readr

Entornos de Análisis de Datos: R

Alberto Torres Barrán

2021-01-07

Introducción

- Paquete para importar y exportar ficheros de texto
- Importar datos:
  - `read_csv()`, para ficheros CSV
  - `read_csv2()`, para ficheros CSV separados por ";"
  - `read_delim()`, para ficheros ASCII delimitados por otros caracteres distintos de "," y ";"
  - `read_tsv()`, para ficheros ASCII delimitados por tabuladores
  - `read_table()`, para ficheros ASCII delimitados por espacios
- Exportar datos: `write_csv()`. `write_csv2()`, etc.
- Más información: *vignette* readr

Ejemplo

```
write_csv(mpg, "mpg.csv")
mpg1 <- read_csv("mpg.csv")
head(mpg1)
## # A tibble: 6 x 11
##   manufacturer model displ  year   cyl trans drv     cty   hwy fl
##   <chr>        <chr> <dbl> <dbl> <dbl> <chr> <chr> <dbl> <dbl> <chr>
## 1 audi         a4      1.8  1999     4 auto~ f        18    29 p
## 2 audi         a4      1.8  1999     4 manu~ f        21    29 p
## 3 audi         a4      2    2008     4 manu~ f        20    31 p
## 4 audi         a4      2    2008     4 auto~ f        21    30 p
## 5 audi         a4      2.8  1999     6 auto~ f        16    26 p
## 6 audi         a4      2.8  1999     6 manu~ f        18    26 p
## # ... with 1 more variable: class <chr>
```

Directorio de trabajo

- Directorio donde apunta RStudio

```
getwd()
## [1] "C:/Users/alberto/Desktop/curso-uah-eadr/src"
```

- Se puede cambiar con `setwd()`
- Normalmente es el directorio que aparece en la pestaña `Files` de RStudio
    1. Se puede cambiar de directorio haciendo click en `...`
    2. Se puede asignar ese nuevo directorio como directorio de trabajo con `More > Set as working directory`
- Directorio por defecto donde se buscan los ficheros a importar
- Alternativamente, podemos especificar el la ruta completa o la ruta relativa:
    1. `.` hace referencia al directorio actual
    2. `..` hace referencia al directorio anterior

Parámetros opcionales

- `col_names`, si TRUE, la primera fila es el nombre de las variables. También se le puede pasar un vector de cadenas de caracteres con los nombres
- `delim`, carácter que separa las columnas (solo en `read_delim()`)
- `na`, vector con cadenas que se interpretan como missing values. Por defecto `NA` y la cadena vacía
- `col_types`, vector de clases para las columnas (ver documentación de `col()`). Por defecto se intenta adivinar el tipo de cada columna a partir de las 1000 primeras líneas.
- `n_max`, número máximo de líneas a leer del fichero
- `skip`, número de líneas a ignorar al princpio del fichero
- `locale`, parámetro que nos permite cambiar el enconding, separador decimal y formato de fechas (ver documentación de `locale()`)

Funciones de R base

- Es muy importante no confundir las funciones de R base `read.csv`, `read.csv2`, `read.table`, etc., con su equivalente de `readr`
- Similitudes:
    1. mismo nombre, pero cambiando la `_` por `.`
    2. realizan la misma operación básica, importar un fichero de texto en R
- Diferencias:
    1. nombre de los parámetros opcionales
    2. tipo de retorno (`data.frame` vs `tibble`)
    3. funciones de `readr` son más rápidas
- Se recomienda usar las funciones de la librería `readr`

Ejemplos accidentes bicicletas Madrid

Fuente

```
accidentes <- read_csv2('../data/AccidentesBicicletas_2017.csv',
                        skip = 2, locale = locale(encoding = 'latin1'))
```

```
head(accidentes, 4)
## # A tibble: 4 x 8
##   Fecha `TRAMO HORARIO` `Nm Tot Victima~ DISTRITO Lugar Numero
##   <chr> <chr>                     <dbl> <chr>    <chr>  <dbl>
## 1 01/0~ DE 6:00 A 6:59                1 ARGANZU~ CALL~    120
## 2 02/0~ DE 21:00 A 21:~               1 SAN BLAS CALL~     14
## 3 03/0~ DE 19:00 A 19:~               1 CENTRO   CALL~      8
## 4 04/0~ DE 21:00 A 21:~               1 CENTRO   CALL~     13
## # ... with 2 more variables: `Tipo Accidente` <chr>, `Tipo
## #   Vehiculo` <chr>
```

