

Comandos Básicos de R

Disciplina de Ciência de Dados 2025.2 - UESC

Gabriel Rodrigues

2025-08-26

Conceitos básicos no R

Operações matemáticas simples

```
# Soma  
soma = 4+4  
multiplicacao = 4*3  
divisao = 15/5  
potenciacao = 2^5  
  
# printar os resultados  
soma
```

```
[1] 8
```

```
multiplicacao
```

```
[1] 12
```

```
divisao
```

```
[1] 3
```

```
potenciacao
```

```
[1] 32
```

Código R - Vetores

Um vetor é a estrutura de dados mais básica do R. Pense nele como uma sequência de elementos que são todos do mesmo tipo. Um vetor só pode conter números, ou só caracteres, ou só lógicos (TRUE/FALSE), mas nunca uma mistura deles.

```
# Vetor numérico
vetor_num <- c(1,2,3,4,5)
# print
vetor_num
```

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

```
# Vetor de caracteres
vetor_char <- c("banana","morango","caju","laranja")
# print
vetor_char
```

```
[1] "banana" "morango" "caju"    "laranja"
```

Também é possível realizar operações em vetores.

```
# Vetor numérico
sequencia <- c(1:100) # sequência de 1 até 100

# Média da sequencia
mean(sequencia)
```

```
[1] 50.5
```

```
# Moda
median(sequencia)
```

```
[1] 50.5
```

```
# soma
sum(sequencia)
```

```
[1] 5050
```

Código R - Matrizes

Uma matriz é uma coleção de elementos dispostos em uma grade bidimensional (linhas e colunas). A principal regra de uma matriz é que todos os elementos devem ser do mesmo tipo. Se você tentar combinar tipos diferentes (por exemplo, números e caracteres), o R forçará todos os elementos a se tornarem o tipo mais flexível, que geralmente é o de caracteres.

```
# criando matriz
m <- matrix(c(1, 2, 3, 4, 5, 6), nrow = 2, ncol = 3)
m
```

```
      [,1] [,2] [,3]
[1,]    1    3    5
[2,]    2    4    6
```

Código R - Data Frames

Um dataframe é a estrutura mais utilizada para armazenar dados tabulares no R, pois é a mais flexível. Ele se parece com uma tabela do Excel ou um banco de dados, com linhas e colunas. A grande vantagem do dataframe é que cada coluna pode ser de um tipo diferente de dado (uma coluna numérica, outra de caracteres, e assim por diante), o que o torna ideal para a maioria dos conjuntos de dados reais.

```
# criando dataframe (df)
df <- data.frame(
  Nome = c("João", "Maria", "Ana"),
  Idade = c(25, 30, 22),
  Casado = c(TRUE, FALSE, TRUE) # dado booleano
)

#print
df
```

```
      Nome Idade Casado
1  João    25    TRUE
2 Maria    30   FALSE
3   Ana    22    TRUE
```

Selecionando itens em um dataframe:

```
# dataframe
df <- data.frame(
  Nome = c("João", "Maria", "Ana"),
  Idade = c(25, 30, 22),
  Casado = c(TRUE, FALSE, TRUE) # dado booleano
)

# selecionar somente Maria e Ana
# maria e ana compõem as linhas 2 e 3 do dataframe
# o R mantém automaticamente os itens nas colunas prévias
# criando subset -> dataframe[x,y] -> dataframe[linha,coluna]

subset <- df[c(2,3),]
subset
```

```
      Nome Idade Casado
2 Maria     30  FALSE
3  Ana     22   TRUE
```

Importando tabela da aula

Vamos utilizar a tabela que preenchemos em aula para mostrar algumas operações básicas com dataframes no R. Esse é o arquivo da tabela gerada separado por tabulação (.tsv):

```
sexo    altura  idade  numero_pe
Masculino  175  25   41
Masculino  175  20   39
Masculino  188  24   43
Masculino  182  22   44
Masculino  183  23   43
Masculino  168  29   40
Masculino  182  28   42
Feminino   167  45   37
Feminino   170  22   39
Feminino   164  40   37
Masculino  175  23   41
Masculino  196  26   44
Masculino  175  27   40
Masculino  180  25   41
Masculino  170  20   42
```

Essa tabela foi preenchida de forma que os alunos preencheram seu gênero, altura, idade e número do calçado - respectivamente.

