Cargar y guardar datos en R

Christian Salas Eljatib, Ph.D.

E-mail: cseljatib@gmail.com

Web: https://eljatib.com

4 de enero de 2023
Santiago, Chile

Contenidos

1. Orden primero
   - ¿Dónde estamos trabajando en el PC?
2. Cargando datos
   - Desde una librería R
   - Desde un archivo físico
   - Desde un archivo ASCII
   - Desde un archivo Dbase (.dbf)
   - Desde un archivo en Internet
3. Guardando datos
   - Como un archivo con valores separados con coma
   - Como un archivo ASCII

Ubiquémonos

Aclaración sobre archivos

Todos los archivos de datos a ocupar en estos apuntes están disponibles en `https://eljatib.com/rlibro`.

¿Dónde estamos trabajando?

```
> getwd()
> list.files() # lista de archivos en el directorio
```

```
> getwd()
[1] "C:/Documents and Settings/Administrator/My Documents"
```

Definiendo un nuevo directorio de trabajo para la sesión

```
> setwd("c:/miCarpeta")
```

Desde una librería R

```
> library(datana)

> data(package = "election")

> data(election)

> help(election)

> head(election)
    gore   bush buchanan
1  47300  34062      262
2   2392   5610       73
3  18850  38637      248
4   3072   5413       65
5  97318 115185      570
6 386518 177279      789
```

Desde un archivo con valores separados con coma

```
> ufc.data <- read.csv("ufcData.csv", header = T)
```

```
> tail(ufc.data) #muestra ultimas seis observaciones
     Plot Tree Species Dbh Height
367   143    1      GF 280    210
368   143    2      GF 330    205
369   143    7      WC 478    205
370   144    1      GF 102    160
371   144    2      DF 315    220
372   144    4      WL 265    250
```

¿Y si quiero especificar una ruta diferente?

```
> otro.data <- read.csv('F:/cursoR/ufcData.csv',header=T)
```

Desde un archivo ASCII

Datos en ASCII (.dat), con descripción en las primeras 11 líneas

```
> euca.data <- read.table('eucaLeaf.dat', header = F,
  skip=11,
  col.names=c('time','tree','shoot','l','w','la','ignore'))
> head(euca.data)
   time tree shoot   l  w    la  ignore
1 EARLY   11    SC 125 32  28.8  30.050
2 EARLY   11    SC 300 50 106.7 103.887
3 EARLY   11    SC 300 43  86.5  88.705
4 EARLY   11    SC 225 35  61.4  55.462
5 EARLY   11    SC 228 37  68.2  59.477
6 EARLY   11    SC 120 24  22.2  21.445
```

Desde un archivo Dbase (extensión .dbf)

(Note que deberá instalar antes el paquete `foreign`)

```
> require(foreign)
> snaspe.dbf <- read.dbf('snaspe09.dbf')
> head(snaspe.dbf)
                          CAPTION      AREA PERIMETER HECTARES
1                       Herquehue 118142180     49692    11814
2      Parque Nacional Conguillo 621338732    110002    62134
3  Reserva Nacional China Muerta 121467905     49579    12147
4        Reserva Nacional Nalcas 312660543     77141    31266
5 Reserva Nacional Malalcahuello 167554383     55839    16755
6    Reserva Nacional Villarrica 199860156     98837    19986
```

Desde un archivo en Internet

Datos de COVID19 en repositorios del Ministerio de Ciencias de Chile

El repositorio es https://raw.githubusercontent.com/MinCiencia/Datos-COVID19/master/output/producto2 y el archivo ahi alojado denominado 2020-03-30-CasosConfirmados.csv

```
> url.origen<-"https://raw.githubusercontent.com/MinCiencia/Datos-
COVID19/master/output/producto2/2020-03-30-CasosConfirmados.csv"

> mis.datosInternet<-read.csv(url(url.origen),header=T)

> head(mis.datosInternet,5)
             Region Codigo.region                          Comuna Codigo.comuna
1 Arica y Parinacota            15                           Arica         15101
2 Arica y Parinacota            15                       Camarones         15102
3 Arica y Parinacota            15                  General Lagos         15202
4 Arica y Parinacota            15                          Putre         15201
5 Arica y Parinacota            15 Desconocido Arica y Parinacota            NA
  Poblacion Casos.Confirmados
1    247552                 6
2      1233                 0
3       810                 0
4      2515                 0
5        NA                NA
```

