Exercícios 4

Manipulação de ficheiros e de dados

Em termos computacionais, a informação é habitualmente transmitida em blocos a que denominamos ficheiros. O **R** disponibiliza uma vasta colecção de funções para manipular ficheiros de diferentes formatos. Neste exercício, pretende-se fazer uso de algumas funções de manipulação de ficheiros, e dos objectos resultantes do seu carregamento numa sessão de **R**.

Para treinar esses conceitos, considere o papel de um analista de dados desportivos. A sua tarefa é recolher, preparar e analisar um conjunto de dados da *Primeira Liga* de futebol portuguesa (Figura 1).

Figura 1: Preferências clubísticas à parte, eis o que os dados revelam.

Recorde algumas das funções que lhe poderão ser úteis na resolução deste tipo de problemas (use o sistema de ajuda para verificar como as usar):

Operações matemáticas comuns mean, sum, colMeans, rowMeans

Outros estatísticos min, max, sd, var

Propriedades de objectos tabulares nrow, ncol, names, colnames, rownames

Sumarizar e contar summary, str, table, margin.table, which, which.max, which.min

Aplicar funções sobre objectos apply, sapply, tapply

Ordenação sort, order, rank

Agregação e manipulação by, aggregate, transform

Manipular strings substr, strsplit, paste, nchar

1. Recolha os dados existentes na seguinte tabela das últimas 25 temporadas:

   `http://www.european-football-statistics.co.uk/atc/atcprt.htm`

   Esta tabela contém alguns dados sobre as 43 equipas que participaram nessas temporadas. Os dados estão agrupados nas seguintes variáveis:

   No. Posição

   Club Nome da equipa

   G Jogos

   W Vitórias

   D Empates

L Derrotas

P Pontos

F Golos marcados

A Golos sofridos

S Temporadas

1 Vencedor

2 Segundo lugar

3 Terceiro lugar

Cup Taças de Portugal

Existem várias formas possíveis de recolher dados de uma página na *World Wide Web*. Uma delas, usando o **R**, e para o caso de os dados estarem em forma de uma tabela, é através da função `readHTMLTable(url)`, que é disponibilizada no pacote (*package*) XML.

Nota: O **R** já vem com alguns pacotes instalados. Pode instalar um novo pacote usando a função `install.packages('nome-do-pacote')`. Para importar o conteúdo de um pacote para a sessão actual, usa-se a função `library(nome-do-pacote)`. Repare que no primeiro caso o nome do pacote é passado como `character`, enquanto que no segundo caso não.

Solução:

```
url = 'http://www.european-football-statistics.co.uk/atc/atcprt.htm'
tabelas = readHTMLTable(url, stringsAsFactors=FALSE)
```

2. Verifique a estrutura da operação anterior. Irá reparar que a função `readHTMLTable()` procura ler todas as tabelas existentes numa página, mesmo as que não sejam óbvias para o leitor, mas que sejam tabelas por definição (código HTML). Guarde a tabela de interesse numa variável chamada `liga`.

Solução:

Esta função devolve uma lista, tal como se pode verificar com a função `str()`.

```
str(tabelas)
liga = tabelas[[3]]
```

3. Uma vez que os dados estejam importados, é necessário confirmar se estão no formato certo e que não há irregularidades.

   Parece que a coluna 8 não é mais do que um artefacto na tabela. Remova-a da tabela.

Solução:

Basta escolher todas as colunas menos a que se pretende apagar.

```
liga = liga[-8]
```

Alternativamente, também se pode igualar a coluna que se pretende apagar a `NULL`. Esta abordagem não funcionaria caso se tratasse de uma matriz e não de um data frame.

```
liga[8] = NULL
```

4. Verifique o tipo de dados em cada uma das colunas. Corrija as colunas que não estiverem no formato mais adequado (por exemplo, uma coluna numérica estar no formato de texto).