Vamos copiar essa tabela e salvar um arquivo chamado `levantamento_aula.tsv`. Usaremos ele para algumas operações.

Importando tabelas no R

Para isso podemos usar as funções `read.table()` e `read.csv()`.

```
# O arquivo 'levantamento_aula.tsv' está no mesmo diretório do código R
# Nesse caso vamos passar o caminho direto para o arquivo
# Importando tabela como DataFrame

# Método 01 - função read.table()
tabela <- read.table('levantamento_aula.tsv', sep='\t', header=TRUE)

# Método 02 - função read.csv()
tabela <- read.csv('levantamento_aula.tsv', sep='\t', header=TRUE)

# Mostrar início da tabela - função head()
head(tabela)
```

	sexo	altura	idade	numero_pe
1	Masculino	175	25	41
2	Masculino	175	20	39
3	Masculino	188	24	43
4	Masculino	182	22	44
5	Masculino	183	23	43
6	Masculino	168	29	40

Ambas as funções tem parâmetros chave que devem ser preenchidos:

- **sep**: Indica o caractere separador da tabela, nesse caso uma tabulação;
- **header**: A presença/ausência de nomes para as colunas do dataframe;

Análise dos dados criados em Aula

Vamos explorar algumas métricas básicas do nosso dataframe usando a função `summary()`.

```
# Importando tabela:
tabela <- read.csv('levantamento_aula.tsv', sep='\t', header=TRUE)

# Média de altura
summary(tabela)
```

sexo	altura	idade	numero_pe
Length:15	Min. :164.0	Min. :20.0	Min. :37.00
Class :character	1st Qu.:170.0	1st Qu.:22.5	1st Qu.:39.50
Mode :character	Median :175.0	Median :25.0	Median :41.00
	Mean :176.7	Mean :26.6	Mean :40.87
	3rd Qu.:182.0	3rd Qu.:27.5	3rd Qu.:42.50
	Max. :196.0	Max. :45.0	Max. :44.00

Nesse caso podemos ver diretamente as medidas de média e mediana (mean & median) nos dados numéricos da nossa tabela, além de informações sobre mínimos e máximos e quartis dos nossos dados.

Análise exploratória

Como vimos em aula, uma correlação pode ser vista a partir dos dados já obtidos - o que é o nosso caso. Podemos procurar por correlações na nossa tabela e tentar plotar algumas delas com o R. Vamos tentar com algumas coisas.

Proporção de Gênero

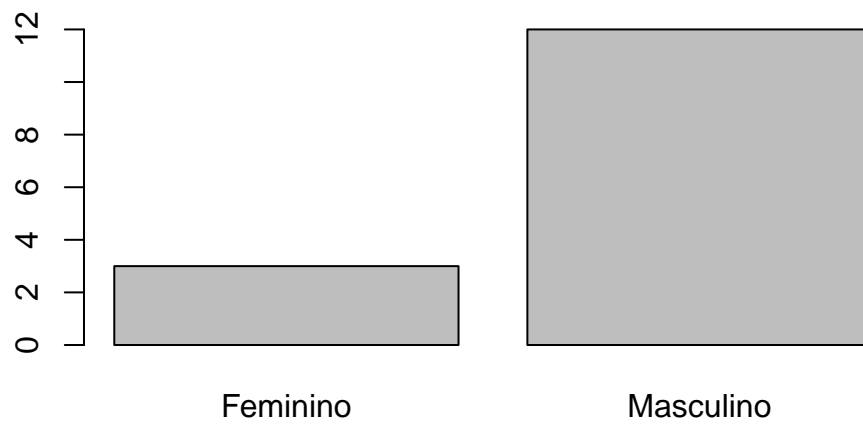
```
# Importando tabela:
tabela <- read.csv('levantamento_aula.tsv', sep='\t', header=TRUE)

# distribuição de gênero na sala
gen <- table(tabela$sexo) # função para contar fatores/classes de um vetor

gen #print
```

```
Feminino Masculino
      3      12
```

