Curso: Estadística

Prof. José Manuel Magallanes, PhD

Leyendo Data desde R

El primer paso para realizar cualquier tipo de análisis es tener los datos. Esto condiciona además el diseño de la investigación.

Los datos pueden estar organizados por terceros. Si es así, debemos poder leer el formato en que estén preparados. A veces los necesitamos como están, a veces hay que reformatearlos.

Otras veces, uno mismo recogerá los datos. Si los datos representan alguna población en su totalidad no tendremos data sesgada; si no, habrá que hechar mano de algun proceso de muestreo. Veamos los siguientes casos:

1. Propietary software.
2. Recolección Ad-hoc.
3. Uso de APIs.
4. "Scraping" tablas de datos.

Tener en claro la ubicación de los archivos es muy importante. Crear una carpeta para los datos en nuestra máquina es la alternativa clásica, pero si es posible usar el link hacia algun repositorio puede ser más eficiente. Si los archivos están en la máquina, los siguientes comandos de R son de utilidad:

```
getwd() # dónde estoy?
```

Acabas de ver donde estás, si quieres cambiar debes utilizar:

```
setwd()
```

La ubicación de la carpeta que necesitas debe estar en los parentesis.

Windows y Mac no describen las rutas de los directorios de la misma manera, por lo que es mejor usar el comando **file.path()**:

```
folder="data"
fileName="anes_timeseries_2012.sav"
fileToRead=file.path(folder,fileName)
```

En *fileToRead* estará la ruta correcta. Si hubiera una mayor secuencia de folders, **file.path** los concatenará sin problema.

Data de "proprietary software"

*Leyendo SPSS:

Abramos este archivo desde el *American National Election Studies Survey* (ANES):

```
#deben instalar el paquete "haven" primero! (se toma su tiempo!)
library(foreign)
folder="data"
fileName="anes_timeseries_2012.sav"
```

```
fileToRead=file.path(folder,fileName)
dataSpss=read.spss(fileToRead, to.data.frame=T, use.value.labels = F)
```

En **dataSpss** está un gran archivo. Además no es aun una tabla de datos (dataframe). Creemos un dataframe con dos variables ("libcpre_self","libcpo_self"), un par de preguntas pre y post eleccions donde se solicita ubicarse en algun valor de 7 puntos, cuyos extremos son "extremandamente liberal" y "extremadamente conservador".

```
varsOfInterest=c("libcpre_self","libcpo_self")
dataSpssSub=dataSpss[varsOfInterest] #convertimos en dataframe y seleccionamos variables de interés
head(dataSpssSub)
```

El archivo anterior en formato STATA:

```
fileName="anes_timeseries_2012.dta"
fileToRead=file.path(folder,fileName)
dataStata=read_dta(fileToRead)
dataStataSub=as.data.frame(dataStata)[varsOfInterest]
head(dataStataSub)
```

Otro formato propietario de mucho uso es el de las hojas de cálculo en Excel:

```
library(readxl) #instalen el paquete!
fileName="idhPeru.xlsx"
fileToRead=file.path(folder,fileName)
dataExcel=read_excel(fileToRead)
head(dataExcel)
```

Al ejecutar el comando anterior, verás que la data se carga pero necesita algo de organización (así la prepara Nacionaes Unidas. . . ni modo).

Go to page beginning

Recolección ad-hoc

Podemos muchas veces usar los formularios de Google Docs para recoger información. Por ejemplo, visite este link.

Luego, hay que producir un URL para la data creada.

```
require(RCurl) #instalen paquete!
# link obtenido de google docs:
link='https://docs.google.com/spreadsheets/d/1bDMM5s3PDC5awrSkILFRPJM1Q0j95TtVxErvvNNOHPU/pub?output=csv

# obtener info:
myCsv = getURL(link)
# cambiar nombre de columnas:
namesOfCols=c('timeStamp','nombre','apellido','tipoCole', 'lugarNace','edad','sexo', 'religion')

# formato csv:
myData=read.csv(textConnection(myCsv),col.names=namesOfCols)

head(myData) #veamos primeras filas
```

Go to page beginning

Uso de los APIs

Hay organizaciones que tienen una política de datos abiertos, por lo que ofrecen mecanismos para acceder a sus datos. Los formatos son por lo general XML o JSON. Traigamos la data producida por el servicio '9-1-1' de la Policia de Seattle, Washington:

```
library(jsonlite) #instalen paquete!
endPoint="https://data.seattle.gov/resource/pu5n-trf4.json"
data911 = fromJSON(endPoint)
head(data911)
```

Go to page beginning

"Scraping" tablas de datos

Aqui descargaremos los datos de esta wikipage

```
# instalen paquetes antes de activarlos!
library(XML)
library(RCurl)

# URL
wiki="https://en.wikipedia.org/wiki/"
link = "List_of_freedom_indices"

# Data
wikiLinkContents = getURL(paste0(wiki,link))
wikiTables = readHTMLTable(wikiLinkContents,
                           stringsAsFactors=FALSE)
```

Veamos que tenemos:

```
#data frame:
is.data.frame(wikiTables) #es un dataframe?
#list:
is.list(wikiTables) #es una lista?
# how many?
length(wikiTables) #cuántos elementos?
```

Al visitar la web, nos damos cuenta la tabla de interés es la segunda:

```
idx=wikiTables[2]
str(idx)
```

Una breve mirada:

```
head(idx)
```

Vemos que necesitamos hacer limpieza de datos!

Go to page beginning