

Programação em R - Exercícios - Lista 1

Carlos Cinelli

Julho, 2016

1. Crie três variáveis no R com as três formas diferentes de criar objetos que você aprendeu:
 - Verifique se as variáveis foram criadas. Crie uma quarta variável concatenando as três variáveis.
 - Remova apenas as três primeiras variáveis do ambiente de trabalho com apenas um comando.
2. Crie variáveis de classes `numeric`, `integer`, `complex`, `character` e `logical` com tamanhos diferentes (maiores do que 1):
 - Verifique se as variáveis estão no ambiente de trabalho. Veja a estrutura, classe e tamanho dessas variáveis.
 - Aplique as funções `is.xxxx` e as funções `as.xxxx` aos objetos e verifique seu comportamento.
 - Crie um vetor numérico e dê nome a cada um dos elementos como “obs1”, “obs2”, utilizando a função `names()`. Utilize a função `paste()` para criar os nomes (olhe a ajuda em `?paste`). Faça alguns subsets do vetor usando os nomes.
 - Salve a variável numérica como `x.rds` usando a função `saveRDS()` e como `x.rda` usando a função `save()`. Tente ler a ajuda das funções para entender como funcionam (`?saveRDS` e `?save`). A seção de exemplo pode ser útil. Não se preocupe, aprenderemos essas funções com mais detalhes em outra aula, o objetivo é exercitar o uso da ajuda do R.
 - Remova todas variáveis do ambiente de trabalho com apenas um comando.
 - Carregue novamente a variável numérica usando a função `load()` no arquivo `x.rda`. Agora carregue usando a função `readRDS()` no arquivo `x.rds`. Você notou alguma diferença entre essas duas formas de carregar a variável?
3. Verifique as classes, estruturas, tamanho e o resultado da função `summary()` nos seguintes vetores:
 - `x1 <- 1:10`
 - `x2 <- rnorm(10)`
 - `x3 <- c("a", "b", "c")`
 - `x4 <- c(T, F, T)`
 - `x5 <- c(1, "a", 2)`
 - `x6 <- c(1, TRUE, 0)`
 - `x7 <- c(1, 2, 3, 4L)`
 - `x8 <- c(1L, 2, TRUE, "a")`
4. Construa os seguintes vetores com 100 observações de uma normal(0,1):

```
set.seed(1)
x <- rnorm(100)
y <- rnorm(100)
```

- Qual é o primeiro elemento de x? Qual o terceiro elemento de y?
- Delete os 10 primeiros elementos de x e de y.
- Qual é 56th elemento de x? Ele é maior do que 56th elemento de y?
- Qual é a média de x? Qual a diferença da média de x com a média de y?
- Qual é o desvio-padrão de x? Qual a variância de y?
- Qual é a correlação de x e y?
- Qual a soma, produto, soma acumulada e produto acumulado de x?
- Como você utilizaria a função `cumsum()` para calcular a média acumulada de x?
- Crie os vetores `x_cres` e `x_decres` com x ordenado de formas crescente e decrescente.
- Crie um vetor z concatenando x e y.
- Selecione apenas os valores de y maiores do que sua média.

- Selecione os valores de x maiores ou iguais a 0.5, ou menores do que -1.5.
- Selecione os valores de x maiores do que 0.5 e menores ou iguais a 1.
- Crie novos vetores com os resultados da soma, multiplicação, divisão, parte inteira da divisão, resto da divisão e exponenciação de x com y .
- Verifique se existem NA's ou NaN's nos vetores criados. Caso existam, substitua os NA's ou NaN's por 0.
- Crie um vetor de diferenças entre x sua primeira defasagem. Crie outro com a diferença da terceira defasagem. Crie um com as segundas diferenças do primeiro lag de x (veja o help da função `diff()`). Verifique a estrutura e tamanho dos vetores.
- Selecione apenas cada segundo elemento do vetor x .

5. Considere os vetores $x \leftarrow 1:6$ e $y \leftarrow 2:1$. Qual o tamanho de cada vetor? Qual o resultado de $x + y$? Explique o que aconteceu.

