

Laboratório de Bioestatística - Introdução ao R

EXERCÍCIOS GUIADOS

Operações Simples

1. Crie um vetor `w` com as componentes 1,-1,2,-2.
`> w<-c(1,-1,2,-2)`
2. Imprima esse vetor na consola do R.
`> print(w)`
3. Obtenha uma descrição de `w` usando a função `str()`.
`> str(w)`
4. Crie o vetor `w+1` e imprima na consola do R.
`> x<-w+1`
`> print(x)`
5. Crie o vetor (0,1,5,10,15,...,75) usando as funções `c()` e `seq()`.
`> j<-seq(5,75,by=5)`
`> j<-c(0,1,j)`
`> print(j)`
6. Crie um vetor alfanumérico com as etiquetas "Paciente", "Número de entrada" e "Doença", com separador : e com dimensão 10.
`S<-paste("Paciente", "Número de entrada", "Doença",sep=":",1:10)`
7. Considere as seguintes condições:
 - a) $x > 23$
`> a <- x>23`
 - b) $x \in [-1, 10]$
`> b <- x>=-1 & x<=10`
 - c) $x = 3$
`c<-x==3`
 - d) $x \leq 45$
`d <- x<=45`Indique as condições lógicas:
 - (a) $a \cap b$
`> a&b`
 - (b) $b \cup c$
`> b|c`
 - (c) $d \cap b$
`> d&b`

Vetores, Matrizes e Data Frames

8. Crie um vetor numérico, um vetor *string* e um vetor lógico:


```
> valor.num <- c(3,4,2,6,20)
> valor.char <- c("koala","kangaroo","monkey")
> valor.logic.1 <- c(F,F,T,T)
```
9. Crie o seguinte vetor (0, 4, 2, 1, 0, 4, 0, 3, 0, 3, 3, 3, 4, 4, 2, 2, 0) imprima-o na consola do R.


```
> y<-c(0, 4, 2, 1, 0, 4, 0, 3, 0, 3, 3, 3, 4, 4, 2, 2, 0)
> y
```
10. Quantos elementos de y são inferiores a 3.


```
>length(y[y<3])
```
11. Procure valores de y menores ou iguais a 3.


```
> y[y<=3]
```
12. Procure valores de y iguais a zero.


```
> y[y==0]
```
13. Procure valores diferentes de zero.


```
> y[y!=0]
```
14. Coloque os seguintes valores (12,14,35,7,6,12,5,22,7,17,9,11) sob a forma de uma matriz de ordem 3×4 e imprima-a na consola do R.


```
> m<-matrix(c(12,14,35,7,6,12,5,22,7,17,9,11),ncol=4)
> m
```
15. Qual é o elemento da matriz m que está na posição “linha 2, coluna 3”?


```
> m[2,3]
```
16. Crie uma *data frame* (não recorra ao editor do R) que corresponde às notas de 4 alunos, à turma a que pertencem e ao número de aluno:

n.º aluno	turma	notas
2355	tp1	0.3
3456	tp1	9.3
2334	tp2	14.2
5456	tp3	15.0

```
> pauta<- data.frame(nros = c(2355, 3456, 2334, 5456),
+ turma = c("tp1", "tp1", "tp2", "tp3"),
+ notas = c(0.3,9.3,14.2,15.0))
> pauta
```
17. Relativamente à *data frame* **pauta** qual é o elemento que está na linha 2 e coluna 2.


```
> pauta[2, 2]
```
18. Aceda à coluna correspondente ao número dos alunos.


```
> pauta$nros
```

19. Qual é a informação que obtém sobre a *data frame* `notas` quando executa os seguintes comandos:

```
> pauta[pauta$notas > 10, ]  
> pauta[pauta$notas > 14, "nros"]  
> pauta[pauta$turma == "tp1", c("nros", "notas")]
```
20. Execute o comando que permite aceder diretamente às colunas da *data frame* `pauta`

```
> attach(pauta)
```
21. Acrescente uma nova coluna correspondente aos resultados finais: "aprovado", "oral", "aprovado", "aprovado", mas agora a partir do editor do R.

```
> pauta<-edit(pauta)
```
22. Consulte as variáveis da *data frame* anterior.

```
> names(pauta)
```

Entrada e Saída de Dados

23. Aceda aos dados `ChickWeight` da biblioteca `datasets` e verifique qual a informação subjacente a esta base de dados.

```
> library(datasets)  
> data(ChickWeight, package='datasets')  
?ChickWeight
```

Funções Gráficas

Considere a *data frame* `births` da biblioteca `Epi`. Esta base de dados diz respeito a nascimentos de bebés num hospital em Inglaterra.

