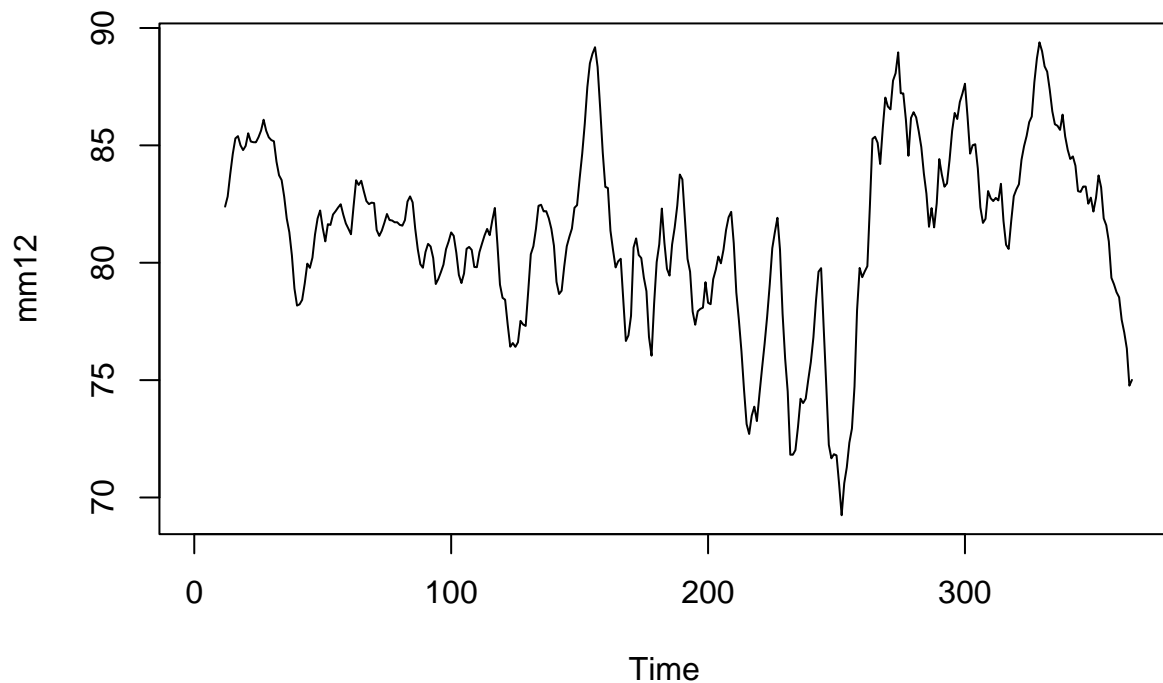
plot(mm6)
```



```
plot(mm9)
```



```
plot(mm12)
```



### Item b)

Para identificar o melhor ajuste, utilizamos uma métrica de erro, como o Erro Médio Absoluto (MAE):

$$MAPE = \frac{1}{n} \sum_{t=1}^n |Y_t - \hat{Y}_t|$$

Assim, usamos:

```
# Calcular MAE para cada média móvel
calcular_mae <- function(obs, previsoes) {
  mean(abs(obs - previsoes), na.rm = TRUE)
}

# Ajustar os dados para as médias móveis
observados <- serie[-(1:5)] # Remover os primeiros valores para alinhar com a média móvel de r=6
mae6 <- calcular_mae(observados, mm6[-(1:5)])
mae9 <- calcular_mae(observados, mm9[-(1:8)])
mae12 <- calcular_mae(observados, mm12[-(1:11)])

# Identificar o melhor modelo
melhor_modelo <- which.min(c(mae6, mae9, mae12))
cat("Melhor modelo:", c("r=6", "r=9", "r=12")[melhor_modelo], "\n")
```

```
## Melhor modelo: r=9
```

## Item c

Com base no modelo que apresentou o menor erro, realizamos as previsões para os meses de janeiro de 1979, fevereiro de 1979, e março de 1979.

