

Criando objetos em R

36

Objetos em R

Estruturas básicas de objetos



Vetores (vector)

Listas (list)

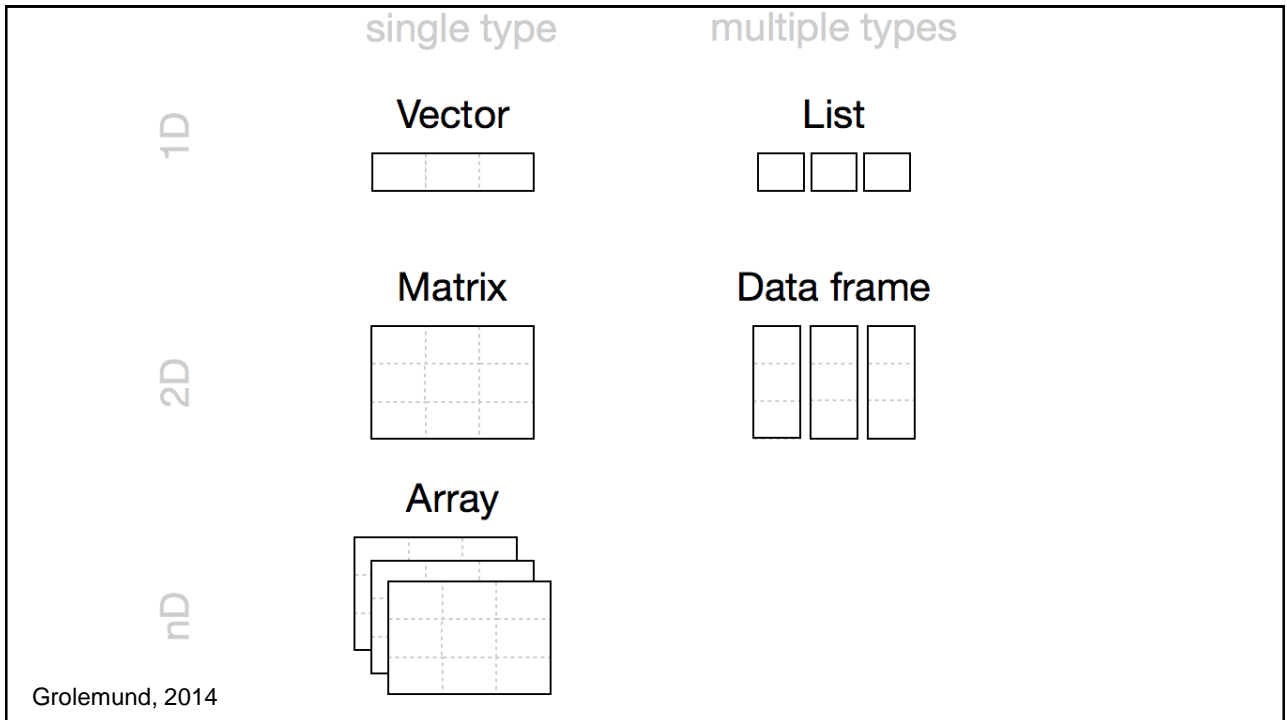
Matrizes (matrix)

Tabelas de dados (data.frame)

Arranjo (array)

Alcoforado, 2021; Wickham, 2014

37



38

Vetores em R

Vetores

Um vetor armazena em um único local **um valor ou uma sequência ordenada de valores**

Utiliza o sinal de atribuição (<- ou =) e a função **c()** para combinar os valores

```
> idade1 <- 23
```

```
idade1
```

```
[1] [23]
```

```
> idade2 <- 42
```

```
idade2
```

```
[1] [42]
```

```
> idade3 <- 55
```

```
idade3
```

```
[1] [55]
```

```
> idades <- c(23,42,55)
```

```
idades
```

```
[1] 23 42 55
```

```
> idades <- c(idade1, idade2,
idade3)
```

```
idades
```

```
[1] 23 42 55
```

Vetores

```
idade1  idade2  idade3
```

```
idades  23    42    55
```

```
idades  idade1 idade2 idade3
```

DICA DE OURO!

