

Universidade Federal do Rio Grande do Norte  
Instituto Metr pole Digital  
**IMD0601 - Bioestat stica**

# Organiza  o dos dados em R

---

Prof. Dr. Tetsu Sakamoto  
Instituto Metr pole Digital - UFRN  
Sala A224, ramal 182  
Email: [tetsu@imd.ufrn.br](mailto:tetsu@imd.ufrn.br)



## Baixe a aula (e os arquivos)

- Para aqueles que não clonaram o repositório:

```
> git clone https://github.com/tetsufmbio/IMD0601.git
```

- Para aqueles que já tem o repositório local:

```
> cd /path/to/IMD0601
```

```
> git pull
```

# Revisão

Sinal de atribuição	<code>&lt;-</code>
Vetor numérico	<code>c(1,2,3,4)</code>
Vetor lógico	<code>c(TRUE, FALSE, NA)</code>
Vetor de caracteres	<code>c("My", "name", "is")</code>
Ajuda	<code>?c</code>
Operações aritméticas	<code>c(1.1, 9, 3.14)*2+100 # 102.20 118.00 106.28</code>
Reciclagem	<code>c(1,2,3,4) + c(0,10) # 1 12 3 14</code>

# Revisão

Sequência de números	<code>1:20; 15:1; seq(1,20, by=0.5); seq(5,10,length=30)</code>
Replicar números	<code>rep(0, times=40); rep(c(0,1,2), times=10)</code>
Tamanho do vetor	<code>length(rep(0, times=40))</code>
Juntar palavras	<code>paste(c("My", "code"), collapse = " ")</code>

# Valores (dados) faltantes

NA - not available

```
x <- c(44, NA, 5, NA)
```

```
3 * x
```

```
is.na(x) # mesmo que x == NA?
```

NaN - not a number

```
0 / 0
```

```
Inf - Inf
```

# Subconjunto de vetores

<code>x[1:10]</code>	Extrai os 10 primeiros elementos de x;
<code>x[c(3,5,7)]</code>	Extrai o 3º, 5º e o 7º elemento de x;
<code>x[c(-3,-5,-7)]</code>	Extrai os elementos de x menos o 3º, 5º e o 7º;
<code>x[-c(3,5,7)]</code>	Extrai os elementos de x menos o 3º, 5º e o 7º;
<code>x[c(-3,5,7)]</code>	Erro, não misture índices positivos e negativos
<code>x[x &gt; 0]</code>	Extrai elementos positivos de x;
<code>x[is.na(x)]</code>	Extrai elementos NA de x;
<code>x[!is.na(x)]</code>	Extrai os elementos não NA de x;
<code>x[0]</code>	Índice fora do intervalo (numeric(0))
<code>x[3000]</code>	Índice fora do intervalo (NA)

# Nomeando os elementos dos vetores

```
vector <- c(140, -50, 20, -120, 240)
days <- c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday")
names(vector) <- days
```

```
vector[2]           # -50
vector["Tuesday"]   # -50
vector[c(3,5)]       # (20,240)
vector[c("Wednesday", "Friday")] # (20,240)
```

```
vect <- c(foo = 11, bar = 2, norf = NA)
vect["foo"]          # 11
```

# Matrizes e tabelas em R

Organização dos dados em R

Carregue a biblioteca swirl no seu Rstudio → `library(swirl);`

Execute a função `swirl()` e faça o exercício 7 (*Matrices and Data Frames*) do curso *R programming*;

---



# Matrizes e Data Frame

- Coleção de elementos (numérico, caracteres, lógicos);
- Número determinado de linhas e colunas;

## Matriz

- Os elementos são do mesmo tipo.

## Data Frame

- Os elementos podem ser de tipos diferentes.