6. Crie os seguintes vetores (**Dica:** use `:`, `seq()`, `rep()` etc):

- 1, 2, 3, 4, ..., 10;
- 10, 9, 8, 7, ..., 1;
- 2, 4, 6, 8, ..., 200;
- 10.5, 9.5, 8.5, ..., 0;
- de 1.06 até 2.98 com 67 elementos;
- 1, 7, 1, 7, ... de tamanho 140.
- "a", "b", "a", "b", ... de tamanho 10
- 1, 1, 1, ... 100 vezes junto com 5, 5, 5, ... 25 vezes.
- 1, 1, 1, 2, 2, 2, ..., 10, 10, 10;

7. Crie um vetor de 1 até 100 e o transforme em matrizes quadráticas conforme ilustrado abaixo.

(**Dica:** use as opções `ncol`, `nrow` e `byrow` da função `matrix()`)

Ordenada por colunas:

#	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]
# [1,]	1	11	21	31	41	51	61	71	81	91
# [2,]	2	12	22	32	42	52	62	72	82	92
# [3,]	3	13	23	33	43	53	63	73	83	93
# [4,]	4	14	24	34	44	54	64	74	84	94
# [5,]	5	15	25	35	45	55	65	75	85	95
# [6,]	6	16	26	36	46	56	66	76	86	96
# [7,]	7	17	27	37	47	57	67	77	87	97
# [8,]	8	18	28	38	48	58	68	78	88	98
# [9,]	9	19	29	39	49	59	69	79	89	99
# [10,]	10	20	30	40	50	60	70	80	90	100

Ordenada por linhas:

#	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]
# [1,]	1	2	3	4	5	6	7	8	9	10
# [2,]	11	12	13	14	15	16	17	18	19	20
# [3,]	21	22	23	24	25	26	27	28	29	30
# [4,]	31	32	33	34	35	36	37	38	39	40
# [5,]	41	42	43	44	45	46	47	48	49	50
# [6,]	51	52	53	54	55	56	57	58	59	60
# [7,]	61	62	63	64	65	66	67	68	69	70
# [8,]	71	72	73	74	75	76	77	78	79	80
# [9,]	81	82	83	84	85	86	87	88	89	90
# [10,]	91	92	93	94	95	96	97	98	99	100

8. Considere os seguintes vetores gerados de uma distribuição t com 1 e 2 graus de liberdade:

```
set.seed(10)
x <- rt(10, df = 1)
y <- rt(10, df = 2)
```

- Crie uma matriz xy combinando x e y como linhas. Qual é a estrutura desta matriz?
 - Crie uma matriz XY combinando x e y como colunas. Qual é a estrutura desta matriz?
 - Crie um vetor z concatenando x e y. Crie com z uma matriz com duas linhas. Crie com z uma matriz com duas colunas.
 - Transforme x e y em matrizes de 10 linhas e uma coluna. Crie uma matriz m com o resultado de x vezes a transposta de y. Notar que a multiplicação é matricial.
 - Selecione: (i) a linha 10 de m; (ii) a coluna 5 de m; (iii) o elemento na terceira linha e segunda coluna; os elementos de m maiores do que zero; (iv) os elementos de m menores do que zero e calcular a média; (vi) a diagonal de m.
9. Crie 2 matrizes, A e B, com 5 linhas e 5 colunas e elementos de uma normal(0,1). Antes de gerar os valores aleatórios, defina a semente `set.seed(1)`. (**Dica:** uma matriz 5 por 5 vai precisar de 25 elementos; a função para gerar dados de uma normal é `rnorm` e a função para criar uma matriz é `matrix`).
- Quais os resultados das funções `length()`, `ncol()`, `nrow()`, `dim()`, `str()`, `min()`, `max()`, `summary()` e `is.matrix` nas matrizes?
 - Crie matrizes A2 e B2 compostas dos elementos de A e B em valores absolutos;
 - Crie uma matriz C com a soma (elemento a elemento) de A e B;
 - Crie uma matriz D com a multiplicação (elemento a elemento) de A e B;
 - Inverta as matrizes. Calcule o determinante das matrizes;
 - Crie uma matriz A3 excluindo a primeira linha e a primeira coluna de A;
 - Calcule a média dos elementos de A tais que $A[i] \leq B[i]$;
 - Crie uma matriz C tal que $C[i] = 1$ se $A[i] \geq B[i]$ e 0 caso contrário;
 - Utilize a função `rownames()` para nomear as linhas de A como “linha1”, “linha2” ..., e a função `colnames()` para nomear as colunas de A como “coluna1”, “coluna2” Utilize a função `paste()` para criar os nomes. Faça alguns subsets utilizando os nomes.

