Introdução ao programa R

Ronald Targino, DEMA-UFC

Notas de aula

2.3 Data frames

```
# Parte 1
# ---------------------------------------------------------------------

# Data frame é uma generalização de uma matriz. Permite que as colunas sejam
# de diferentes tipos (numéricas, lógicas, alfanuméricas...). Em geral, essa
# classe de objetos é utilizada para bases (conjuntos) de dados.

# Função data: carrega os conjuntos de dados especificados ou lista os
# conjuntos de dados disponíveis.

data()  # lista dos conjuntos de dados disponíveis
data("AirPassengers")  # carrega a base de dados AirPassengers
# AirPassengers contém os números mensais de passageiros de companhias
# aéreas no período de 1949 a 1960.
library(dslabs)  # carrega o pacote dslabs que contem a base de dados murders.
data(murders)  # carrega a base de dados murders
# murders contém o número de assassinatos por armas de fogo nos estados
# americanos no ano de 2010.
str(murders)  # estrutura da base de dados
head(murders)  # apresenta as primeiras linhas da base de dados
tail(murders)  # apresentas as últimas linhas da base de dados
names(murders)  # nome das variáveis (cabeçalho)
```

```
# Gerando vetores para construir o data frame
rm(list = ls())
set.seed(12)
v1 = sample(1:6, 14, replace = TRUE)
set.seed(13)
v2 = sample(1:6, 14, replace = TRUE)
set.seed(14)
v3 = letters[sample(1:10, 7, replace = TRUE)]
v4 = paste0(letters[1:7], 1:7)
d0 = data.frame(v1, v2, v3, v4)  # o cabeçalho é dado pelo nome dos objetos
d0  # Atenção!
```

```
##    v1 v2 v3 v4
## 1   2  3  i a1
## 2   2  5  i b2
## 3   3  2  d c3
## 4   6  5  d d4
## 5   5  6  j e5
## 6   5  6  a f6
## 7   4  4  i g7
## 8   2  5  i a1
## 9   3  4  i b2
```

```
## 10  2  3  d c3
## 11  5  1  d d4
## 12  2  2  j e5
## 13  1  5  a f6
## 14  6  4  i g7
```

```
colnames(d0)  # cabeçalho do banco de dados; nome das variáveis.
```

```
## [1] "v1" "v2" "v3" "v4"
```

```
dim(d0)  # dimensão do data.frame d0
```

```
## [1] 14  4
```

```
str(d0)  # estrutura do data.frame d0
```

```
## 'data.frame':    14 obs. of  4 variables:
##  $ v1: int  2 2 3 6 5 5 4 2 3 2 ...
##  $ v2: int  3 5 2 5 6 6 4 5 4 3 ...
##  $ v3: Factor w/ 4 levels "a","d","i","j": 3 3 2 2 4 1 3 3 3 2 ...
##  $ v4: Factor w/ 7 levels "a1","b2","c3",..: 1 2 3 4 5 6 7 1 2 3 ...
```

```
# Acesso aos dados e alteração dos dados
d0[1:2, c(2, 4)]
```

```
##   v2 v4
## 1  3 a1
## 2  5 b2
```

```
d0[c(1, 3), ]
```

```
##   v1 v2 v3 v4
## 1  2  3  i a1
## 3  3  2  d c3
```

```
d0[1:5, 1] = 999
d0
```

```
##     v1 v2 v3 v4
## 1  999  3  i a1
## 2  999  5  i b2
## 3  999  2  d c3
## 4  999  5  d d4
## 5  999  6  j e5
## 6    5  6  a f6
## 7    4  4  i g7
## 8    2  5  i a1
## 9    3  4  i b2
## 10   2  3  d c3
## 11   5  1  d d4
## 12   2  2  j e5
## 13   1  5  a f6
## 14   6  4  i g7
```

```
# Cálculo de medidas resumo
mean(v1)  # média
```

```
## [1] 3.428571
```

```r
median(v1)  # mediana
```

```
## [1] 3
```

```r
sd(v1)  # desvio padrão
```

```
## [1] 1.696797
```

```r
var(v1)  # variância
```

```
## [1] 2.879121
```

```r
max(v1)  # máximo
```

```
## [1] 6
```

```r
min(v1)  # mínimo
```

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

```r
quantile(v1)  # mínimo, quartis e máximo
```

```
##   0%  25%  50%  75% 100%
##    1    2    3    5    6
```

```r
quantile(v1, probs = c(0.25, 0.6))  # percentis
```

```
## 25% 60%
## 2.0 3.8
```

```r
summary(v1)  # algumas medidas resumo
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   2.000   3.000   3.429   5.000   6.000
```

```r
ls()  # lista de objetos disponíveis
```

```
## [1] "d0" "v1" "v2" "v3" "v4"
```

```r
# Atenção!
mean(v1)  # objeto v1
```

```
## [1] 3.428571
```

```r
mean(d0$v1)  # objeto v1 do d0
```

```
## [1] 358.9286
```

```r
# Para acessar diretamente o v1 do d0: função attach.
attach(d0)  # Neste exemplo, teremos um conflito entre os objetos.