# Matrizes e Data Frame

Criando matrizes      `m <- matrix(1:20, 4, 5)`

`m <- c(1:20); dim(m) <- c(4,5)`

Criando Data Frame      `patients <- c("Bill","Gina","Kelly","Sean")`  
`data.frame(patients, m)`

Combinar colunas      `cbind(v1(m1), v2(m2))`

Combinar linhas      `rbind(v1(m1),v2(m2))`

Nomear colunas      `colnames(v)`

Nomear linhas      `rownames(v)`

# Matrizes e Data Frame

```
m <- matrix(1:20, 4, 5)
```

```
m
```

	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	1	5	9	13	17
[2,]	2	6	10	14	18
[3,]	3	7	11	15	19
[4,]	4	8	12	16	20

```
m[2,3] # elemento da 2ª linha
```

```
# e 3ª coluna
```

```
m[1,] # primeira linha
```

```
m[,1] # primeira coluna
```

# Matrizes e Data Frame

```
> m <- matrix(1:20, 4, 5, byrow = TRUE)
```

```
> m
```

	[,1]	[,2]	[,3]	[,4]	[,5]
[1,]	1	2	3	4	5
[2,]	6	7	8	9	10
[3,]	11	12	13	14	15
[4,]	16	17	18	19	20

# Matrizes e Data Frame

```
> m <- matrix(1:20, 4, 5)
> colnames(m) <- c("a", "b", "c", "d", "e")
> m
```

	a	b	c	d	e
[1,]	1	2	3	4	5
[2,]	6	7	8	9	10
[3,]	11	12	13	14	15
[4,]	16	17	18	19	20

```
# elemento da 2ª L e 3ª C
m[2,3]
m[2,"c"]

# primeira linha
m[1,]

# primeira coluna
m[,1]
m[, "a"]
```

# Explorando os dados em R

Organização dos dados em R

Carregue a biblioteca swirl no seu Rstudio → `library(swirl);`

Execute a função `swirl()` e faça o exercício 12 (*Looking at Data*) do curso *R programming*;

---

# Revisão

<code>ls()</code>	Listar as variáveis no ambiente;
<code>class(v)</code>	Verifica a classe de um objeto <code>v</code> ;
<code>dim(v)</code>	Mostra as dimensões de uma tabela <code>v</code> (linha x coluna);
<code>nrow(v)</code>	Mostra o número de linhas em <code>v</code> ;
<code>ncol(v)</code>	Mostra o número de colunas em <code>v</code> ;
<code>object.size(v)</code>	Mostra o espaço ocupado na memória por <code>v</code> ;
<code>head(v)</code>	Mostra as 6 primeiras linhas de <code>v</code> ;
<code>tail(v)</code>	Mostra as 6 últimas linhas de <code>v</code> ;
<code>summary(v)</code>	Mostra um breve sumário de <code>v</code> ;
<code>table(v)</code>	Conta o número de ocorrência de cada categoria em <code>v</code> ;
<code>str(v)</code>	Mostra a estrutura de <code>v</code> ;

# iris.data e iris.name

Source: <https://archive.ics.uci.edu/ml/datasets/iris>



Dados de comprimento e largura de pétalas e sépalas de três variedades de íris;

- iris.data → contém os dados;
- Iris.names → contém algumas descrições sobre os dados;

<b>Data Set Characteristics:</b>	Multivariate	<b>Number of Instances:</b>	150	<b>Area:</b>	Life
<b>Attribute Characteristics:</b>	Real	<b>Number of Attributes:</b>	4	<b>Date Donated</b>	1988-07-01
<b>Associated Tasks:</b>	Classification	<b>Missing Values?</b>	No	<b>Number of Web Hits:</b>	2411032



# iris.data e iris.name



## *Exercício:*

1. Veja a documentação da função `read.csv()` e carregue os dados do `iris.data`;
2. Veja a descrição de cada coluna dos dados em `iris.names` e nomeie as colunas;
3. Explore os dados utilizando as funções vistas nesta aula e responda:
  - a. Qual a dimensão da tabela?
  - b. Quais são os tipos de dados de cada coluna?
  - c. Quanto estes dados estão ocupando na memória?
  - d. Quantas amostras estão presentes em cada classe de Íris?
  - e. Tem algum dado faltante na tabela?