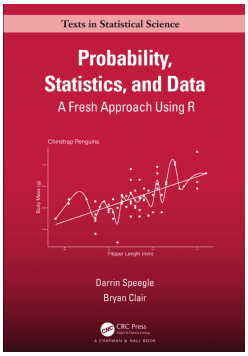
Resolução de Problemas do Livro

Probability, Statistics, and Data: A Fresh Approach Using R (Speegle, D.; Clair, B.)



por

Igo da Costa Andrade

Referência

SPEEGLE, D.; CLAIR, B.. **Probability, Statistics, and Data**: A Fresh Approach Using R. Local, CRC Press, 2022.

Capítulo 1: Dados em R[1]

```
# Bibliotecas Necessárias
library(tidyverse)
library(knitr)
library(kableExtra)
library(latex2exp)
library(fosdata)
library(HistData)
library(eaf)
library(tikzDevice)
options(tikzMetricPackages = c("\\usepackage{amsmath}",
                               "\\usepackage{xcolor}",
                               "\\usepackage{tikz}",
                               "\\usetikzlibrary{calc}"))
```

Exercícios

1.4 Neste exercício, construa o gráfico da função $f(p) = p(1-p)$ para $p \in [0, 1]$.

a. Use `seq` para criar um vetor p de números de 0 a 1 espaçados por 0.2.

b. Use a função `plot` para graficar p na coordenada x e $p(1-p)$ na coordenada y. Leia a página de ajuda para `plot` e faça testes com o argumento `type` para encontrar uma boa escolha para este gráfico.

c. Repita, mas criando um vetor p de números de 0 a 1 espaçados por 0.01.

Solução:

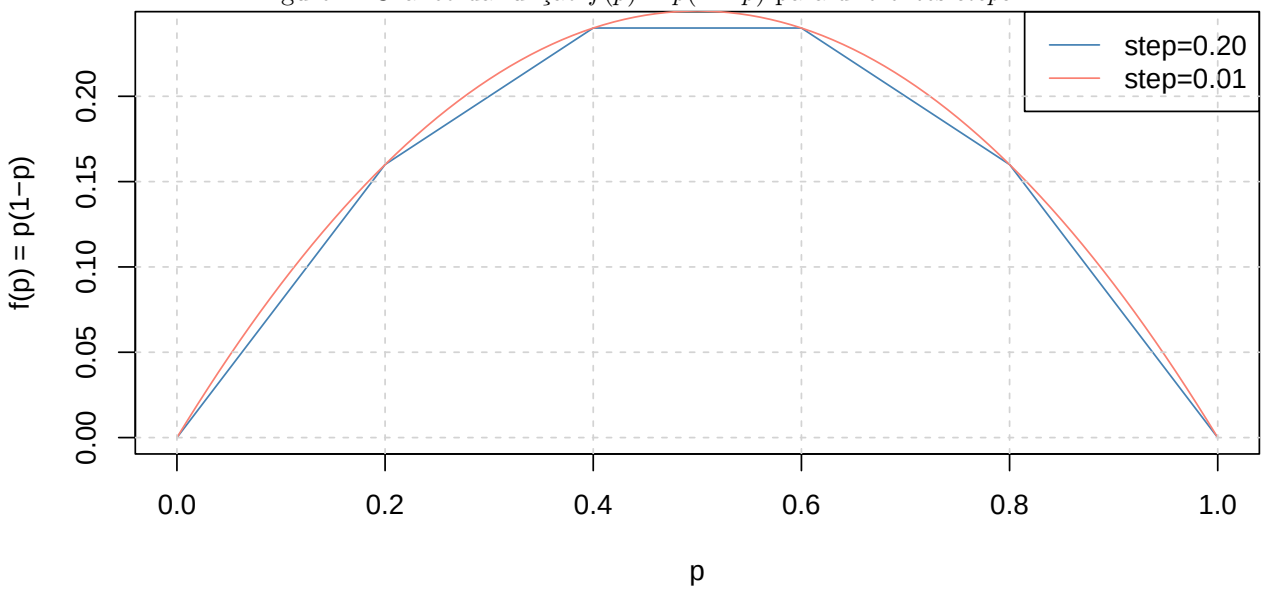
Vetor p com *step* igual a 0.1:

```
p1 <- seq(0, 1, by=0.2)
Fp1 <- p1 * (1-p1)
```