10. Suponha uma lista do tipo

```
lista <- list( elemento1 = rnorm(10),
              elemento2 = 1L,
              elemento3 = "a",
              elemento4 = c(1,"b"),
              elemento5 = c(TRUE, TRUE, FALSE),
              elemento6 = c(10i, 20i))
```

- Quais os resultados de `names()`, `length()`, `str()`, `dim()`, `summary()` e `is.list()` na lista?
- Qual o primeiro elemento da lista? E o terceiro? Quais as formas de selecionar estes elementos?
- Os elementos da lista são de classes diferentes? Quais as classes de cada elemento?
- Adicione um novo elemento na lista, chamado `elemento7`, com o resultado de `elemento1` mais `elemento2`.
- Delete o `elemento1` da lista (**Dica:** atribuir o valor NULL a um elemento da lista deleta este elemento. Isso também vale para `data.frames`!).
- Existe uma função no R chamada `unlist` que, como diz o nome, “desfaz” a lista, a transformando em um vetor simples. Qual o objeto resultante de `unlist(lista)`?

11. Listas podem ter estruturas arbitrariamente complexas, com listas dentro de listas. Considere a seguinte lista2.

```
lista2 <- list( lista_dentro_da_lista = lista,
               outra_lista = list(numero=c(1,2,4),
                                   mais_lista=list(c(1,3,4), TRUE)))
```

- Quais os resultados de `names()`, `length()`, `str()` e `summary()` na `lista2`?
- Como acessar o `elemento1` contido em `lista_dentro_da_lista` que está em `lista2`?
- Como acessar o segundo elemento do objeto `mais_lista` dentro de `outra_lista` da `lista2`?

12. Rode o seguinte comando para criar um `data.frame`:

```
set.seed(14)
df <- data.frame(x = rnorm(10), y = rnorm(10))
```

- Verifique os resultados de `length()`, `nrow()`, `ncol()`, `dim()`, `class()`, `is.data.frame()`, `str()`, `summary()` em `df`. Quantas observações temos na base de dados? Quantas variáveis? Qual a classe do objeto? Quais as classes das variáveis?
 - Crie um vetor `z <- rlnorm(10)`. Adicione este vetor como mais uma variável, chamada `z`, em `df`? Quais as diferentes formas de fazer isso?
 - Como adicionar mais uma linha com `x=1`, `y =2`, e `z=3` ao `data.frame`?
 - Crie uma variável `w` em `df` com o resultado de `x+y+z`.
 - Delete a variável `x` de `df` (**Dica:** veja a dica do exercício 10!).
 - Delete as últimas cinco linhas de `df`.
13. Carregue a base de dados `wi.venda.rds` em um objeto chamado `dados`. Estes dados são de oferta online de apartamentos no Plano Piloto (Asa Sul, Asa Norte, Sudoeste e Noroeste).
- Verifique os resultados de `length()`, `nrow()`, `ncol()`, `dim()`, `class()`, `is.data.frame()` em `dados`. Quantas observações temos na base de dados? Quantas variáveis? Qual a classe do objeto?
 - Verifique os resultados de `names()` e `colnames()`. São os mesmos?
 - Aplique `str()`, `summary()` em `dados` para entender melhor quais são os dados do `data.frame`. Quais as classes de cada coluna?
 - Quais foram a média, mediana, desvio-padrão e variância dos preços dos anúncios de apartamento no Plano Piloto neste dia? (faça com `with()`, `$` e `[]`)
 - Quais foram a média, mediana, desvio-padrão e variância do número de quartos dos anúncios? (faça com `with()`, `$` e `[]`)
 - Quais foram a média, mediana, desvio-padrão e variância do tamanho dos apartamentos? (faça com `with()`, `$` e `[]`)
 - Qual a correlação entre preços e metro quadrado? (faça com `with()`, `$` e `[]`)
 - A coluna `bairro` contém a informação dos bairros dos anúncios. Quantos anúncios temos na Asa Sul (em número e percentual)? Quantos anúncios temos na Asa Norte (em número e percentual)? (faça com `with()`, `$` e `[]`)
 - Crie um vetor chamado `preco_asa_sul` com os preços da Asa Sul. E outro chamado `preco_asa_norte` com os preços da Asa Norte. Qual dos bairros tem a maior mediana de preços de apartamento? (faça com `with()`, `$` e `[]`)
 - Crie uma coluna `pm2` em `dados` com o resultado de `preco` dividido por `m2` e repita o exercício anterior (criando os vetores `pm2_asa_sul` e `pm2_asa_norte`). Qual dos dois bairros tem a maior mediana de preços por metro quadrado? (faça com `with()`, `$` e `[]`)
 - Delete a coluna `link` de `dados`.
14. **Exercício extra:** `data.frame` é uma lista.
- Rode o comando `is.list()` em `dados`. Qual o resultado?
 - A função `unclass()` retira a classe “especial” dos objetos revertendo-o à sua classe mais básica. Rode o seguinte código abaixo. Note que um `data.frame` nada mais é do que uma lista com certas características (por exemplo: atributo `row.names`; cada coluna com o mesmo número de linhas).