```
prever <- function(serie, r) {  
  previsoes <- rep(NA, 3)  
  for (i in 1:3) {  
    ultimos <- tail(serie, r)  
    previsoes[i] <- mean(ultimos)  
    serie <- c(serie, previsoes[i]) # Atualizar série com a previsão  
  }  
  return(previsoes)  
}  
  
# Usar o modelo com melhor ajuste para prever  
if (melhor_modelo == 1) {  
  previsoes <- prever(serie, 6)  
} else if (melhor_modelo == 2) {  
  previsoes <- prever(serie, 9)  
} else {  
  previsoes <- prever(serie, 12)  
}  
  
cat("Previsões para janeiro, fevereiro e março de 1979:", previsoes, "\n")
```

```
## Previsões para janeiro, fevereiro e março de 1979: 75.81222 77.0658 76.41534
```

## Questão 19 (capítulo 3)

### Item a)

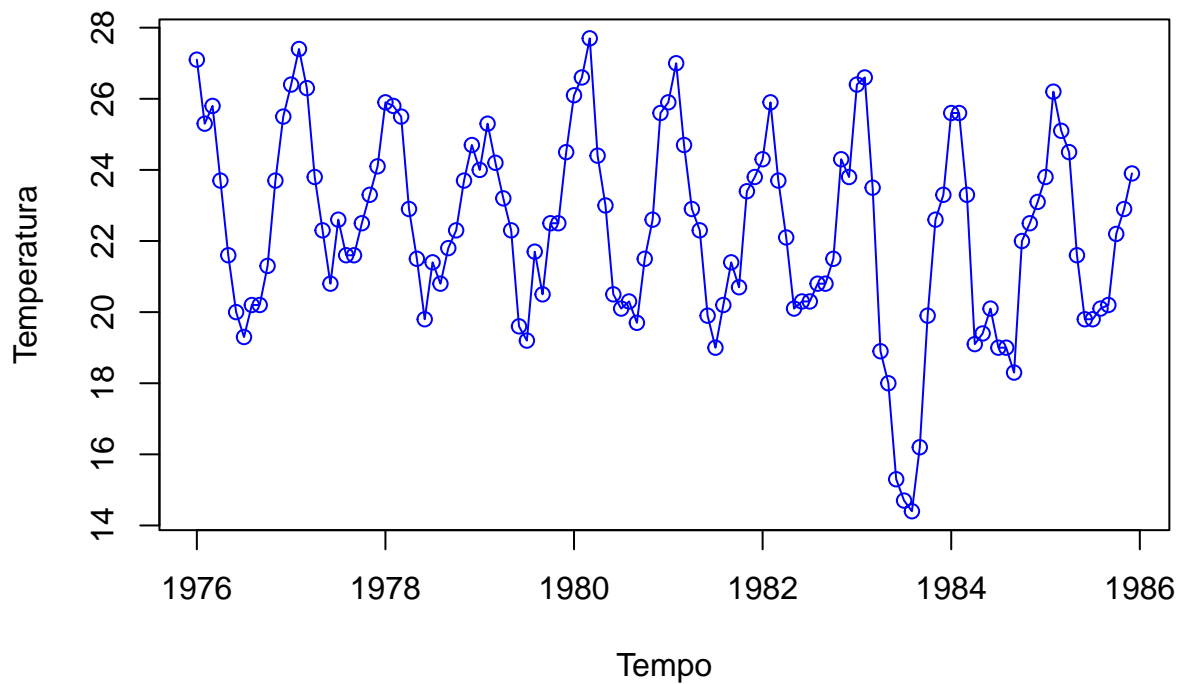
Carregando e organizando os dados:

```
library(readxl)  
temperatura <- read_excel("temperatura.xls") |>  
  clean_names() |>  
  select(ubatuba)  
  
serie <- ts(temperatura, frequency = 12, start = c(1976, 1)) # Dados mensais começando em janeiro de 1976
```

Para fazer a análise visual, vamos plotar a série a seguir:

```
plot(serie, main = "Série Temporal: Temperatura em Ubatuba", ylab = "Temperatura", xlab = "Tempo", col = "blue")
```

## Série Temporal: Temperatura em Ubatuba



Como a série não mostra crescimento (tendência ascendente) nem queda (tendência descendente) ao longo do tempo, assumimos uma ausência de tendência, então não vamos incluir componente de tendência no modelo.

Porém, como temos repetições regulares no padrão dos dados (picos e vales) em intervalos fixos, a série parece apresentar sazonalidade, componente que deve ser incluído no modelo. Também temos algumas flutuações aleatórias, então um ruído será introduzido no modelo.

Apenas com uma análise visual, teríamos o modelo:

$$Z_t = S_t + e_t$$

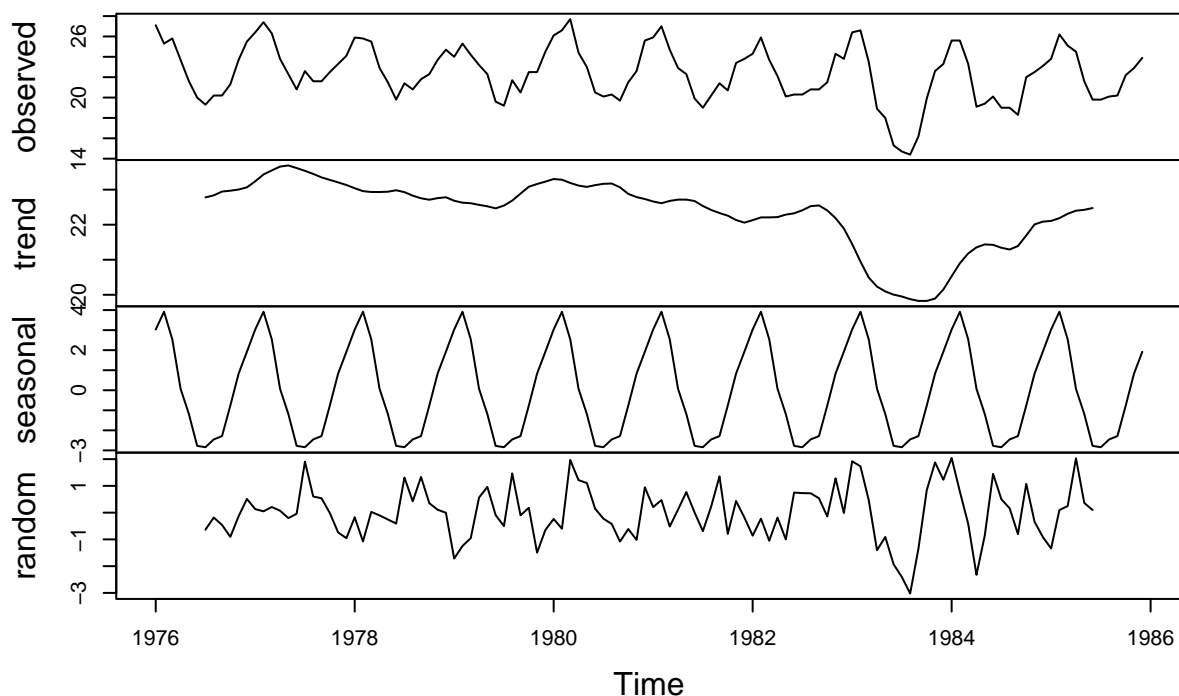
### Item b)

Para analisar os componentes, vamos fazer uma decomposição da série temporal:

```
# Decomposição da série temporal
decomposicao <- decompose(serie, type = "additive")
plot(decomposicao) # Gráfico com todas as componentes
```



## Decomposition of additive time series



Observamos que, na maior parte do tempo, a tendência segue quase uma linha reta, indicando que é praticamente ausente nesse intervalo da série. É possível observar claramente uma sazonalidade, devido a presença de picos e vales em intervalos regulares. Já os resíduos parecem ter um comportamento aleatório.

### Item c)

```
# Teste de tendência
modelo_tendencia <- lm(serie ~ time(serie))
summary(modelo_tendencia)

##
## Call:
## lm(formula = serie ~ time(serie))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -7.3628 -1.8989  0.0131  1.8718  5.1284
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  491.28480   165.15733    2.975  0.00356 **
## time(serie)  -0.23670    0.08337   -2.839  0.00533 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 2.636 on 118 degrees of freedom
## Multiple R-squared:  0.06394,    Adjusted R-squared:  0.05601
## F-statistic: 8.061 on 1 and 118 DF,  p-value: 0.005329
```

Os resultados indicam que a série apresenta uma tendência decrescente significativa ( $p - valor = 0.00356$ ), apesar de explicar uma proporção pequena da variação total na série ( $R^2 = 6.4\%$ ). Assim, o componente de tendência deve ser incluído em modelos futuros, mas a baixa explicação da variância sugere que outros fatores podem ter um papel importante.

```
# Teste de sazonalidade
kruskal.test(serie ~ cycle(serie))
```

```
##
## Kruskal-Wallis rank sum test
##
## data:  serie by cycle(serie)
## Kruskal-Wallis chi-squared = 95.121, df = 11, p-value = 1.643e-15
```

O p-valor extremamente baixo indica que há uma diferença significativa entre os ciclos sazonais da série. Ou seja, os dados apresentam uma sazonalidade estatisticamente significativa, o que implica que a sazonalidade é uma característica importante a ser considerada na modelagem da série temporal.

```
# Teste para ruído branco
Box.test(decomposicao$random, lag = 12, type = "Ljung-Box")
```

```
##
## Box-Ljung test
##
## data:  decomposicao$random
## X-squared = 73.003, df = 12, p-value = 8.747e-11
```

Como temos  $p - valor$  alto, há evidência de autocorrelação significativa nos resíduos da série. Logo, os resíduos não apresentam dependência temporal significativa e podem ser considerados ruído branco.