[1] Título original: *Data in R*

```r
p2 <- seq(0, 1, by=0.01)
Fp2 <- p2 * (1-p2)


pdf(file = "figure/chap-01/problema-1.4.pdf",
    width = 8,
    height = 4.5)
plot(
  p1, Fp1, type="l", lty=1, col="steelblue",
  xlab="p", ylab="f(p) = p(1-p)",
)
lines(
  p2, Fp2, lty=1, col="salmon", xlab="", ylab=""
)
legend(x="topright", lty=c(1, 1),
       legend = c("step=0.20", "step=0.01"),
       col=c("steelblue", "salmon"),
)
grid(lty="dashed")

dev.off()
eaf::pdf_crop("figure/chap-01/problema-1.4.pdf")
```

Figure 1: Gráfico da função $f(p) = p(1-p)$ para diferentes *steps*



1.5 Use `R` para calcular a sima dos quadrados de todos os números de 1 a 100: $1^2 + 2^2 + \cdots + 99^2 + 100^2$.

Solução:

```r
soma <- sum((1:100)^2)
print(soma)
```

```
## [1] 338350
```

$$\sum_{i=1}^{100} x_i = 1^2 + 2^2 + \cdots + 99^2 + 100^2 = 338.350$$

■

1.6 Seja x o vetor obtido da execução do comando `R x <- seq(from=10, to=30, by=2)`.

a. Qual é o comprimento de `x`?
b. O que é `x[2]`?
c. O que é `x[1:5]`?
d. O que é `x[1:3*2]`?
e. O que é `x[1:(3*2)]`?
f. O que é `x > 25`?
g. O que é `x[x > 25]`?
h. O que é `x[-1]`?
i. O que é `x[-1:-3]`?

Solução:

```r
# Definição do vetor x
x <- seq(from = 10, to = 30, by = 2)
x
```

```
##  [1] 10 12 14 16 18 20 22 24 26 28 30
```

```r
# a. Qual é o comprimento de x?
length(x)
```

```
## [1] 11
```

O comprimento do vetor `x`, ou seja a quantidade de elementos desse vetor é igual a `length(x)` $= 11$.

```r
# b. O que é x[2]?
x[2]
```

```
## [1] 12
```

`x[2]` é o segundo elemento do vetor `x`, e seu valor é `x[2]` $= 12$.

```r
# c. O que é x[1:5]?
x[1:5]
```

```
## [1] 10 12 14 16 18
```

`x[1:5]` é um subconjunto do vetor `x` represetado pelos elementos desde a primeira posição até a quinta posição.

```r
# d. O que é x[1:3*2]?
x[1:3*2]
```

```
## [1] 12 16 20
```

`x[1:3*2]` é um subconjunto do vetor `x` representado pelos elementos nas posições:

$$\texttt{1:3*2} = \texttt{c(1, 2, 3)*2} = \texttt{c(2, 4, 6)}$$

```r
# e. O que é x[1:(3*2)]?
x[1:(3*2)]
```

```
## [1] 10 12 14 16 18 20
```

`x[1:(3*2)]` é o subconjunto de `x` representado pelos elementos de 1 até 6, visto que:

$$\texttt{1:(3*2)} = \texttt{1:6} = \texttt{c(1, 2, 3, 4, 5, 6)}$$

```r
# f. O que é x > 25?
x > 25
```

```
##  [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE
```

`x > 25` é um vetor lógico (booleano), resultado da vefificação para cada elemento de `x` se o referido elemento é maior que 25 (`TRUE`) ou não (`FALSE`).

```r
# g. O que é x[x > 25]?
x[x > 25]
```

```
## [1] 26 28 30
```

`x[x > 25]` é um subconjunto de `x` representado pelos elementos de `x` que são maiores que 25.

```r
# h. O que é x[-1]?
x[-1]
```

```
##  [1] 12 14 16 18 20 22 24 26 28 30
```

`x[-1]` é o subconjunto de `x` após a exclusão do primeiro elemento.

```r
# i. O que é x[-1:-3]?
x[-1:-3]
```

```
## [1] 16 18 20 22 24 26 28 30
```

`x[-1:-3]` é o subconjunto de `x` após a exclusão dos elementos nas posições 1, 2, e 3. ■

1.7 `R` possui uma vetor denominado `rivers` o qual contém os comprimentos do maiores rios Norte Americanos.

a. Use `?rivers` para apreender sobre o conjunto de dados.
b. Encontre a média e o desvio padrão dos dados dos rios usando as funções de `R` base `mean` e `sd`.
c. Crie um histogram (`hist`) dos dados de rios.
d. Obtenha cinco números resumos (`summary`) dos dados dos rios.
e. Encontre o maior e o menor comprimento de rios do conjunto.

f. Crie uma lista de todos os (comprimentos de) rios maiores que 1000 milhas.