Guardando datos desde R a un archivo físico

Supongamos quieremos guardar una dataframe como la siguiente

```
> x1<-c(1,4,5,0,3)
> x2<-c(10,-4,35,1,3)
> g.x<-c('A0','A1','A2','A3','A4')
> df.yo <- data.frame(x1,x2,g.x)
> df.yo
  x1 x2 g.x
1  1 10  A0
2  4 -4  A1
3  5 35  A2
4  0  1  A3
5  3  3  A4
```

Como un archivo con valores separados con coma (.csv)

Para guardar una dataframe como un archivo

- Escoger el formato del archivo a ser creado (asumamos es "csv")
- Necesitamos conocer el nombre de la dataframe, y el nombre del archivo físico donde guardaremos dicha dataframe.
- Supongamos que guardaremos la dataframe `mi.dataframe` en el archivo físico denominado `miprimerCsv.csv`

Función `write.csv()`

```
> write.csv(df.yo, file = "miPrimerCsvR.csv")
```

- El archivo `miPrimerCsvR.csv` ha sido guardado en la misma carpeta donde estamos trabajando.
- Recuerde que para saber cual es su carpeta actual de trabajo en una sesión en R puedes emplear la función `getwd()`

Como un archivo csv, pero definiendo una nueva opción

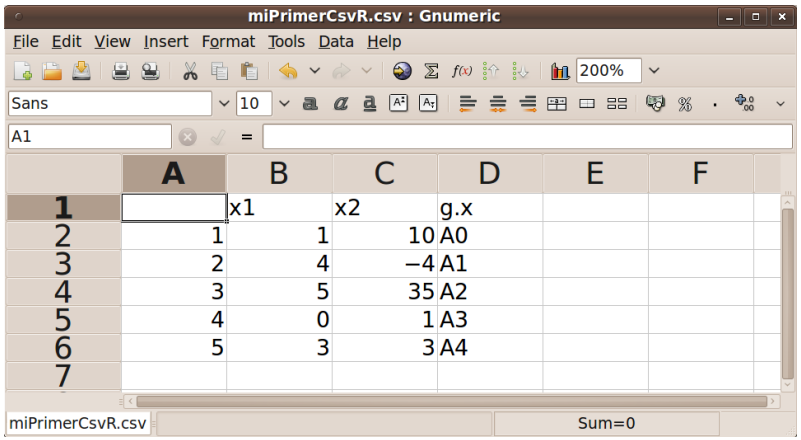
Función `write.csv()`, pero definiendo opción `row.names`

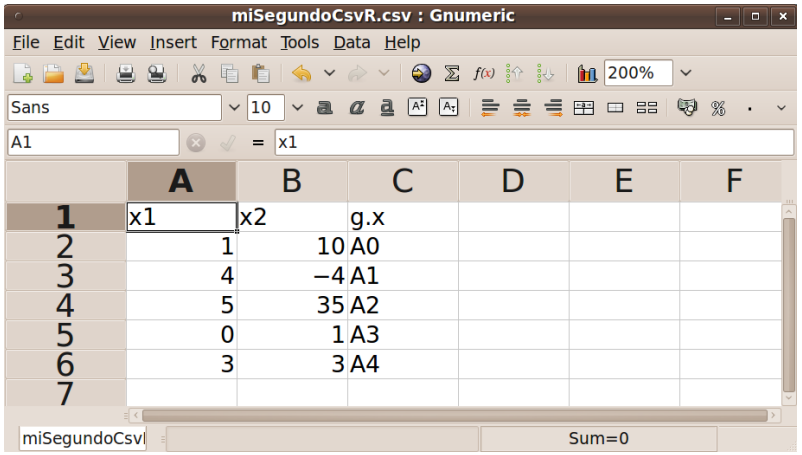
antes usamos

```
> write.csv(df.yo,
 file = "miPrimerCsvR.csv")
```

y compare con

```
> write.csv(df.yo,
            file = "miSegundoCsvR.csv", row.names=FALSE)
```





