

ME115 - Linguagem R

Atividade Prática 04

1º semestre de 2023

Introdução

Nessa atividade, exploraremos:

1. Objetos do tipo `factor`;
2. Funções da família `apply`;
3. A função `replicate()`.

Exercícios inspirados em <http://r-tutorials.com/>.

Atividade

1. Crie uma matriz como indicado abaixo e a seguir use a função da família `apply` que seja mais apropriada.

```
my.matrix <- matrix(data = c(6, 34, 923, 5, 0, 112:116, 5, 9, 34, 76, 2, 545:549),  
                     nrow = 5)
```

```
my.matrix
```

```
##      [,1] [,2] [,3] [,4]  
## [1,]    6  112    5 545  
## [2,]   34  113    9 546  
## [3,]  923  114   34 547  
## [4,]    5  115   76 548  
## [5,]    0  116    2 549
```

- a. Obtenha a média das linhas de `my.matrix` tal como o resultado abaixo.

```
## [1] 167.00 175.50 404.50 186.00 166.75
```

- b. Obtenha a média das colunas de `my.matrix` tal como o resultado abaixo.

```
## [1] 193.6 114.0 25.2 547.0
```

- c. Ordene as colunas de `my.matrix` em ordem crescente tal como o resultado abaixo.

```
##      [,1] [,2] [,3] [,4]  
## [1,]    0  112    2 545  
## [2,]    5  113    5 546  
## [3,]    6  114    9 547  
## [4,]   34  115   34 548  
## [5,]  923  116   76 549
```

2. Considere o conjunto de dados `mtcars` disponível no R.

- a. Usando as funções `lapply()`, `sapply()` e `mapply()` da família `apply`, obtenha os valores mínimos de cada coluna do banco de dados `mtcars` e guarde os valores nas variáveis `l`, `s`, `m`, respectivamente.
- b. Armazene as três saídas `l`, `s`, `m` em uma lista chamada `list.objects`.

- c. Use uma função do tipo *apply* adequada para obter a classe de cada um dos três elementos da lista em `listobjects`.
3. A seguir, use `mapply()`.
 - a. Obtenha uma lista de 8 elementos que deve alternar entre “ME” and “115”. Os comprimentos desses 8 elementos alternados diminuem passo a passo de 8 para 1.
 - b. Ajuste a função do item (a) para obter os números de elemento adequados (1:8) para os 8 elementos da lista. Dica: argumento `USE.NAMES`.
4. Considere o conjunto de dados `Titanic` disponível no R.
 - a. Use uma função da família *apply* apropriada para obter a soma de homens e mulheres a bordo.
 - b. Faça uma tabela com a soma dos sobreviventes, segundo sexo.
 - c. Faça uma tabela com a soma de passageiros, segundo sexo e idade.
5. Considere a função `lapply()`.
 - a. Crie `listobj` uma lista de quatro matrizes, tal que:

```
first <- matrix(38:67, 3)
second <- matrix(56:91, 3)
third <- matrix(82:144, 3)
fourth <- matrix(46:93, 6)
```
 - b. Extraia a segunda coluna de cada matriz da lista de matrizes.
 - c. Extraia a terceira linha de cada matriz da lista de matrizes.
6. Usando a família *apply* para trabalhar com classes de *data frames* (usaremos o conjunto de dados `iris` visto em aula).
 - a. Descubra qual coluna em `iris` não é numérica.
 - b. Identifique os níveis da coluna não numérica. Dica: use função `levels()`.
 - c. Considere a função `unique()` e compare os resultados.
7. Considere o conjunto de dados `PlantGrowth` disponível no R. Use a função `tapply()` para calcular o desvio padrão dos valores de peso de cada grupo.

Desafio

8. O código a seguir simula o desempenho de um teste t para dados não normais. Use `sapply()` e uma função construída por você para extrair o p-valor de cada uma das 100 tentativas armazenadas em `trials`.

```
trials <- replicate(100, t.test(rpois(10, 10), rpois(7, 10)), simplify = FALSE)
```

9. Considere o exercício anterior. Produza o mesmo resultado livrando-se da função que você construiu e use `"["` diretamente.

Agradecimento

O material foi produzido pela Profa. Tatiana Benaglia para o curso de ME115.