Solução:

```
# b1. Média do comprimento dos rios
rivers_mean <- mean(rivers)
rivers_mean
```

```
## [1] 591.1844
```

$$\overline{x}_{\text{rivers}} = \sum_{i=1}^{n=141} \frac{x_i}{n} = \frac{735 + 320 + \cdots + 1.770}{141} = \frac{83.357}{141} = 591,18 \text{ milhas}$$

```
# b2. Desvio Padrão do comprimento dos rios
rivers_sd <- sd(rivers)
rivers_sd
```

```
## [1] 493.8708
```

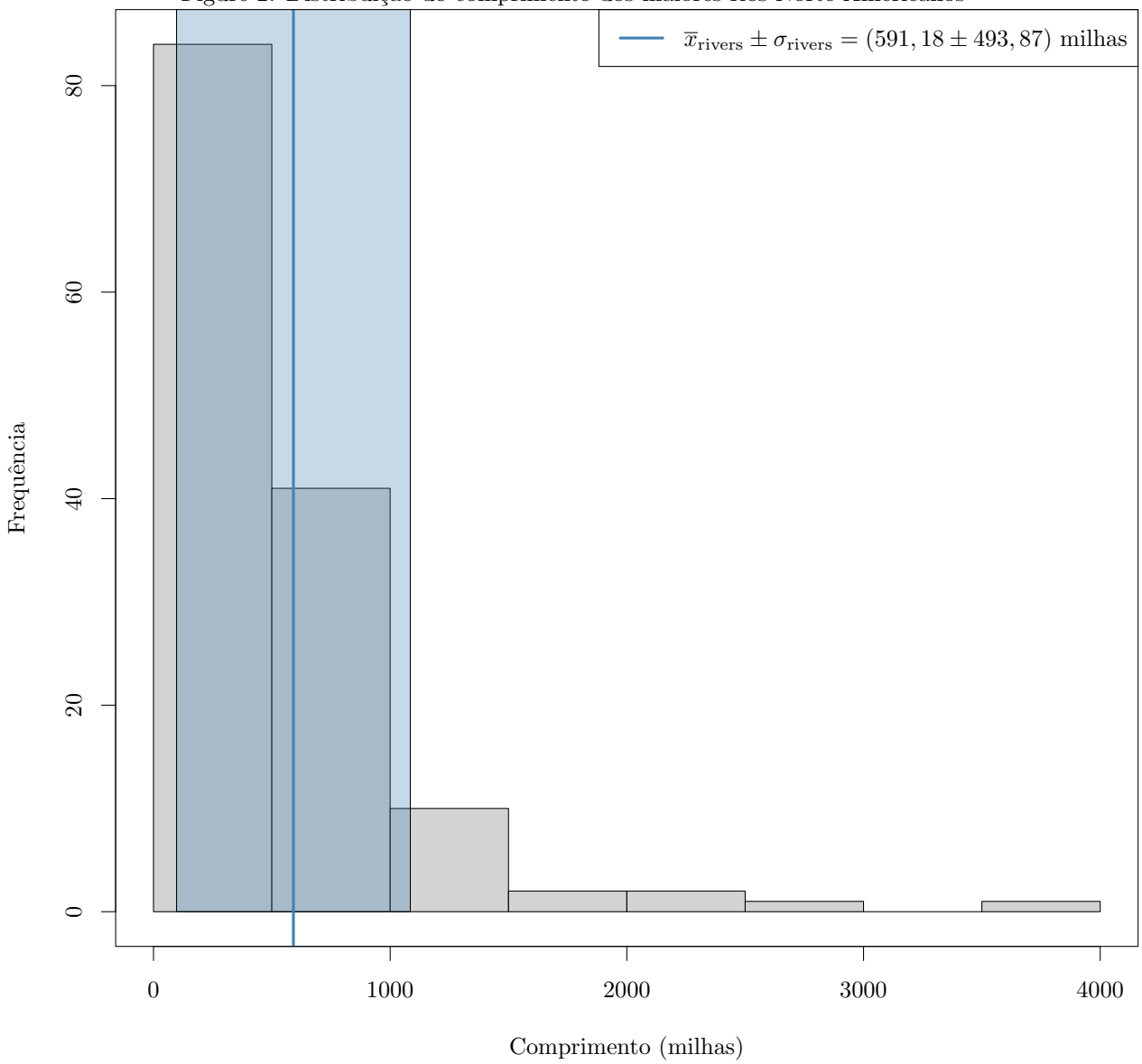
$$\sigma_{\text{rivers}} = \sqrt{\sum_{i=1}^{n=141} \frac{(x_i - \overline{x}_{\text{rivers}})^2}{n}} = \sqrt{\frac{(735 - 591,18)^2 + (320 - 591,18)^2 + \cdots + (1.770 - 591,18)^2}{141}}$$

