

ESTRUTURAS DE DADOS EM R

Listas em R

Explore um dos tipos de dados mais versáteis e poderosos da linguagem R: as listas. Estruturas flexíveis que permitem armazenar e organizar diferentes tipos de elementos em um único objeto.



Eduardo Ogasawara

eduardo.ogasawara@cefet-rj.br

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

O que são Listas?



Estrutura Versátil

Listas são objetos R que contêm elementos de diferentes tipos: números, strings, vetores, matrizes, data frames e até outras listas.



Elementos Diversos

Uma lista pode conter matrizes ou funções como seus elementos, oferecendo máxima flexibilidade na organização de dados.



Criação e Manipulação

Criadas usando a função `list()`, as listas podem ser manipuladas através de fatiamento `[]` e acesso direto `[[]]`.



Criando Listas

Estrutura Básica

As listas podem conter qualquer tipo de elemento. Use a função `list()` para criar uma nova lista com diferentes tipos de dados.

Para adicionar um novo item, simplesmente use uma posição vazia. As listas podem até conter outras listas aninhadas!

```
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")
```

```
mybag <- list(weight, height, subject, 0,  
"a")  
mybag
```



Resultado: A lista `mybag` contém 5 elementos de tipos diferentes: dois vetores numéricos (`weight` e `height`), um vetor de caracteres (`subject`), um número (0) e uma string ("a").

Corte da Lista

Colchetes Simples []

Ao acessar listas com colchetes simples, você obtém uma **sublista** que mantém a estrutura de lista original.

```
slice <- mybag[c(1,3)]  
is.list(slice) # TRUE  
is.vector(slice) # TRUE
```

Colchetes Duplos [[]]

Com colchetes duplos, você extrai o **elemento em si**, não mais como lista, mas como seu tipo original (vetor, número, etc).

```
h <- mybag[[1]]  
is.vector(h) # TRUE  
is.list(h) # FALSE
```

A diferença entre [] e [[]] é fundamental: o primeiro mantém a estrutura de lista, enquanto o segundo extrai o conteúdo.

Membros da Lista

Acesso Direto aos Elementos

Para acessar os elementos individuais de uma lista e trabalhar com eles diretamente, utilize colchetes duplos `[]`.

O elemento extraído assume seu tipo original, permitindo operações específicas para aquele tipo de dado.

```
h <- mybag[[1]]  
h  
## [1] 60 72 57 90 95 72
```

```
is.vector(h) ## TRUE  
is.list(h)   ## FALSE
```



Dica: Use `[]` quando precisar trabalhar diretamente com o conteúdo, não com a estrutura da lista.

Criando Lista com Atributos

01

Nomeação de Elementos

Elementos de listas podem ser nomeados para facilitar identificação e acesso. Isso funciona como um atributo do objeto.

03

Organização Clara

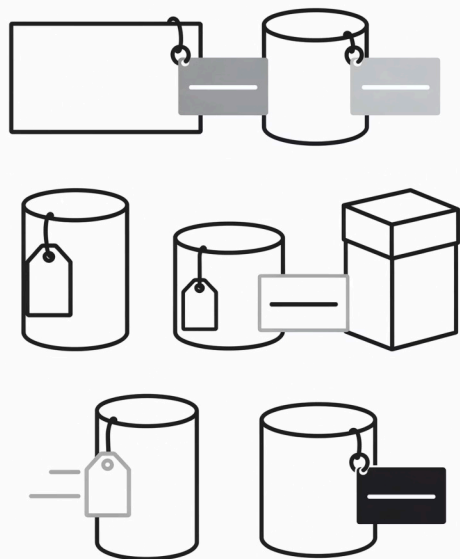
Listas nomeadas facilitam a manutenção do código e reduzem erros ao trabalhar com estruturas complexas.

```
mybag <- list(weight=weight, height=height,  
  subject=subject, valor=0, nome="a")  
mybag$weight # Acessa o vetor weight
```

02

Operador \$

O operador `$` é usado para acessar diretamente um atributo nomeado da lista, tornando o código mais legível.

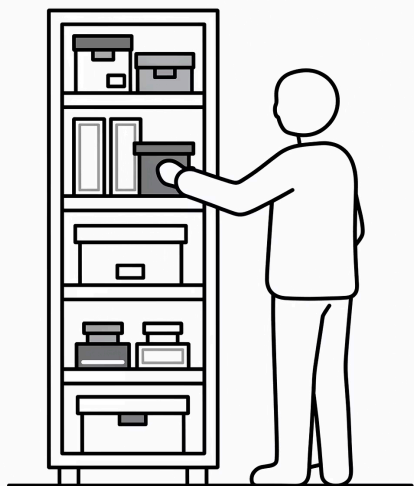


Removendo Atributos

Manipulação Dinâmica

Você pode adicionar novos atributos ou remover existentes facilmente usando o operador `$` ou atribuindo `NULL`.

Essa flexibilidade torna as listas ideais para estruturas de dados que evoluem durante a execução do programa.



```
mybag$bmi <- mybag$weight / mybag$height^2  
mybag[[4]] <- NULL  
mybag$nome <- NULL
```

```
mybag  
## $weight  
## [1] 60 72 57 90 95 72  
## $height  
## [1] 1.75 1.80 1.65 1.90 1.74 1.91  
## $subject  
## [1] "A" "B" "C" "D" "E" "F"  
## $bmi  
## [1] 19.59 22.22 20.94 24.93 31.38 19.74
```



Resultado: A lista agora contém um novo atributo `bmi` calculado, e os atributos `valor` e `nome` foram removidos.

Indexação: Guia Rápido

O R oferece três operadores distintos para acessar elementos de estruturas de dados, cada um com comportamento específico e propósito único.

Ideia Central

Os colchetes simples `[]` preservam a estrutura original, enquanto `[[]]` extrai o conteúdo diretamente.



`[]` Subconjunto

Retorna múltiplos elementos mantendo a estrutura do objeto original



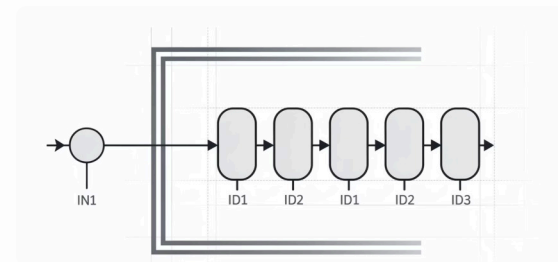
`[[]]` Elemento Único

Extrai um único elemento, removendo a camada externa da estrutura



`$` Acesso por Nome

Forma simplificada de acessar elementos nomeados em listas e data frames



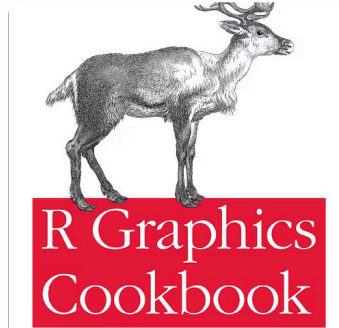
Referências



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

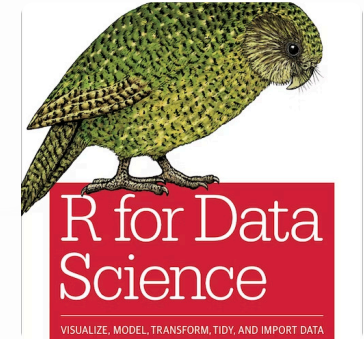
<https://r-graphics.org>



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>