
Introdução ao Software R

Primeiros Passos

www.de.ufpb.br

www.youtube.com/channel/UC8QTeEyzHqYRjojKneTgLBa



UFPB



**Departamento de
ESTATÍSTICA**

O RStudio como calculadora



- ▶ A forma de uso mais básica do RStudio é usá-lo como calculadora. Os operadores matemáticos básicos são: + para soma, - subtração, * multiplicação, / divisão e ^ exponenciação. Digite as seguintes operações na linha de comandos do RStudio:

2 + 2

2 * 2

2 - 2

2 / 2

2^2

- ▶ Use parênteses para separar partes dos cálculos, por exemplo, para fazer a conta $4 + 16$, dividido por 4, elevado ao quadrado:

$((4+16)/4)^2$

- ▶ Um objeto pode ser criado com a operação de “atribuição”, o qual se denota como uma flecha, com o sinal de menos e o símbolo > ou <, dependendo da direção em que se atribui o objeto. Ou com um único sinal de igual. É importante dizer que o nome de um objeto deve começar com uma letra qualquer, maiúscula ou minúscula, que pode ser seguida de outras letras, números, ou caracteres especiais como o ponto.

```
x <- 10 # x receberá o valor 10  
15 -> y # y receberá o valor 15  
X <- 6 # X receberá o valor 6  
Y = 15 # Y receberá o valor 15
```

- ▶ Observe que existe diferença entre maiúscula e minúscula. Observe que o símbolo # indica para o RStudio um comentário.

- ▶ Pode-se fazer cálculos com esse objetos criados. Você também pode armazenar o resultado de um cálculo em um objeto qualquer.

```
t <- sqrt(4) #objeto t irá receber o valor  
             #da operação indicada
```

- ▶ Para mostrar o conteúdo do objeto criado `t`, digite apenas o nome do objeto na linha de comando do RStudio, como abaixo:

```
t  
[1] 2
```

- ▶ Vamos criar um conjunto de dados de contém o número de espécies de aves (riquezas) coletadas em 10 locais. As riquezas são 22, 28, 37, 34, 13, 24, 39, 5, 33, 32.

```
aves<-c(22,28,37,34,13,24,39,5,33,32)
```

- ▶ A letra minúscula `c` significa concatenar (colocar junto). Entenda como “agrupe os dados entre parênteses dentro do objeto” que será criado neste caso no objeto `aves`. A função `length` fornece o número de observações (`n`) dentro do objeto.

```
length(aves)
```

- ▶ Também podemos criar objetos que contêm letras ou palavras ao invés de números. Porém, as letras ou palavras devem vir entre aspas.

```
letras <- c("a","b","c","da","ew")  
palavras <- c("Manaus","Boa Vista",  
              "Belém","Brasília")
```

- ▶ Crie um objeto “misto”, com letras e com números. Funciona? Esses números realmente são números? Note a presença de aspas, isso indica que os números foram convertidos em caracteres. Evite criar vetores “misto”, a menos que tenha certeza do que está fazendo.

- ▶ Podemos fazer diversas operações usando o objeto `aves`, criado anteriormente.

```
max(aves) # valor máximo  
min(aves) # valor mínimo  
sum(aves) # Soma dos valores  
aves^2  
aves/10
```

- ▶ Agora vamos usar o que já sabemos para calcular a média dos dados das aves.

```
n.aves<-length(aves) # número de observações (n)  
media.aves<-sum(aves)/n.aves # média
```

Acessar valores dentro de um objeto



- ▶ Caso queira acessar apenas um valor do conjunto de dados use colchetes []. Isto é possível porque o RStudio salva os objetos como vetores, ou seja, a sequência na qual você incluiu os dados é preservada. Por exemplo, vamos acessar o quinto valor do objeto `aves`.

```
aves[5] # Qual o quinto valor de aves?
```

```
palavras[3] # Qual a terceira palavra?
```

- ▶ Para acessar mais de um valor use `c` para concatenar dentro dos colchetes:

```
aves[c(5,8,10)]
```

```
# acessa o quinto, oitavo e décimo valores de aves
```


Acessar valores dentro de um objeto



- ▶ Para excluir um valor, por exemplo, o primeiro, use

```
# note que o valor 22 foi excluído  
aves[-1]
```

- ▶ Caso tenha digitado um valor errado e queira corrigir o valor, especifique a posição do valor e o novo valor. Por exemplo, o primeiro valor de aves é 22, caso estivesse errado, por exemplo, deveria ser 100, basta alterarmos o valor da seguinte maneira.

```
# O primeiro valor de aves deve ser 100  
aves[1]<-100  
aves  
# Vamos voltar ao valor antigo  
aves[1]<-22
```

- ▶ Em alguns casos é necessário, ou recomendado, que você transforme seus dados antes de fazer suas análises. Transformações comumente utilizadas são log e raiz quadrada.

```
# Raiz quadrada dos valores de aves  
sqrt(aves)  
# logaritmo na base 10 de aves  
log10(aves)  
# logaritmo natural de aves  
log(aves)
```

- ▶ Para salvar os dados transformados dê um nome ao resultado. Por exemplo:

```
aves.log <- log10(aves)
```

Listar e remover objetos salvos



- ▶ Para listar os objetos que já foram salvos use `ls()` que significa listar. Para remover objetos use `rm()` para remover o que está entre parênteses.

```
rm(aves.log)
```

```
aves.log # você verá a mensagem:
```

```
Error in eval(expr, envir, enclos): objeto 'aves.log' não encontrado
```