$$= 493,87 \text{ milhas}$$

```
# c. Histograma da distribuição de comprimentos de rios
tikz("tex/chap-01/problema-1.7c.tex",standAlone = TRUE,
     packages=c("\\usepackage{amsmath}",
                "\\usepackage{tikz}",
                "\\usepackage{xcolor}",
                "\\usetikzlibrary{calc}",
                "\\usepackage[active,tightpage,psfixbb]{preview}",
                "\\PreviewEnvironment{pgfpicture}"))

hist(rivers,
     xlab="Comprimento (milhas)", ylab="Frequência", main=""
);

rect(
  xleft=rivers_mean-rivers_sd, xright=rivers_mean+rivers_sd,
  ybottom=0, ytop=100, col= rgb(0.2745,0.5098, 0.7059,alpha=0.3)
)

abline(v=rivers_mean, lty=1, lwd=3, col="steelblue")
legenda <- sprintf(
  "$\\overline{x}_{\\text{rivers}} \\pm \\sigma_{\\text{rivers}} = (%s \\pm %s)$ milhas",
  fmt(rivers_mean), fmt(rivers_sd))

legend(x="topright", lty=c(1), lwd=c(3),
       legend = c(legenda),
```

```
       col="steelblue",
)

box()

dev.off();
tools::texi2pdf("tex/chap-01/problema-1.7c.tex", clea=TRUE)
system(paste(getOption("pdfviewer"), "tex/chap-01/problema-1.7c.tex", sep=""));
file.rename("problema-1.7c.pdf", "figure/chap-01/problema-1.7c.pdf")
eaf::pdf_crop("figure/chap-01/problema-1.7c.pdf")
```

Figure 2: Distribuição do comprimento dos maiores rios Norte Americanos




```
# d. Medidas Resumo dos dados de rios
summary(rivers)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   135.0   310.0   425.0   591.2   680.0  3710.0
```

```
# e. Maior e Menor comprimentos de rios
rivers_max <- max(rivers)
rivers_min <- min(rivers)
```

O maior comprimento de rio do conjunto de dados foi de 3.710 milhas enquanto o menor comprimento foi de 135.

```
# f. Lista de comprimentos de rios maiores que 1000 milhas
rivers[rivers > 1000]
```

```
##  [1] 1459 1450 1243 2348 1171 3710 2315 2533 1306 1054 1270 1885 1100 1205 1038
## [16] 1770
```

1.8 Considere o conjunto de dados `airquality`.

a. Quantas observações de quantas variáveis existem?

b. Quais os nomes das variáveis?

c. Qual o tipo de dados de cada variável?

d. Você concorda com o tipo de dados associado a cada variável? Existem escolhas melhores?

Solução:

```
# a.
n_row <- nrow(airquality)
n_col <- ncol(airquality)
```

O conjunto de dados `airquality` consiste em 153 observações de 6 variáveis.

```
# b.
col_names <- colnames(airquality)
col_names
```

```
## [1] "Ozone"   "Solar.R" "Wind"    "Temp"    "Month"   "Day"
```

Os nomes das variáveis são Ozone, Solar.R, Wind, Temp, Month, e Day.

```
# c.
tbl <- sprintf("%8s %8s", "Variável", "Tipo")
tbl <- sprintf("%s\n%s", tbl, "================")
for (name in col_names) {
  row <- sprintf("%8s %8s", name, class(airquality[[name]]))
  tbl <- sprintf("%s\n%s", tbl, row)
}

cat(tbl)
```

```
## Variável     Tipo
## =================
##    Ozone  integer
##  Solar.R  integer
##     Wind  numeric
##     Temp  integer
##    Month  integer
##      Day  integer
```

```r
# d.

head(airquality)
```

```
##   Ozone Solar.R Wind Temp Month Day
## 1    41     190  7.4   67     5   1
## 2    36     118  8.0   72     5   2
## 3    12     149 12.6   74     5   3
## 4    18     313 11.5   62     5   4
## 5    NA      NA 14.3   56     5   5
## 6    28      NA 14.9   66     5   6
```

A variável `Month` poderia ser melhor descrita como do tipo `factor`. ■