Ejercicios extra (bucles)

David García-Callejas

1

Usando el dataset "iris":

a) almacena en un vector el valor medio de cada columna

b) almacena en otro vector el número de valores únicos de cada columna pista: usa la función n_distinct del paquete tidyverse

```
# cargamos el paquete tidyverse
library(tidyverse)

# creamos los vectores de resultados
vector.medias <- numeric(length = length(iris))
vector.unicos <- numeric(length = length(iris))

# bucle que vaya columna a columna
for(i in 1:length(iris)){

  # calculamos la media sólo si la columna i es numérica
  if(class(iris[,i]) == "numeric"){
    vector.medias[i] <- mean(iris[,i])
  }
  # la función n_distinct es válida también con caracteres
  vector.unicos[i] <- n_distinct(iris[,i])
}

# alternativa con la familia apply
# sapply aplica una función a cada columna de un dataframe
medias.2 <- sapply(iris,mean)
unicos.2 <- sapply(iris,n_distinct)
```

2

Usando el dataset "sample500tuits_starwarsandaluz.txt": a) crea un dataframe con dos columnas: num.caracteres y num.palabras, que almacene el número de caracteres y el número de palabras de cada tuit

b) muestra por pantalla el número medio de caracteres y palabras de los datos

c) ¿cómo se harían estas operaciones sin usar bucles?

```
# cargamos el paquete stringr
library(stringr)

# leemos el archivo
```

```r
my.file <- "../data/sample500tuits_starwarsandaluz.txt"
conn <- file(my.file,open = "r")
lineas <-readLines(conn)

# creamos el dataframe de resultados
# ya con la longitud adecuada (tantas filas como tuits hay en el vector lineas)
resultados <- data.frame(num.caracteres = numeric(length(lineas)),
                         num.palabras = numeric(length(lineas)))

# vamos fila a fila calculando las dos columnas
# la función para contar el número de palabras viene de stackoverflow
# https://stackoverflow.com/questions/8920145/
# count-the-number-of-all-words-in-a-string
for(i in 1:length(lineas)){
  resultados$num.caracteres[i] <- nchar(lineas[i])
  resultados$num.palabras[i] <- str_count(lineas[i], '\\w+')
}

# mostramos los valores medios en pantalla
cat("en el archivo: ",my.file, "\nhay",length(lineas),
    "tuits, con una media por tuit de\n",
    mean(resultados$num.caracteres),
    "caracteres\n",mean(resultados$num.palabras),"palabras")
```

```
## en el archivo:  ../data/sample500tuits_starwarsandaluz.txt
## hay 500 tuits, con una media por tuit de
##  105.522 caracteres
##  16.07 palabras
```

```r
# estas operaciones se pueden hacer en una sola linea,
# aprovechando la vectorización de R
caracteres2 <- nchar(lineas)
palabras2 <- str_count(lineas,'\\w+')
```

3

Crea un script que, usando la carpeta "data":

a) muestre por pantalla todos los archivos con extensión .csv que hay en esa carpeta

b) lea los archivos cuyo nombre contenga "starwars_personajes", y almacene en un dataframe tres campos:

- el nombre y ruta del archivo
- el número de filas
- el número de columnas

```r
# ruta a la carpeta que quiero evaluar
mi.carpeta <- "/home/david/Work/Projects/R_courses/CEA2020_Intro/data/"

# recupero los archivos que hay en esa carpeta
mis.archivos <- list.files(mi.carpeta)

# los muestro por pantalla
cat("en la carpeta:",mi.carpeta,
    "\nhay",length(mis.archivos),"archivos:\n")
```

```
## en la carpeta: /home/david/Work/Projects/R_courses/CEA2020_Intro/data/
## hay 14 archivos:
```

```
cat(mis.archivos,sep = "\n")
```

```
## Earthquake_data.csv
## earthquake_metadata.txt
## Earthquake_wide_example.csv
## hojas_excel.xlsx
## proyecto
## sample20000tuits_madrid16a.txt
## sample500tuits_starwarsandaluz.txt
## starwars_info_personajes.csv
## starwars_names_coma.csv
## starwars_names.csv
## starwars_personajes_naves.csv
## starwars_personajes_peliculas.csv
## starwars_personajes_vehiculos.csv
## terrestrial_protected_areas.xlsx
```

```
# selecciono los archivos que me interesan
archivos.sw <- mis.archivos[grep("starwars_personajes",mis.archivos)]

# creo el dataframe de resultados,
# con un número de filas igual al
# número de archivos que tengo (length(archivos.sw))
resultados.archivos <- data.frame(ruta = character(length(archivos.sw)),
                                  nombre = character(length(archivos.sw)),
                                  num.filas = numeric(length(archivos.sw)),
                                  num.columnas = numeric(length(archivos.sw)),
                                  stringsAsFactors = FALSE)

# leo archivo a archivo
# tened en cuenta que repetir la misma operación para abrir archivos
# solo funciona en todos si todos tienen la misma estructura,
# es decir, el mismo caracter para separar columnas (;) y punto decimal
# por eso este ejemplo está restringido a estos tres archivos solamente.
for(i in 1:length(archivos.sw)){
  # leo el archivo pasando al argumento "file" la ruta completa:carpeta+nombre
  mi.archivo <- read.csv2(file = paste(mi.carpeta,archivos.sw[i],sep = ""),
                          stringsAsFactors = FALSE)

  # almaceno los resultados en el dataframe directamente
  # la carpeta origen es siempre la misma
  resultados.archivos$ruta[i] <- mi.carpeta
  # nombre
  resultados.archivos$nombre[i] <- archivos.sw[i]
  # num filas
  resultados.archivos$num.filas[i] <- nrow(mi.archivo)
  # num columnas
  resultados.archivos$num.columnas[i] <- length(mi.archivo)
}
```