

Lista - data-frames 2

Ana C V de Melo

Introdução à Computação - R

objetivo: manipulação dos data-frames

Para fazer esses exercícios V. precisa ter assistido às aulas da semana e/ou consultado material didático sobre data-frames.

Exercício 1

1. Abra o programa RStudio.
2. Na janela de *Files* (canto direito inferior) crie um diretório com seu nome/número da lista.
3. Copie do *moodle* todos os arquivos disponíveis da Lista atual. Veja que você tem pelo menos um script (programa) para a aula.
4. Atualize o diretório de execução do RStudio para a sessão atual: clique em *Session/Set Working Directory/Choose Directory...* e escolha o diretório onde colocou os seus arquivos.
5. Abra o script para iniciar a lista.

A sua tarefa de hoje será complementar o script para o que se pede nos próximos exercícios.

Exercícios

1. No seu script,
 - leia os dados da planilha *alunosnotas.csv* e armazene os dados na variável *dadosalunos*. Lembre-se de que essa planilha possui cabeçalho.
 - Imprima a planilha para ver seu conteúdo.

```
dadosalunos <- read.csv(file = "alunosnotas.csv",  
                        ... complete ...) # lê a planilha com cabeçalho  
print(dadosalunos)
```

 - confira se a variável é um data-frame, usando a função predefinida (*is.data.frame()*). Essa função dá como resultado verdadeiro (TRUE) ou falso (FALSE), dependendo se a variável é um data-frame ou não.
2. Os operadores relacionais podem ser aplicados a 2 valores (expressões - valor, variável ou uma expressão). Veja, por exemplo, o valor resultado das seguintes expressões relacionais:

```
a <- 18.7  
b <- 4.8  
a > b  
a <= 20  
(a + 20)/b >= 10  
(a + 10)/b > ((2*a + 20) * (1/(2*b)))
```

3. Os operadores relacionais também podem ser aplicados a vetores na linguagem R. Veja, por exemplo,

```
v1 <- seq(1,16, by=.8)
v2 <- seq(1,16, length.out = 19)
print(v1)
print(v2)
v1 > 5
(v1 + 0.3) >= v2
```

quando aplicados a vetores, resultam em um vetor de valores *TRUE* ou *FALSE*, observe os resultados.

Podemos também extrair subconjuntos (*subset*) de vetores que satisfazem a uma determinada expressão lógica. Apenas os que resultam em *TRUE* serão selecionados no subconjunto:

```
subset(v1, v1 > 5)
subset(v1, (v1 + 0.3) >= v2 )
subset(v2, (v1 + 0.3) >= v2 )
```

4. Sabemos que cada coluna de um data-frame é um vetor de valores. Por isso, podemos usar os operadores relacionais sobre as colunas dos data-frames. Considerando o data-frame *dadosalunos*, formule (e execute) as expressões para cada um dos itens abaixo:
- elabore a expressão lógica que compara se a nota da P1 é maior que a nota da P2;
 - selecione o subconjunto de notas da P1 que foram maiores que as notas da P2;
 - elabore a expressão lógica que compara se a nota da Sub é menor que as notas da P2 ou da P1;
 - selecione o subconjunto de notas Sub que são menores que as notas da P2 ou da P1;
 - elabore a expressão lógica que compara se a nota da Sub é maior ou igual às notas da P2 e da P1;
 - selecione o subconjunto de notas Sub que são maiores que as notas da P2 e da P1;
5. Além de poder construir subconjuntos de vetores, podemos também construir subconjuntos de data-frames utilizando *subset()* e expressões lógicas. Temos também, para a elaboração desses subconjuntos de data-frames, o operador *%in%* e a função *which()*. A partir do data-frame *dadosalunos*, mostre os seus subconjuntos onde:
- as notas da P1 são maiores que as notas da P2;
 - as notas da Sub são menores que as notas da P2 ou da P1;
 - as notas da Sub são maiores que as notas da P2 e da P1;
 - contém apenas os alunos *c("Marcos", "Rafael", "Maria", "Denis")*
 - em quais linhas do data-frame *dadosalunos*, as notas da P1 são menores que 5?
6. A partir do data-frame *dadosalunos*, crie os novos data-frames com as seguintes restrições:
- alunos_com_sub*: contém info apenas dos alunos que fizeram a Sub
 - alunos_sem_sub*: contém info apenas dos alunos que NÃO fizeram a Sub
 - alunos_sub_maior5*: contém info apenas dos alunos que na Sub obtiveram nota maior que 5
 - alunos_p1_p2*: contém info apenas dos alunos que fizeram a P1 e a P2 (podem ter feito a Sub ou não)
 - alunos_p2_sub*: contém info apenas os alunos que fizeram exclusivamente a P2 e a Sub (não fizeram a P1)
 - alunos_p1_p2_sub*: contém info apenas os alunos que fizeram as 3 provas
- Imprima na tela cada um dos data-frames criados para conferir os valores. Para as operações com a função *subset()*, que extrai parte dos dados dos data-frames, usamos os operadores lógicos: *&* (and) e *|* (or). Lembre-se de que conhecemos também a função *is.na()*.
7. Mostre os dados sobre as 3 provas (P1, P2, Sub) de forma gráfica, utilizando as funções: *plot()*, *barplot()* e *hist()*. Veja no *help* cada uma dessas funções e suas possíveis configurações.