Ejemplos jugadores mundial

Fuente

```
worldcup <- read_tsv('../data/worldcupplayerinfo_20140701.tsv')
```

```
worldcup <- read_tsv('../data/worldcupplayerinfo_20140701.tsv',
                     col_types = list("X11" = col_skip(), "X12" = col_skip()))

worldcup
## # A tibble: 736 x 10
##    Group Country  Rank Jersey Position   Age Selections Club  Player
##    <chr> <chr>   <dbl>  <dbl> <chr>    <dbl>      <dbl> <chr> <chr>
##  1 A     Brazil      3      1 Goalie      31          9 Bota~ Jeffe~
##  2 A     Brazil      3     12 Goalie      34         80 Toro~ Julio~
##  3 A     Brazil      3     22 Goalie      31          6 Atle~ Victor
##  4 A     Brazil      3      2 Defender    31         75 Barc~ Dani ~
##  5 A     Brazil      3     13 Defender    30         12 Baye~ Dante
##  6 A     Brazil      3      4 Defender    27         36 Chel~ David~
##  7 A     Brazil      3     15 Defender    27          5 Napo~ Henri~
##  8 A     Brazil      3     23 Defender    32         72 Roma  Maicon
##  9 A     Brazil      3      6 Defender    26         31 Real~ Marce~
## 10 A     Brazil      3     14 Defender    32          9 Pari~ Maxwe~
## # ... with 726 more rows, and 1 more variable: Captain <chr>
```

Ejemplo espacios

Fuente (ligeramente modificado)

```
massey <- read_table('../data/massey-rating.txt')
```

```
massey
## # A tibble: 10 x 11
##      UCC   PAY   LAZ   KPK RT      COF   BIH   DII ACU   Team      Conf
##    <dbl> <dbl> <dbl> <dbl> <chr> <dbl> <dbl> <dbl> <chr> <chr>     <chr>
##  1     1     1     1     1 1         1     1     1 1     Ohio St   B10
##  2     2     2     2     2 2         2     2     2 2     Oregon    P12
##  3     3     4     3     4 -         4     3     4 3     Alabama   SEC
##  4     4     3     4     3 4         3     5     3 4     TCU       B12
##  5     6     6     6     5 5         7     6     5 -     Michigan~ B10
##  6     7     7     7     6 7         6    11     8 8     Georgia   SEC
##  7     5     5     5     7 -         8     4     6 5     Florida ~ ACC
##  8     8     8     9     9 10        5     7     7 7     Baylor    B12
##  9     9    11     8    13 11       11    12     9 -     Georgia ~ ACC
## 10    13    10    13    11 8        9     10    11 10    Mississi~ SEC
```

```
massey <- read_table('../data/massey-rating.txt', na = c("-"))
```

```
massey
## # A tibble: 10 x 11
##      UCC   PAY   LAZ   KPK    RT   COF   BIH   DII   ACU Team      Conf
##    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <chr>     <chr>
##  1     1     1     1     1     1     1     1     1     1 Ohio St   B10
##  2     2     2     2     2     2     2     2     2     2 Oregon    P12
##  3     3     4     3     4    NA     4     3     4     3 Alabama   SEC
##  4     4     3     4     3     4     3     5     3     4 TCU       B12
##  5     6     6     6     5     5     7     6     5    NA Michigan~ B10
##  6     7     7     7     6     7     6    11     8     8 Georgia   SEC
##  7     5     5     5     7    NA     8     4     6     5 Florida ~ ACC
##  8     8     8     9     9    10     5     7     7     7 Baylor    B12
##  9     9    11     8    13    11    11    12     9    NA Georgia ~ ACC
## 10    13    10    13    11     8     9    10    11    10 Mississi~ SEC
```

Libreria readxl

- Fichero de ejemplo que viene con la librería

```
library(readxl)
excel_ex <- readxl_example("datasets.xlsx")
excel_ex
## [1] "C:/Users/alberto/Documents/R/win-library/4.0/readxl/extdata/datasets.x
```

- Podemos listar las hojas de un fichero Excel:

```
excel_sheets(excel_ex)
## [1] "iris"     "mtcars"   "chickwts" "quakes"
```

- Leer como tibble/dataframe:

```
cars <- read_excel(excel_ex, sheet = "mtcars")
head(cars)
## # A tibble: 6 x 11
##     mpg   cyl  disp    hp  drat    wt  qsec    vs    am  gear  carb
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1  21       6   160   110  3.9   2.62  16.5     0     1     4     4
## 2  21       6   160   110  3.9   2.88  17.0     0     1     4     4
## 3  22.8     4   108    93  3.85  2.32  18.6     1     1     4     1
## 4  21.4     6   258   110  3.08  3.22  19.4     1     0     3     1
## 5  18.7     8   360   175  3.15  3.44  17.0     0     0     3     2
## 6  18.1     6   225   105  2.76  3.46  20.2     1     0     3     1
```

Parámetros útiles

- `range`: rango de celdas a importar, en lugar de la hoja completa (por ejemplo: "C3:F14")
- `sheet`: número o nombre de la hoja a leer. Por defecto la primera
- `col_names`: `TRUE` si la primera fila contiene los nombres de las columnas
- `na`: vector con cadenas que se interpretan como missing values. Por defecto celdas vacias
- `col_types`: tipo de cada columna. Por defecto se intenta inferir de los datos. Posibles valores: "skip", "guess", "logical", "numeric", "date", "text" or "list"

Otros formatos

- `readr` y `readxl` solo tienen funciones para importar ficheros de texto/Excel
- Para otros formatos, existen librerías específicas:
  - `haven`, para ficheros de SPSS, Stata y SaS
  - `DBI` junto con otro paquete específico dependiendo de la BD (`RMySQL`, `RSQLite`, etc.) nos permite hacer *querys* contra una BD
  - `jsonlite`, para ficheros JSON
  - `xml2`, para ficheros XML