24. Identifique as variáveis da *data frame* `births`:

```
> library(Epi)  
> data(births)  
> str(births)
```
25. Construa um histograma para a variável `bweight`:

```
> hist(births$bweight)
```
26. Consulte algumas das opções da função `hist()` e construa um histograma com algumas das opções disponíveis.

```
> ?hist  
> hist(births$bweight, col="gray", border="white")
```
27. Verifique a relação entre o peso dos bebés e a semana gestacional da *data frame* `births`:

```
> attach(births)  
> plot(gestwks, bweight)
```

28. Construa um gráfico com o peso dos bebês e a idade da mãe:
`>plot(matage,bweight)`
29. Acrescente ao gráfico anterior as etiquetas dos eixos:
`>plot(matage,bweight,xlab="Idade da mãe", ylab="Peso dos
+ bebês (g)")`
30. Relativamente ao gráfico `>plot(matage,bweight)` altere a cor dos pontos para verde.
`>plot(matage,bweight,pch=16,col="green")`
31. Coloque agora os pontos a preto e em forma de círculos:
`>points(matage,bweight,pch=19)`
32. Considere o gráfico `>plot(gestwks,bweight)` e defina cores diferentes para os bebês do sexo feminino e masculino:
`>plot(gestwks,bweight)
>points(gestwks[sex==1],bweight[sex==1],col="blue")
>points(gestwks[sex==2],bweight[sex==2],col="red")`
33. Acrescente uma legenda e um título ao gráfico:
`>legend("topleft",pch=1,legend=c("rapazes","raparigas"),
+ col=c("blue","red"))
>title("Peso à nascença vs semana gestacional em 500 nascimentos
+ individuais")`
34. Utilize uma função que, de forma mais rápida, adicione diferentes cores a diferentes níveis de uma variável qualitativa.
Criar um vector de cores com a mesma dimensão da variável qualitativa
`>plot(gestwks,bweight,pch=16,col=c("blue","red")[sex])`
35. Defina diferentes símbolos para as mães com idade superior ou igual a 40 anos e para as mães com idade inferior.
Em primeiro lugar tem de se especificar uma função tal que identifique as idades das mães com mais de 40 anos
`>mães.mais.quarenta<-(matage>=40)+1
Adicionamos 1 para que a função seja numérica com valores 1 e 2
>plot(gestwks,bweight,pch=c(3,18)[mães.mais.quarenta],
+col=c("blue","red")[mães.mais.quarenta]
#Também é possível criar funções mais complexas dentro de funções. Em vez de se ter criado a função "mães.mais.quarenta" era possível fazer:
>plot(gestwks,bweight,pch=c(3,18)[(matage>=40)+1],
+col=c("blue","red")[(matage>=40)+1])`
36. Utilize uma função que permita identificar as coordenadas dos pontos num gráfico:
`>identify(gestwks,bweight)`

37. Guarde o gráfico correspondente ao peso dos bebés e às semanas gestacionais num ficheiro PDF. `pdf(file="plot.1",height=3,width=4)`
`>plot(gestwks,bweight)`
`>dev.off()`
38. Construa dois gráficos no mesmo *device* correspondentes a: (1) idades das mães vs semanas de gestação e (2) idades das mães vs peso à nascença:
`>par(mfrow=c(1,2))`
`>plot(matage~(gestwks+bweight))`

Ciclos e Funções

39. (a) Crie um ciclo, recorrendo à função `for()`, que dado um vetor numérico imprima um número por linha juntamente com o seu quadrado e o seu cubo.

```
> x<-c(1,3,5) # por exemplo
> n<-length(x)
> for(i in 1:n)
+ cat(x[i], ' ao quadrado =', x[i]^2, '; ao cubo =', x[i]^3, "\n")
```

- (b) Use a função `while()` para alcançar o mesmo resultado.

```
> i<-1 # inicializar o contador
> while(i<(n+1))
+ {
+ cat(x[i], ' ao quadrado =', x[i]^2, '; ao cubo =', x[i]^3, "\n")
+ i<-i+1
+ }
```

40. Crie as funções F1 e F2, cujos argumentos de entrada são x (vetor (x_1, x_2, \dots, x_n)) e n , com os *output* $(x_1, x_2^2, \dots, x_n^n)$ no caso da função F1 e $1 + x_1 + \frac{x_2^2}{2} + \dots + \frac{x_n^n}{n}$ no caso da função F2.

```
> x<-c(3,6,9) # por exemplo
> n<-length(x)
> F1 <- function(x,n)
+ x^(1:n)
> F1(x,n)

> F2 <- function(x, n)
+ 1 + sum((x^(1:n))/(1:n))
> F2(x,n)
```

