

Pré-requisitos para o curso de R

Daniel C. Mota e Kleber G. Abitante

- a. O R possui os seguintes tipos de dados:
 - **numeric**: números que contêm decimal ou números inteiros. Exemplo: 1.1 (O R trabalha com o ponto (".") como separador de decimal);
 - **logical**: TRUE (T) ou FALSE (F);
 - **character**: são letras ou palavras. Deve estar entre aspas duplas ou simples. Por exemplo: "pib", 'ipca'; e
 - **factor**: é um tipo especial de caracter usado para variáveis categóricas, ou seja, variáveis que tem um conjunto conhecido e fixo de valores possíveis. Por exemplo, os meses do ano.

Criando uma variável

- O R usa o operador de atribuição (`<-`) para atribuir um valor a uma variável. Exemplos:
`x <- 10` (foi criada a variável `x` na memória, a qual está recebendo o valor 10)
`y <- "pib"`

Identificando o tipo de dado

- A função `class()` pode ser usada para identificar o tipo de dado de uma variável.

Exemplo: `class(x)`

Estruturas de dados - definição

As estruturas de dados são as formas em que é possível armazenar dados. As principais estruturas do R são `vector`, `matrix`, `array`, `list` e `data.frame`.

- 1. vector: é a estrutura mais simples de dados do R e pode armazenar apenas um único tipo de dado.

Exemplo: `c(1),c("pib","ipca")`

- `matrix`: é uma estrutura que possui duas dimensões (linhas e colunas) e pode armazenar apenas dados de uma classe.

Exemplo:

```
minha_mat <- matrix(1:16, nrow = 4, byrow = TRUE)
mat_1 <- matrix(c(2, 0, 1, 1), nrow = 2)
```

- O argumento `byrow = TRUE` indica que a `matrix` deve ser populada por linha (o *default* é por coluna); e
- Os argumentos `nrow` e `ncol` indicam os números de linhas e colunas que a matriz deve ter, respectivamente.

- array é similar à matriz, pois deve possuir apenas um tipo de dado, mas ele pode possuir mais de duas dimensões.

Exemplo:

```
ex_array <- array(1:24, dim = c(2, 4, 3))
```


- d. `list`: é a estrutura de dados mais versátil do R, pois pode armazenar diferentes tipos de dados e também armazenar outras estruturas de dados.

Exemplo:

```
list_1 <- list(c("black", "yellow", "orange"),  
c(TRUE, TRUE, FALSE, TRUE, FALSE, FALSE),  
matrix(1:6, nrow = 3))
```

- `data.frame`: é estrutura de dados mais utilizada no R. Possui duas dimensões (linhas e colunas) e pode armazenar diferentes tipos de dados em cada coluna. Exemplo de `data.frame`:

```
valor_altura <- c(180, 155, 160, 167, 181)
valor_idade <- c(20, 50, 35, 48, 70)
nomes_alunos <- c("João", "Maria", "Edson", "Carlos", "Ana")
dataf <- data.frame(aluno = nomes_alunos, altura = valor_altura, idade =
valor_idade)
```

Visualizar a estrutura da variável

- Para variáveis do tipo vector, usar a função `str()`;
- para `data.frame`, `matrix` ou `list`, usar a função `str()` ou clicar no símbolo de “play” do lado do nome da variável no painel Ambiente.

Operadores matemáticos básicos

- $+$: adição
- $*$: multiplicação
- $/$: divisão
- $-$: subtração
- $^$: potenciação

Selecionar os elementos em um array

- 1 Selecionar a pessoa “PF” (que está na segunda dimensão do array) e o tipo de operação “Desconto de cheques” (que está na terceira dimensão do array) (ex_array é o nome do array):
`ex_array[, "PF", "Desconto de cheques",]`

Selecionar os elementos em um array

- ② Selecionar uma data (que está na primeira dimensão) e o valor “saldo” (que está na quarta dimensão):
`ex_array["2020-01-01", , , "saldo"]`

Selecionar elementos em vetor

Considere um vetor chamado `ex_vector`:

- `ex_vector[1]` (selecionar o primeiro valor do vetor)
- `exVector[c(1,2)]` (selecionar o primeiro e o segundo valores do vetor)
- `exVector[c(1:3)]` (selecionar os primeiros 3 valores do vetor (valores de 1 a 3))

Selecionar colunas em data.frame

- O `data.frame iris` já vem carregado no R;
- Selecionar coluna em um `data.frame` pelo nome:
 - a. apenas uma coluna (a coluna chamada “Petal.Length”):
 - `iris[, "Petal.Length"]`
 - `iris$Petal.Length`
 - b. mais de uma coluna (as colunas com os nomes “Sepal.Length” e “Sepal.Width”):
 - `iris[, c("Sepal.Length", "Sepal.Width")]`

Selecionar colunas em data.frame

- Selecionar coluna em um data.frame pelo número da coluna:
 - a. apenas uma coluna:
 - `iris[,1]` (retorna apenas a primeira coluna)
 - `iris[,5]` (retorna apenas a quinta coluna)
 - b. mais de uma coluna:
 - `iris[,c(1,5)]` (retorna a primeira e a quinta colunas)

Selecionar linhas em data.frame

- Selecionar linha em um data.frame pelo nome:
 - a. apenas uma linha que possui o nome “data2021”:
 - `iris["data2021",]`
 - b. mais de uma linha:
 - `iris[c("data2021", "data2022"),]` (retorna as linhas com os nomes “data2021” e “data2022”)

Selecionar linhas em data.frame

- Selecionar linha em um data.frame pelo número:
 - a. apenas uma linha:
 - `iris[1,]` (retorna apenas a primeira linha)
 - `iris[10,]` (retorna apenas a décima linha)
 - b. mais de uma linha:
 - `iris[c(1,10),]` (retorna a primeira e a décima linha)

Selecionar linhas em data.frame

- Selecionar linha com base em um critério
 - `iris[iris$Species=="setosa",]` (retorna todas as linhas cuja coluna "Species" é igual à "setosa")
 - `iris[iris$Sepal.Length>5,]` (retorna todas as linhas cuja coluna "Sepal.Length" é maior que 5)
 - `iris[iris$Sepal.Length>5&iris$Sepal.Length<6,]` (retorna todas as linhas cuja coluna "Sepal.Length" é maior que 5 e menor que 6)
 - `iris[iris$Species=="setosa"|iris$Species=="versicolor",]` (retorna todas as linhas cuja coluna "Species" é igual à "setosa" ou "versicolor")
 - `iris[iris$Species%in%c("setosa","versicolor"),]` (mesmo resultado anterior, mas de forma mais compacta)

Selecionar linhas e colunas

- Quando apenas uma linha ou coluna é selecionada, por padrão o R converte essa linha/coluna em um vector unidimensional:

```
class(iris[, "Petal.Length"])
```

- Se for necessário evitar isso, utilize a opção `drop=F`:

```
iris[, "Petal.Length", drop=F]
```

```
class(iris[, "Petal.Length", drop=F])
```

Selecionar linhas e colunas em `matrix`

- A única diferença em relação à seleção em `data.frame` é que a matriz não permite o uso do símbolo `$` para selecionar colunas. As demais formas de seleção são iguais.