



PROGRAMAÇÃO EM R

# Funções Apply no R

Aprenda a processar dados de forma eficiente usando as funções apply, lapply e sapply para iteração implícita em estruturas de dados.



Eduardo Ogasawara

[eduardo.ogasawara@cefet-rj.br](mailto:eduardo.ogasawara@cefet-rj.br)

<https://eic.cefet-rj.br/~eogasawara>

## Loops Implícitos – `apply`, `lapply`

### `lapply` e `sapply`

Executam uma função para cada coluna de um data frame automaticamente, sem necessidade de loops explícitos.

**O primeiro caractere define o tipo de retorno:**

- **l** (`lapply`) – retorna uma lista
- **s** (`sapply`) – retorna um vetor simples ou matriz

O segundo parâmetro é a função a ser invocada. Parâmetros adicionais são repassados para essa função.

### `apply` – a função genérica

A função `apply` é mais flexível e trabalha com matrizes. Seu segundo parâmetro define a direção do processamento:

- **1** – aplica a função para cada linha
- **2** – aplica a função para cada coluna

Ideal quando você precisa controlar exatamente como os dados serão processados.

# Criando uma Tabela Básica

Primeiro, vamos criar um data frame simples com dados de peso, altura e identificação de sujeitos para demonstrar as funções apply.

```
weight <- c(60, 72, 57, 90, 95, 72)
height <- c(1.75, 1.80, 1.65, 1.90, 1.74, 1.91)
subject <- c("A", "B", "C", "D", "E", "F")

d <- data.frame(weight=weight, height=height, subject=subject)
head(d)
```



## Resultado do data frame

```
## weight height subject
## 1   60  1.75    A
## 2   72  1.80    B
## 3   57  1.65    C
## 4   90  1.90    D
## 5   95  1.74    E
## 6   72  1.91    F
```

Agora temos uma estrutura de dados organizada pronta para aplicar as funções de iteração implícita.

## Usando lapply e sapply

Ambas as funções processam cada coluna, invocando a função passada como segundo parâmetro. Os parâmetros seguintes são repassados para a função.

### lapply – retorna lista

```
lapply(d[,1:2], mean, na.rm=TRUE)
```

#### Saída:

```
## $weight  
## [1] 74.33333  
##  
## $height  
## [1] 1.791667
```

Retorna uma lista nomeada com os resultados para cada coluna processada.


### sapply – retorna vetor

```
sapply(d[,1:2], mean, na.rm=TRUE)
```

#### Saída:

```
## weight height  
## 74.333333 1.791667
```

Simplifica o resultado em um vetor nomeado, mais conveniente para cálculos posteriores.

 **Dica:** Use lapply quando precisar de estruturas complexas e sapply quando quiser resultados simplificados e diretos.

## apply com Matrizes

A função apply oferece controle total sobre a direção do processamento. O segundo parâmetro define se aplicamos a função nas linhas (1) ou colunas (2).

```
m <- as.matrix(d[,1:2])
```

### Processando Linhas (1)

```
apply(m, 1, min, na.rm=TRUE)
```

#### Saída:

```
## [1] 1.75 1.80 1.65 1.90 1.74 1.91
```

Encontra o valor mínimo em cada linha da matriz.

### Processando Colunas (2)

```
apply(m, 2, min, na.rm=TRUE)
```

#### Saída:

```
## weight height  
## 57.00 1.65
```

Encontra o valor mínimo em cada coluna da matriz.

O primeiro parâmetro de apply é sempre uma matriz. Essa função é essencial para operações matriciais eficientes.

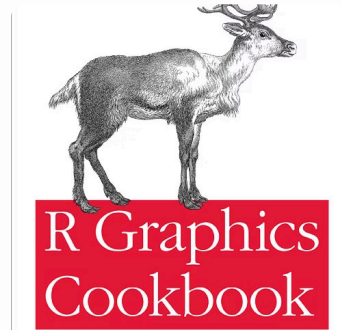
# Referências



## Hands-on Programming

Aprenda R criando suas próprias funções e simulações

<https://rstudio-education.github.io/hopr/index.html>



## R Graphics Cookbook

Domine visualizações de dados em R

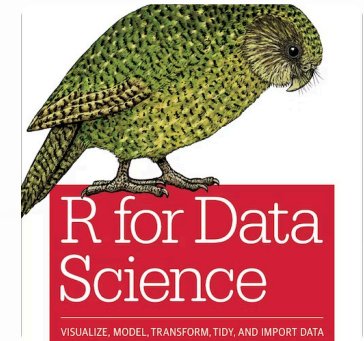
<https://r-graphics.org>



## R Packages

Desenvolva seus próprios pacotes R

<https://r-pkgs.org/index.html>



## R for Data Science

Guia completo para ciência de dados

<https://r4ds.had.co.nz>