41. Escreva uma função que tenha um único argumento: uma matriz. A função deve devolver uma matriz idêntica à original, mas em que cada número ímpar é duplicado. Por exemplo, a função aplicada à matriz

$$\begin{pmatrix} 1 & 1 & 3 \\ 5 & 2 & 6 \\ -2 & -1 & -3 \end{pmatrix}$$

resulta em

$$\begin{pmatrix} 2 & 2 & 6 \\ 10 & 2 & 6 \\ -2 & -2 & -6 \end{pmatrix}$$

```
> Func <- function(mat)
+ {
+   mat[mat%%2 == 1] <- 2 * mat[mat%%2 == 1]
+   mat
+ }
> mat<-matrix(c(1,5,-2,1,2,-1,3,6,-3),3,3)
> Func(mat)
```

NOTA: o operador %% devolve o resto da divisão, e.g., $5\%2$ é igual a 1.

EXERCÍCIOS LIVRES

Operações Simples

1. Realize as seguintes operações atribuindo sempre um objecto à expressão:

(a) $4 + 5 * (\log_{10}(4) - e^3)$

(b) $\sqrt{37} + 4 + 5 * (\log_{10}(4) - e^3)$

(c) $\sin(\pi \div 2) + \cos(\frac{2\pi}{3})$

2. Verifique os objectos que criou na sessão de trabalho.

3. Crie uma sequência de valores por ordem decrescente entre 17 e 26.

4. Quantos elementos tem a sucessão de valores anterior?

5. Crie uma sucessão de valores entre -15 e 89, cujo espaçamento é de 0.5.

6. Crie uma sucessão de 30 valores a partir de 15 com um espaçamento de 2.3.

7. Crie um vetor alfanumérico com as etiquetas "Paciente" e "Centro de Saúde" com separador nulo e com dimensão 20.

8. Considere as seguintes condições:

(a) $x > 13$

(b) $x < 13 \wedge x = 23$

(c) $x \in [0, +\infty[$

Qual a condição lógica das seguintes expressões:

- $(a) \cap (c)$

- $(b) \cup (c)$

- $\sim (b) \cap (c)$

Vetores, Matrizes e Data Frames

9. Crie dois vetores para as variáveis:

peso	62	70	52	98	90	70
altura	1.70	1.82	1.75	1.94	1.84	1.61

10. Quantos elementos têm mais de 90kg e quantos têm menos de 1.70m.
11. Para o vetor **peso** altere o elemento que está na posição 4 para 1.70.
12. Calcule o Índice de Massa Corporal (IMC) utilizando os vetores anteriores e guarde os resultados num objecto. Que tipo de objecto é que é o IMC e qual é a sua dimensão.
13. Para o vetor IMC acrescente o valor 23.
14. Para o vetor IMC atribua um nome para cada elemento de acordo com a seguinte tabela:

Valor IMC	Estado
20-24	Peso normal
25-29	Excesso de peso
30-35	Obesidade
≥ 35	Super obesidade

15. Crie o vetor **x** de modo que contenha os seguintes valores: -2,-1,5,6,7,...,15.
16. Crie o vetor **tipo.sangue** com os seguintes elementos:

A	A	AB	AB	O	B	B
A	B	O	O	O	AB	AB
A	B	O	O	O	A	A

Identifique o tipo e a dimensão.

17. É possível concatenar os vetores **tipo.sangue** e **peso**? Em caso afirmativo realize a operação.
18. Transforme o vetor **tipo.sangue** num fator. Quantos níveis tem este fator. Se quiser obter um fator do tipo ordinal, que função terá de utilizar?
19. Verifique para que servem as seguintes funções: **solve(A,B)** e **colSum()**.
20. Considere as seguintes matrizes:

$$A = \begin{pmatrix} 5 & 8 & 0 \\ -2 & 7 & 0.5 \\ 4 & 3 & 3 \end{pmatrix}, B = \begin{pmatrix} 3 & 2 \\ 0 & -2 \\ 7 & 3 \end{pmatrix}, C = \begin{pmatrix} -3 & 4 \\ 0.3 & 5 \\ 7 & 12 \end{pmatrix}$$

Realize as seguintes operações matriciais:

- (a) $A \times B$
- (b) $(B + C) \times 2C$
- (c) B^{-1}

- (d) $\det(A)$
- (e) $(\det(A \times B^T))$
- (f) Obtenha a matriz adjunta de A (\hat{A}). (Nota: $A^{-1} = \frac{\hat{A}}{\det(A)}$)
- (g) Calcule a média das linhas de A.
- (h) Identifique os elementos a_{21}, b_{13}, c_{12} .
- (i) Selecione a coluna 3 da matriz A.
- (j) Selecione a linha 2 da matriz C.
- (k) Selecione as colunas 2 e 3 da matriz B.
- (l) Atribua os seguintes nomes às colunas de A: Coluna1, Coluna2, Coluna3 e os seguintes nomes às linhas de B: Linha1, Linha2 e Linha3.