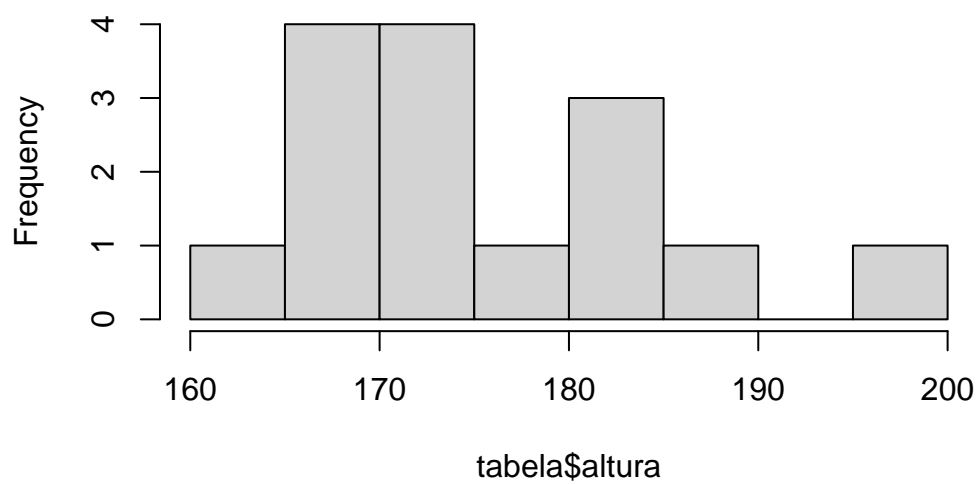
```
# gráfico de barras nativo do R
barplot(gen)
```



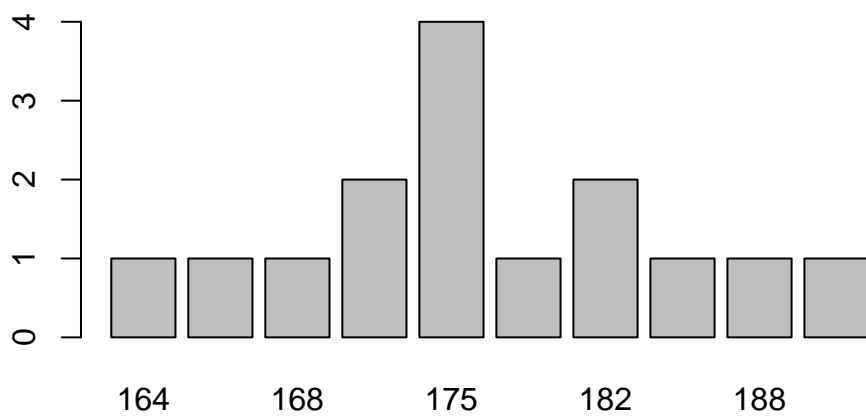
Altura dos alunos

```
# Importando tabela:  
tabela <- read.csv('levantamento_aula.tsv', sep='\t', header=TRUE)  
  
# histograma da altura dos alunos  
hist(tabela$altura)
```

Histogram of tabela\$altura



```
# gráfico de barras das alturas  
barplot(table(tabela$altura))
```



Atividade para a próxima aula

- Quais perguntas você faria antes de analisar essa tabela?
- Quais possíveis correlações podem ser feitas a partir desse dataframe?
- Liste os insights que podem ser obtidos com esses dados.