> **Solução:**
>
> Solução dada em conjunto com o exercício seguinte.
>
> **Nota:** No caso de resolver este exercício em separado do exercício 5, dependendo de como resolver, é possível que alguns números sejam importados de forma errada (e.g., 1.908 em vez de 1908). Assim, sugere-se que resolva os dois em simultâneo, como nesta resolução, ou primeiro o 5 e depois o 4.

5. Frequentemente, os dados incluem erros ou inconsistências na sua formatação. Esta tabela é mais um exemplo disso. Essas incongruências têm que ser tratadas antes de se proceder à análise de dados. Repare nas colunas **F**, **A** e **P**. Qual o problema que identifica?

   Depois de ter identificado o problema, reconhecerá a utilidade do seguinte exemplo para o resolver.

```
> x = c("aaa", "bb.b", "cc,c")
> x = gsub("\\.", "", x)
```

   Nota: É possível resolver este exercício e o anterior numa só vez.

> **Solução:**
>
> A maior parte dos números (mas não todos!) usa o ponto como separador dos milhares. É necessário remover o ponto, para que os números sejam correctamente identificados em **R**, caso contrário, poderão vir a ser reconhecidos como números decimais. A segunda coluna é a única que deverá manter-se como texto, pois contém o nome dos clubes.
>
> ```
> str(liga)
> liga[-2] = apply(liga[-2], 2, function(coluna) as.numeric(gsub("\\.", "", coluna)))
> ```

6. Concluída a preparação dos dados, há que passar à análise. Quais são as equipas com mais derrotas do que a média?

> **Solução:**
>
> ```
> subset(liga, L > mean(L))
> ```

7. Acrescente uma nova coluna com a média de golos por jogo de cada equipa.

> **Solução:**
>
> ```
> liga = transform(liga, média_golos = F / G)
> ```

8. Como acrescentou uma coluna ao conjunto de dados, já vale a pena guardar os dados num ficheiro. Guarde-os num novo ficheiro com a seguinte especificação:

   Nome PrimeiraLiga_25anos.dat

Separador tabulação

Separador decimal vírgula

Valores em falta (NA) hífen

Nome das colunas sim

Nome das linhas não

Solução:

```
nome = 'Primeira_Liga_25anos.data'
write.table(file=nome, sep='\t', dec=',', na='-', col.names=TRUE, row.names=FALSE)
```

9. Nem sempre um maior número de pontos corresponde a um maior número de vitórias. Identifique as equipas em que a sua posição na tabela (está ordenada por pontos) não corresponde à posição se a tabela estivesse ordenada por vitórias.

Solução:

```
liga$Club[order(liga$W, decreasing=TRUE) != liga$No.]
```

10. Qual é a equipa que tem mais golos marcados do que sofridos, e que simultaneamente tem menos pontos?

    Sugestão: Acrescente primeiro uma coluna que diz se a equipa tem ou não mais golos marcados do que sofridos.

Solução:

```
liga = transform(liga, positivo = F - A > 0)
liga$Club[max(which(liga$positivo))]
```

11. Haverá uma relação entre o número de golos e o número de temporadas, isto é, se, em média, as equipas com mais experiência na *Primeira Liga* tendem a marcar mais golos?

    Sugestão: Determine primeiro qual a média de golos marcados por cada valor de experiência (número de épocas jogadas).

Solução:

```
média_época = aggregate(liga$G, list(liga$S), mean)
média_época = transform(média_época, média = x / Group.1)
> média_época
Group.1         x        m
1       1  32.00000 32.00000
2       2  66.00000 33.00000
3       3  99.71429 33.23810
4       4 136.00000 34.00000
5       5 163.33333 32.66667
```

```
6        6 192.00000 32.00000
7        7 238.00000 34.00000
8        8 264.00000 33.00000
9       11 374.00000 34.00000
10      13 430.00000 33.07692
11      14 452.00000 32.28571
12      15 486.00000 32.40000
13      16 518.00000 32.37500
14      17 566.00000 33.29412
15      18 651.00000 36.16667
16      19 638.00000 33.57895
17      20 654.00000 32.70000
18      24 788.00000 32.83333
19      25 818.00000 32.72000
```

Olhando para os valores, não parece haver uma relação entre o número de épocas jogadas e a média de golos marcados.