Os dados de um vetor devem ser do mesmo TIPO, caso contrário, haverá “coerção” para um mesmo tipo (mais flexível). Ex: todos números ou todos texto

Alcoforado, 2021

39

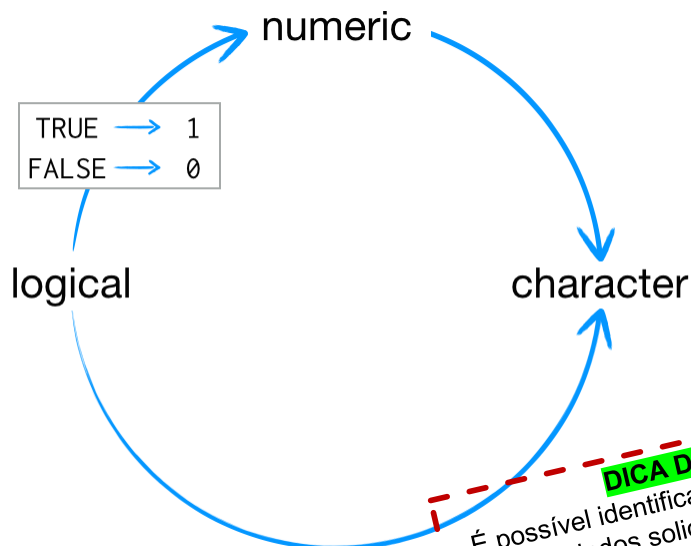
Dados em R

Tipos básicos de dados



Alcoforado, 2021; <http://adv-r.had.co.nz/Data-structures.html>

40



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

DICA DE OURO!
É possível identificar a estrutura, classe e tipo dos dados solicitando as funções `str()`, `class()` e `typeof()`

41

Vetores em R

Criando vetores

```
> x <- c("Pedro", "Ana", "Roberto")
> x
[1] "Pedro" "Ana"   "Roberto"
```

```
> x <- c(25,32,54)
> x
[1] 25 32 54
```

```
> x <- c(25, "32",54)
> x
[1] "25" "32" "54"
```

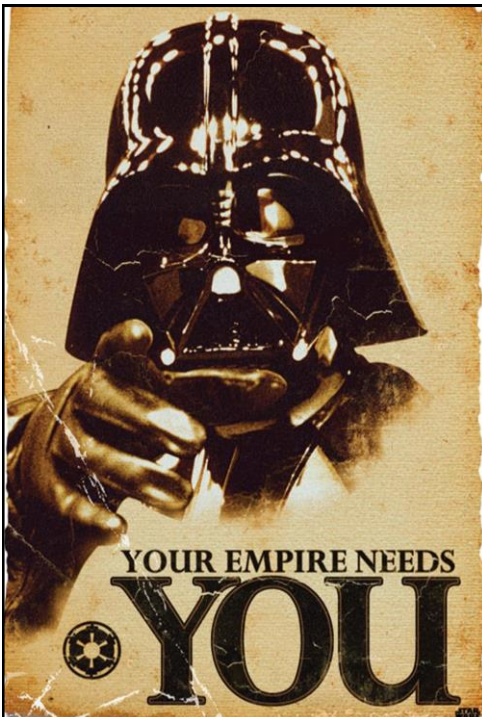
```
> x <- c(24, 2, FALSE, TRUE)
> x
[1] 24 2 0 1
```

```
> x <- c(24, 2, FALSE, TRUE, "hello")
> x
[1] "24"  "2"   "FALSE" "TRUE" "hello"
> x <- c(25, "32",54)
> x
[1] "25" "32" "54"
```

DICA DE OURO!