Como un archivo ASCII

Para guardar una dataframe como un archivo

- Debemos escoger el formato del archivo a ser creado (asumimos ahora ".dat")
- Necesitamos conocer el nombre de la dataframe, y el nombre del archivo físico donde guardaremos dicha dataframe.
- Supongamos que guardaremos la dataframe `mi.dataframe` en el archivo físico denominado `miprimerDat.dat`

Función `write.table()`

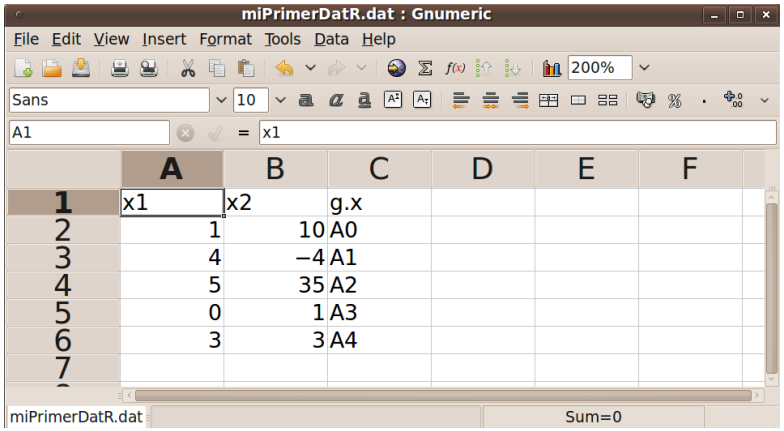
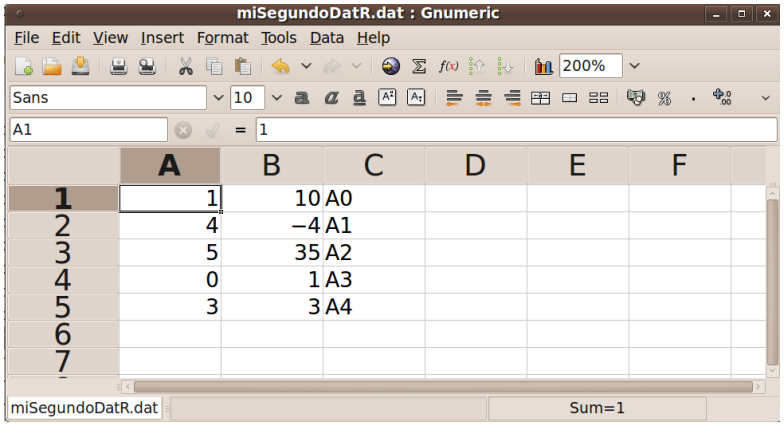
```
> write.table(df.yo, file = "miPrimerDatR.dat",
              quote = FALSE, row.names = FALSE)
```

- El archivo `miPrimerDatR.dat` ha sido guardado en la misma carpeta donde estamos trabajando.

Como un archivo ASCII, pero definiendo una nueva opción

Función `write.table()`, pero definiendo opción `row.names`

```
> write.table(df.yo, file = "miPrimerDatR.dat",
              quote = FALSE, row.names = FALSE)
```

y compare con

```
> write.table(df.yo,
              file = "miSegundoDatR.dat", quote = FALSE,
              row.names = FALSE, col.names = FALSE)
```