Este facto pode ser confirmado com uma regressão linear (esta parte não era pedida no exercício, nem faz parte da matéria da disciplina, mas poderá ser útil aos alunos no futuro).

```
coef(lm(média_época$m ~ média_época$Group.1))
        (Intercept) média_época$Group.1
        32.95632059          0.01388322
```

Essas duas variáveis têm um coeficiente de correlação muito baixo, cerca de 1,4%.

12. Normalmente os clubes são mais conhecidos por uma só palavra. Por exemplo, o clube *SC de Braga* é conhecida por *Braga*, e o clube *A Naval 1º de Maio* é mais conhecido apenas por *Naval*.

    Pretende-se alterar o nome dos clubes para a denominação mais comum. De modo a não alterar um a um manualmente, considere o seguinte algoritmo (não é perfeito, mas funciona bem na maior parte dos casos):

    1. Separar o nome por palavras;
    2. Escolher a palavra com maior número de caracteres;
    3. Usar essa palavra como nome para a equipa.

    Por exemplo, o clube com o nome `"CF Os Belenenses"` pode ser separado num vector com 3 palavras:

    ```
    [1] "CF"         "Os"         "Belenenses"
    ```

    Estas palavras têm os seguintes números de caracteres:

    ```
    [1]  2  2 10
    ```

    Neste vector, o valor mais alto está na posição:

    ```
    [1] 3
    ```

    Então, voltando ao vector com as palavras separadas, escolhe-se a palavra nessa posição:

    ```
    [1] "Belenenses"
    ```

Nota: Neste exercício, há uma função que pode ser particularmente útil, caso utilize listas como um resultado intermediário: `unlist()`. Esta função simplifica uma lista para a forma de um vector.

Solução:

```
palavra_mais_longa = function(nome) {
        palavras = unlist(strsplit(nome, " "))
        mais_longa = which.max(nchar(palavras))
        return (palavras[mais_longa])
}

sapply(liga$Club, palavra_mais_longa)
```

```
            FC Porto            SL Benfica           Sporting CP
            "Porto"             "Benfica"            "Sporting"
            SC de Braga  Vitória SC Guimarães   CS Marítimo Madeira
            "Braga"           "Guimarães"            "Marítimo"
            Boavista FC      CF Os Belenenses  FC Paços de Ferreira
            "Boavista"          "Belenenses"            "Ferreira"
            Vitória FC Setúbal          UD de Leiria        Gil Vicente FC
            "Vitória"              "Leiria"             "Vicente"
            CD Nacional Madeira            Rio Ave FC   A Académica Coimbra
            "Nacional"                  "Rio"           "Académica"
            SC Beira Mar CF Estrela da Amadora            SC Farense
            "Beira"             "Estrela"             "Farense"
            SC Salgueiros 08      GD Estoril Praia          GD de Chaves
            "Salgueiros"             "Estoril"             "Chaves"
            Moreirense FC    A Naval 1º de Maio         FC de Alverca
            "Moreirense"               "Naval"            "Alverca"
            SC Campomaiorense          SC Olhanense   CF União da Madeira
            "Campomaiorense"           "Olhanense"             "Madeira"
            FC de Tirsense             FC Arouca           FC Penafiel
            "Tirsense"              "Arouca"            "Penafiel"
            Leça FC          FC Famalicão        CD Santa Clara
            "Leça"           "Famalicão"               "Santa"
            Varzim SC            Leixões SC         SC de Espinho
            "Varzim"             "Leixões"            "Espinho"
            CD Aves         SCU Torreense         FC Felgueiras
            "Aves"           "Torreense"          "Felgueiras"
            CD Tondela       Portimonense SC           CD Feirense
            "Tondela"        "Portimonense"            "Feirense"
            CD Trofense
            "Trofense"
```