O nome não pode começar com um número e não pode usar alguns símbolos especiais: ^, !, \$, @, +, -, / ou *

42



Agora é com **você!**

Listas de exercícios no Google
Colab:

[Exercícios 2.1](#)

43

Vetores em R

Operações básicas com Vetores

Operação	Descrição
<code>x:y</code>	Gera uma sequência de x a y com aumento de uma unidade
<code>seq(from= , to = , by=)</code>	Gera uma sequência de valores com aumento definido
<code>x[n]</code>	Acessa o elemento de um vetor x na posição n
<code>["condição"]</code>	Acessa os valores que obedecem à condição
<code>which("condição")</code>	Acessa os índices (posições) que obedecem à condição
<code>x[n] <- a</code>	Atualiza a posição n com o valor a
<code>x[n] <- NA</code>	Define um valor ausente para a posição n
<code>x[-n]</code>	Exclui o valor da posição n

Alcoforado, 2021

44

Vetores em R

Exemplos

```
> 1:5
```

```
[1] 1 2 3 4 5
```

```
> seq(from= 5, to = 20, by= 3)
```

```
[1] 5 8 11 14 17 20
```

```
> x <- c("Pedro", "Ana", "Roberto")
```

```
> x[2]
```

```
[1] "Ana"
```

```
> which(x=="Ana")
```

```
[1] 2
```

```
> x[2] <- NA
```

```
> x
```

```
[1] "Pedro" NA "Roberto"
```

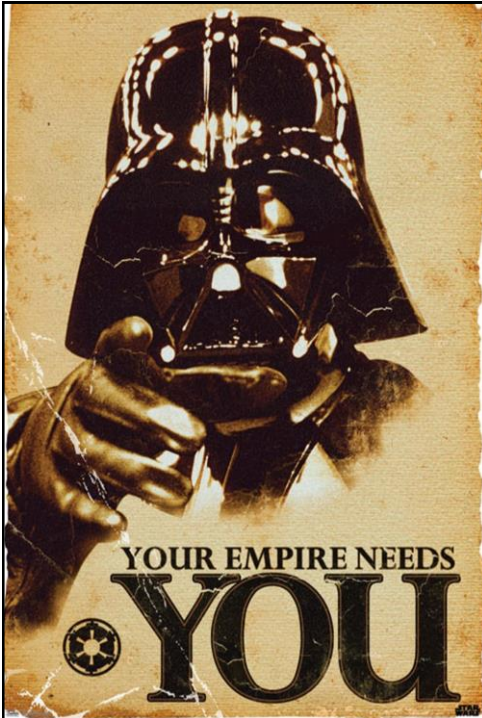
```
> x[-2]
```

```
[1] "Pedro" "Roberto"
```

Curiosidade: "<-" ou "="?

Ambos possuem o mesmo escopo. Porém, o operador "=" pode ser utilizado tanto para atribuição como para definição de parâmetros (em chamadas de função)

45



Agora é com **você!**

Listas de exercícios no Google
Colab:

[Exercícios 2.2](#)

46

Vetores em R

Aplicando funções a Vetores

Operação	Descrição
<code>length()</code>	Determina o tamanho do vetor
<code>min()</code> , <code>max()</code>	Determina o valor mínimo e máximo do vetor, respectivamente
<code>range()</code>	Determina a amplitude do vetor, respectivamente
<code>round()</code>	Arredonda o valor de vetor
<code>sort()</code>	Ordena os elementos de um vetor
<code>sum()</code>	Soma os elementos do vetor
<code>prod()</code>	Multiplica os elementos do vetor
<code>mean()</code> , <code>median()</code>	Calcula a média e mediana dos elementos do vetor, respectivamente
<code>var()</code> , <code>sd()</code> , <code>IQR()</code>	Calcula a variância, desvio padrão e intervalo interquartilico do vetor
<code>summary()</code>	Calcula medidas resumo do vetor

Alcoforado, 2021

47

Vetores em R

Exemplos

```
> length(x)
```

```
[1] 3
```

```
> idades <- c(10,86,30, 25)
```

```
> min(idades)
```

```
[1] 10
```

```
> sort(idades)
```

```
[1] 10 25 30 41
```

```
> mean(idades)
```

```
[1] 37.75
```

```
> round(mean(idades),1)
```

```
[1] 37.8
```