```

```
## The following objects are masked _by_ .GlobalEnv:
##
##     v1, v2, v3, v4
```

```r
ls()
```

```
## [1] "d0" "v1" "v2" "v3" "v4"
```

```r
# Para eliminar o conflito, use uma das opões: (a) remover os objetos v1,
# v2, v3 e v4; (b) usar a função 'with'; (c) especificar a coluna ou a
# variável do banco de dados; (d) trocar o nome das variáveis no banco de
# dados.
with(d0, mean(v1))
```

```
## [1] 358.9286
```

```
mean(d0[, 1])  # especificando coluna referente ao v1
```

```
## [1] 358.9286
```

```
mean(d0$v1)  # especificando a variável em d0
```

```
## [1] 358.9286
```

```
colnames(d0) = c("renda", "empregado", "setor", "filial")
d0[1:7, ]
```

```
##   renda empregado setor filial
## 1   999         3     i     a1
## 2   999         5     i     b2
## 3   999         2     d     c3
## 4   999         5     d     d4
## 5   999         6     j     e5
## 6     5         6     a     f6
## 7     4         4     i     g7
```

```
detach(d0)  # retirando d0 do caminho de procura (em geral, uma boa ação!)
attach(d0)  # adicionando d0 ao caminho de procura (agora com colunas renomeadas)
mean(renda)
```

```
## [1] 358.9286
```

```
mean(v1)
```

```
## [1] 3.428571
```

```
detach(d0)

# Parte 2
# ------------------------------------------------------------------------
# Nomeando colunas e editando o banco de dados
d1 = data.frame(80:86, letters[1:7], paste0(letters[1:7], 1:7))
d1
```

```
##   X80.86 letters.1.7. paste0.letters.1.7...1.7.
## 1     80            a                        a1
## 2     81            b                        b2
## 3     82            c                        c3
## 4     83            d                        d4
## 5     84            e                        e5
## 6     85            f                        f6
## 7     86            g                        g7
```

```
colnames(d1)  # nome das variáveis (cabeçalho do banco de dados)
```

```
## [1] "X80.86"                    "letters.1.7."
## [3] "paste0.letters.1.7...1.7."
```

```
d1 = data.frame(q1 = 80:86, q2 = letters[1:7], q3 = paste0(letters[1:7], 1:7))
d1
```

```
##   q1 q2 q3
## 1 80  a a1
```

```
## 2 81  b b2
## 3 82  c c3
## 4 83  d d4
## 5 84  e e5
## 6 85  f f6
## 7 86  g g7
```

```
colnames(d1)  # nome das variáveis (cabeçalho do banco de dados)
```

```
## [1] "q1" "q2" "q3"
```

```
str(d1)  # estrutura da banco de dados
```

```
## 'data.frame':    7 obs. of  3 variables:
##  $ q1: int  80 81 82 83 84 85 86
##  $ q2: Factor w/ 7 levels "a","b","c","d",..: 1 2 3 4 5 6 7
##  $ q3: Factor w/ 7 levels "a1","b2","c3",..: 1 2 3 4 5 6 7
```

```
is(d1)  # tipo do objeto
```

```
## [1] "data.frame" "list"       "oldClass"   "vector"
```

```
d1 = edit(d1)  # editar o banco de dados d1
d1
d2 = edit(data.frame())  # editar um novo banco de dados
d2
```

```
# Parte 3
# ------------------------------------------------------------------------
# Algumas funções

set.seed(12)
a = rpois(15, 160)  # 10 números aleatórios da Poisson(4)
set.seed(123)
p = round(rnorm(15, 60, 3), 1)  # 10 números aleatórios da Normal(60,9)
set.seed(1234)
s = sample(letters[1:5], 15, replace = TRUE)  # amostra de tamanho 10 com repetição
set.seed(12345)
f = sample(1:20, 15, replace = FALSE)  # amostra de tamanho 10 sem repetição

d3 = data.frame(a, p, s, f)  # o cabeçalho é dado pelo nome dos objetos
d3
```

```
