

Lista de exercícios II - Subsetting - MAP2112

Introdução à Lógica de Programação e Modelagem Computacional

Prof. Luis Carlos de Castro Santos

Monitora Lucy Anne de Omena Evangelista

Universidade de São Paulo

18 de abril de 2020

Exercício 1. Possuindo a lista de elementos atômicos:

```
lisatom <- c(2,3,4,1,2,3,5,6,6,7)
```

Faça os seguintes filtros, lembrando que pode haver mais de uma forma para tal:

- a. Pegar o primeiro elemento.
- b. Pegar o ultimo elemento.
- c. Pegar o quarto, quinto e sexto elemento.
- d. Pegar o primeiro, segundo, terceiro, oitavo e nono elemento.
- e. Pegar todos os elementos, exceto o quinto.
- f. Pegar todos os elementos, exceto o quinto, sexto e sétimo.
- g. Pegar todos os elementos, exceto o segundo, quarto, sexto e décimo.
- h. Pegar todos os elementos, exceto o último.
- i. Pegar todos os elementos menores ou iguais a 4.
- j. Pegar todos os elementos menores ou iguais a 4 de forma ordenada.
- k. Pegar todos os elementos iguais a 3.
- l. Pegar todos os elementos iguais a 3 ou 6.
- m. Pegar todos os elementos iguais a 3 ou maiores do que 4.

Vale destacar que para cada filtro/subsetting, há sempre um vetor de valores booleanos (ou seja, com valores TRUE e FALSE) atrelada. Se entrarmos diretamente com esse vetor ao invés das operações desejadas, teremos o mesmo resultado. Não é necessário fazer isso para todos os filtros, mas é importante entender que isso está sendo feito pelo R. Toda lista de elementos atômicos é um vetor.

Exercício 2. Faça `setNames(lisatom, letters[1:10])` e verifique o valor de `lisatom` na posição 'a'. Faça `lisatom <- setNames(lisatom, letters[1:10])` e verifique o valor de `lisatom` na posição 'a'. Quais diferenças percebeu ao executar cada uma das operações? Por que acontecem?

- a. Pegue todos os elementos entre os índices "a" e "f" de `lisatom`.

Exercício 3. Suponha essa lista de coisas gostosas, cores, e alturas aleatórias:

```
lista <- list(c("bolo","chocolate","brócolis", "frango frito", "torta"),
             c("azul", "amarelo", "vermelho", "rosa", "roxo", "turquesa"),
             c(150, 167, 180, 190, 173))
```

- a. Coloque títulos nos índices da lista, a sua escolha.
- b. Qual a diferença de mostrar `lista[1]`, `lista[[1]]`, `lista$nome1` e `lista['nome1']` ?
- c. Verifique os tipos de cada sublista com a função `class()`.
- d. Mostre o primeiro e terceiro elementos de lista. Qual é o tipo de cada elemento?
- e. Selecione o 'brócolis' de pelo menos duas formas diferentes.
- f. Selecione a sua cor favorita no vetor de cores. Se ela não estiver lá, escolha 'turquesa' (se quiser, mas tem que escolher uma das cores).
- g. Substitua o 'brócolis' ou o 'frango frito' por outra coisa gostosa.

Exercício 4. Considere a seguinte matriz:

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

- a. Eu poderia escrever: `matriz <- matrix(4, 5, 1:20)` para definir a matriz? Se sim ou se não, por que? Se não, consigo encontrar uma forma de inserir os valores nessa posição?
- a. Mostre a matriz.
- b. Mostre os elementos da ultima linha.
- c. Mostre os elementos da ultima coluna.
- d. Mostre o primeiro elemento da primeira coluna.
- e. Mostre o terceiro elemento da terceira coluna.
- f. Mostre as 2 primeiras linhas das 3 primeiras colunas.
- g. Mostre as 2 primeiras linhas das colunas 1, 3, e 5.

h. Em qual das operações realizadas utilizar `drop=FALSE` seria útil? Qual é a função desse parâmetro?

i. Qual é a diferença de usar `matriz[c(1,2)]` e `matriz[c(1,2),]`?

j. Selecione todas as colunas em que o valor da primeira linha da matriz é maior do que 9.

k. Nomeie as linhas e colunas da matriz. Use `colnames()` e `rownames()` para isso.

Exercício 5. Considere o seguinte dataframe:

```
cores <- c("azul", "branco", 'verde', 'azul', 'branco', 'branco', 'verde',
'verde', 'azul', 'branco')
dono <- c("Ada", "Bertha", "Ada", "Cris", 'Cris', 'Bertha', 'Cris', 'Ada',
'Bertha', 'Cris')
uso <- c(5,8,3,9,0,1,3,6,6,4)
df <- data.frame(cores, dono, uso)
```

a. Quais são as diferenças de se utilizar um data frame ao invés de uma matriz?

b. Posso selecionar a coluna cores de quatro formas diferentes: `df['cores']`, `df[1]`, `df$cores`, `df[[1]]`. O que difere essas formas?

c. Mostre quais são os valores únicos de cada uma das colunas.

d. Selecione os três primeiros valores da coluna dono.

e. Mostre todas as informações onde a dona é a Bertha.

f. Mostre todas as informações onde a cor é branco e a dona é a Cris.

g. Mostre todos os usos onde a cor é azul.

Exercício 6. Conserte cada um dos erros comuns de subsetting em dataframes:

a. `df[df$primeira = 4,]`

b. `df[-1:4,]`

c. `df[df$primeira <= 5]`

d. `df[df$primeira == 4 | 6,]`

Exercício 7. Sendo `df` um dataframe, por que `df[1:20]` retorna um erro? No que isso difere ou é similar a `df[1:20,]`?

Exercício 8. O que faz o comando `df[is.na(df)] <- 0` ? Este comando funciona da forma que está?

Exercício 9. Descreva todas as formas que pensar sobre como extrair o terceiro valor de uma coluna 'primeiro' em um dataframe 'df'.

Exercício 10. Como é possível colocar as colunas de um dataframe em ordem alfabética?

Exercício 11. Por que `1 == '1'` é verdadeiro? Por que `-1 < FALSE` é verdadeiro? Por que `"one" < 2` é falso?

Exercício 12. O que a função `dim()` retorna quando aplicada em um vetor de uma dimensão? Quando `NROW()` ou `NCOL()` deve ser utilizado?

Exercício 13. Quais são todas as formas que uma lista é diferente de um vetor atômico?

Exercício 14. O que acontece se você tentar definir nome de linhas (`rownames`) que não são únicos em matrizes e em dataframes?