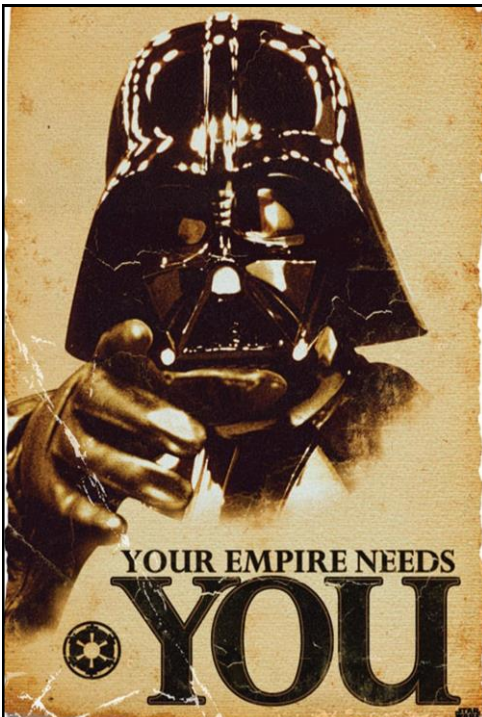
```
> sd(idades)
```

```
[1] 33.27036
```

```
> summary(idades)
```

```
Min. 1st Qu. Median Mean 3rd Qu. Max.
10.00 21.25 27.50 37.75 44.00 86.00
```

48



Agora é com **você!**

Listas de exercícios no Google
Colab:

[Exercícios 2.3](#)

49

Vetores em R

Criando suas próprias funções

Sintaxe

```
nome <- function (argumento1 , ... , argumento n) {  
  Comandos da função  
}
```

Elementos

nome: trata-se do nome pelo qual você chamará a função posteriormente

function (): comando para a definição de sua função

argumento: argumentos de entrada de dados que serão necessários para a função

{ }: todos os comandos e fórmulas que definem a função ficam delimitados entre as chaves

return(): comando não obrigatório, mas essencial quando se deseja recuperar algum resultado da função

Alcoforado, 2021; Figueira, 2016

50

Operações matemáticas em R

Exemplos

Criando a função “maior” que, dados dois valores (a e b), retorna o maior entre eles

Criando a função

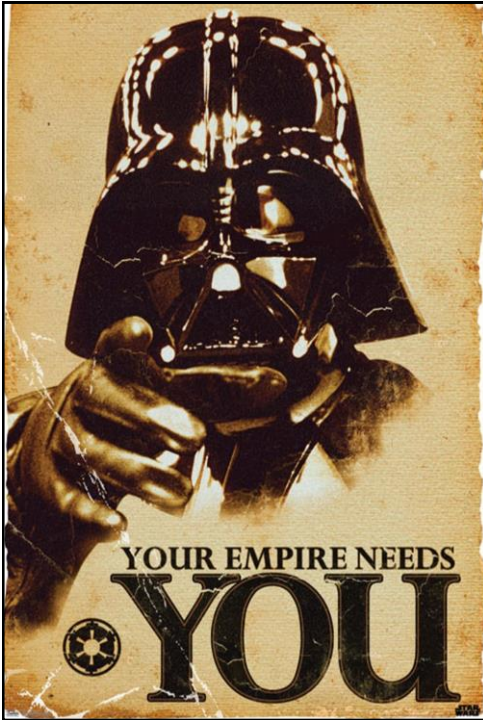
```
> maior <- function (a,b) {  
  
  if (a < b) {  
    return (b)  
  } else {  
    return (a)  
  }  
}
```

Chamando a função

```
> maior(5,8)  
[1] 8  
  
> x <- 5  
> y <- 8  
> maior(x,y)  
[1] 8
```

Alcoforado, 2021; Figueira, 2016

51



Agora é com **você!**

Listas de exercícios no Google
Colab:

[Exercícios 2.4](#)