##      a    p s  f
## 1  141 58.3 d 14
## 2  152 59.3 b 19
## 3  148 64.7 e 16
## 4  129 60.2 d 11
## 5  156 60.4 a 18
## 6  158 65.1 e  8
## 7  161 61.4 d  2
## 8  144 56.2 b  6
## 9  143 57.9 b 17
## 10 152 58.7 d 13
## 11 158 63.7 d  7
## 12 151 61.1 d  1
## 13 171 61.2 e 15
## 14 179 60.3 d 10
```

```
## 15 172 58.3 c 12
```

```
summary(d3)
```

```
##        a               p         s           f
##  Min.   :129.0   Min.   :56.20   a:1   Min.   : 1.00
##  1st Qu.:146.0   1st Qu.:58.50   b:3   1st Qu.: 7.50
##  Median :152.0   Median :60.30   c:1   Median :12.00
##  Mean   :154.3   Mean   :60.45   d:7   Mean   :11.27
##  3rd Qu.:159.5   3rd Qu.:61.30   e:3   3rd Qu.:15.50
##  Max.   :179.0   Max.   :65.10         Max.   :19.00
```

```
colMeans(d3[, 1:2])  # médias das colunas 2 e 3 de d3
```

```
##         a         p
## 154.33333  60.45333
```

```
colSums(d3[, 1:2])  # somas das colunas 2 e 3 de d3
```

```
##      a      p
## 2315.0  906.8
```

```
which(d3[, 1] == 179)  # identifica, na coluna 1 de d3, a posição do registro 179
```

```
## [1] 14
```

```
which(d3$a == 179)  # identifica a posição do registro 179 da variável 'a' em d3
```

```
## [1] 14
```

```
which(d3[1, ] == "a")  # identifica, na linha 1 de d3, a posição do registro 'a'
```

```
## integer(0)
```

```
# identifica, na coluna 1 de d3, as posições com registros menores que 150
which(d3[, 1] < 150)
```

```
## [1] 1 3 4 8 9
```

```
# identifica, na coluna 3 de d3, as posições com registros diferentes de 'd'
which(d3[, 3] != "d")
```

```
## [1]  2  3  5  6  8  9 13 15
```

```
# identifica, em d3, as posições(linhas e colunas) com registros iguais a
# 'd'
which(d3 == "d", arr.ind = TRUE)
```

```
##      row col
## [1,]   1   3
## [2,]   4   3
## [3,]   7   3
## [4,]  10   3
## [5,]  11   3
## [6,]  12   3
## [7,]  14   3
```

```
d3 = data.frame(altura = a, peso = p, setor = s, filial = f)  # alterando o cabeçalho
attach(d3)  # identifica as colunas de d3 pelo nome

mean(peso)
```

```
## [1] 60.45333
```

```
median(altura)
```

```
## [1] 152
```

```
peso + 10
```

```
##  [1] 68.3 69.3 74.7 70.2 70.4 75.1 71.4 66.2 67.9 68.7 73.7 71.1 71.2 70.3
## [15] 68.3
```

```
setor[7]  # sétima observação do setor
```

```
## [1] d
## Levels: a b c d e
```

```
peso^2
```

```
##  [1] 3398.89 3516.49 4186.09 3624.04 3648.16 4238.01 3769.96 3158.44
##  [9] 3352.41 3445.69 4057.69 3733.21 3745.44 3636.09 3398.89
```

```
sum(filial)
```

```
## [1] 169
```

```
min(altura)
```

```
## [1] 129
```

```
max(altura)
```

```
## [1] 179
```

```
sort(peso)  # ordena o vetor 'peso'
```

```
##  [1] 56.2 57.9 58.3 58.3 58.7 59.3 60.2 60.3 60.4 61.1 61.2 61.4 63.7 64.7
## [15] 65.1
```

```
tabela = table(setor)  # distribuição de frequência do vetor setor
tabela
```

```
## setor
## a b c d e
## 1 3 1 7 3
```

```
# Instalar pacote "xtable" (caso não esteja instalado)
# install.packages("xtable", dependencies = TRUE) # instalar pacote "xtable"

# Carregar o pacote para uso
require(xtable)
```

```
# Converte o objeto tabela em código latex
xtable(tabela)
```

```
## % latex table generated in R 3.6.1 by xtable 1.8-4 package
## % Wed Mar 11 20:22:18 2020
## \begin{table}[ht]
## \centering
## \begin{tabular}{rr}
##   \hline
##  & setor \\
##   \hline
```

```
## a &   1 \\
##   b &   3 \\
##   c &   1 \\
##   d &   7 \\
##   e &   3 \\
##    \hline
## \end{tabular}
## \end{table}
```