21. Construa uma lista com a seguinte informação:

Uma experiência conduzida pelo laboratório "ExperiênciasMil", consistiu na análise de três tipos de fármacos para o tratamento da diabetes. A um voluntário do sexo feminino, com 43 anos, foi administrado os três medicamentos, considerando períodos de lavagem de 3 semanas entre cada administração. Após cada toma foi avaliada a glicémia e para cada fármaco obtiveram-se os seguintes resultados: 167, 245, 165. Foi ainda avaliada a glicémia antes dos tratamentos: 345.

22. Realize as seguintes operações:

```
> data(iris)
> plantas<-iris
> plantas
```

- (a) A partir do editor do R altere o nome das variáveis para português.
- (b) Qual a dimensão da *data frame*?
- (c) Aceda só aos primeiros 15 registos.
- (d) Liste só a variável correspondente às espécies.
- (e) Identifique as espécies que têm um comprimento da pétala superior a 6.
- (f) Proceda à operação `attach()` para aceder diretamente às variáveis. Para a espécie "setosa", quantas plantas desta espécie têm a largura da sépala inferior a 3?
- (g) Proceda à operação `detach()`, realize as seguintes operações e verifique o que elas fazem:


```
> by(data=plantas$tamanhopétala, INDICES=plantas$espécie,
      +FUN=summary)
> boxplot(plantas$tamanhopétala~plantas$espécie, xlab="Espécie",
      +ylab="Tamanho da Pétala", main="Tamanho da pétala por espécie")
```
- (h) Guarde a *data frame* `plantas` usando a função `write.table()`.

Entrada e Saída de Dados

23. Aceda à *data frame* `Orange` da biblioteca `datasets`.

24. Construa uma tabela de contingência para as variáveis "year" e "sex" da *data frame* **Melanoma**, que se encontra na biblioteca **MASS**.
25. Leia os ficheiros que se encontram nos seguintes endereços da Internet:
"http://personality-project.org/r/datasets/maps.mixx.epi.bfi.data"
"http://socserv.socsci.mcmaster.ca/jfox/Courses/UCLA/Duncan.txt"
e guarde-os.

Funções Gráficas

26. Considere ainda a *data frame* **births** e crie um factor tal que identifique a idade das mães entre 30 e 40 anos.
27. Use a função anterior para construir um gráfico para as semanas gestacionais e onde se identifiquem as mães com idades entre os 30 e 40 anos.
28. Quais são os valores mínimo e máximo para as semanas gestacionais?
29. Considere a *data frame* **Animals** da biblioteca **MASS**. Construa um gráfico com legendas nos eixos para o peso do cérebro vs o peso do corpo.
30. Repita o passo anterior mas agora considere os logaritmos neperianos das variáveis. Utilize as etiquetas das linhas para identificar os pontos com maior peso do corpo.
31. Altere a cor de fundo do gráfico anterior.
32. Repita os gráficos anteriores mas agora lado a lado na mesma página e guarde-os num ficheiro JPEG.
33. Considere a *data frame* **Rabbit** da biblioteca **MASS**. Esta base de dados diz respeito ao estudo de 5 coelhos observados em duas ocasiões: depois do tratamento com um placebo e depois de um tratamento com 5-HT₃. Após cada tratamento, uma dose de *phenylbiguanide* foi injetada em intervalos de 10 minutos (em valores ascendentes) sendo registada a pressão sanguínea.
 - (a) Construa um gráfico que corresponda à pressão sanguínea dos coelhos vs a dose.
 - (b) Para o gráfico anterior acrescente título e etiquetas dos eixos.
 - (c) Adicione ao gráfico anterior um título, altere a cor dos pontos para os coelhos que levaram uma dose de 200mg, acrescente na margem de baixo do gráfico: "Consideraram-se os coelhos com os dois tratamentos".

Ciclos e Funções

34. A seguinte função calcula a média e o desvio padrão de um vetor numérico.

```
> media.dp <- function(x){  
+ m <- mean(x)  
+ dp <- sd(x)  
+ c(média=m, desvio_padrao = dp) }  
>  
> x<-c(2,4,6,8,10,12)    # por exemplo  
> media.dp(x)
```

Modifique a função de forma que:

- (a) seja aplicada automaticamente a um vetor de 20 números gerados com base na distribuição normal padrão, `rnorm()`, e calcule apenas o desvio padrão;
- (b) se houver valores omissos, `NA`, a média e o desvio padrão são calculados a partir dos restantes valores.