52



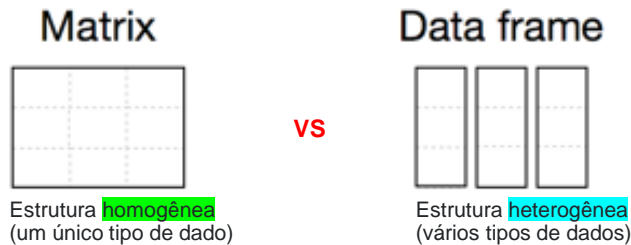
Criando Tabelas de dados
(*data frames*)

53

Data frames em R

Data Frame

Semelhante a uma matriz, trata-se de um arranjo retangular de valores, ou seja, não é mais um “vetor unidimensional”. Suas dimensões percorrem duas direções: (número de linhas) x (número de colunas)



Alcoforado, 2021; Grolemund, 2014

54

Dataframes em R

Os data frames também são **listas**, mas têm algumas restrições:

- você não pode usar o mesmo nome para duas variáveis diferentes
- todos os elementos (colunas) são vetores
- todos os elementos (colunas) têm o mesmo comprimento (linhas).

data frame

1	"R"	TRUE
2	"S"	FALSE
3	"T"	TRUE
numeric	character	logical

DICA DE OURO!
A estrutura dos data frames é semelhante ao de **matrizes**, permitindo operações em linhas, o que não é possível com listas (linhas indefinidas)

55

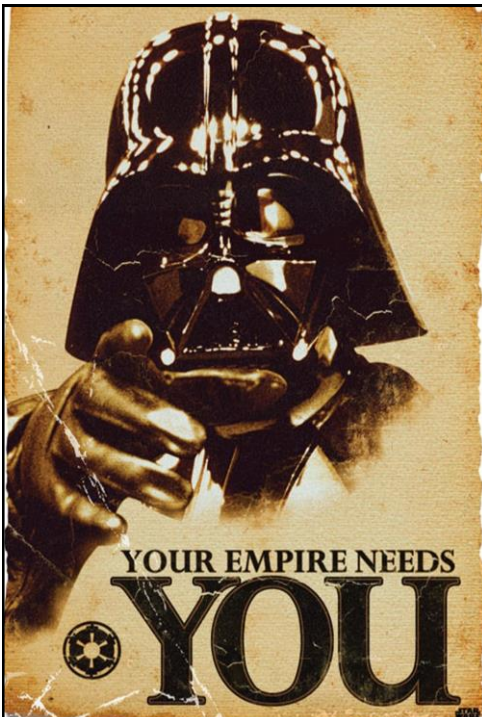
Data frames em R

Criando um data frame

```
> paciente_id <- c(1:3)
> teste_resultado = c("Negativo", "Positivo", "Positivo")
> teste_valor <- c(70, 80, 220)
> teste_data <- c("2000-05-10", "2004-04-10", "2005-03-10")
> dados <- data.frame(paciente_id,
                      teste_resultado,
                      teste_valor,
                      teste_data)
```

```
> print(dados)
  paciente_id teste_resultado teste_valor teste_data
1           1      Negativo          70 2000-05-10
2           2       Positivo          80 2004-04-10
3           3       Positivo         220 2005-03-10
```

56



Agora é com **você!**

Listas de exercícios no Google
Colab:

Exercícios 2.5

57

Referências

Alcoforado LF. Utilizando A Linguagem R: Conceitos, manipulação, visualização, modelagem e elaboração de relatórios. Rio de Janeiro: Alta Book, 2021. 367p

Figueira L. Criando funções no R. Estatística é com R! Disponível em: <http://www.estadisticacomr.uff.br/?p=224>

Grolemund G. Hands-On Programming with R. O'Reilly Media; 1ª edição, 2014. Disponível em: <https://rstudio-education.github.io/hopr>

Wickham H. Advanced R. CRC Press; 1ª edição, 2014. Disponível em: <http://adv-r.had.co.nz/>

58

58



Universidade de Brasília
Faculdade de Ciências da Saúde
Departamento de Saúde Coletiva

Obrigado

Ivan Zimmermann
ivan.zimmermann@unb.br



59

59