Geeting and Cleaning Data - Week 1 Notes

Carmelo Ramirez

16/10/2020

Raw and Processed Data

- *Data*: Data are values of qualitative or quantitative vairables, belonging to a set of items.

Raw Data

- The original source of data
- Often hard to use for data analyses
- Data analysis includes processing
- Raw data may only need to be processed once

Processed Data

- Data that is ready for analysis
- Processing can include merging, subsetting, transforming, etc.
- There may be standards for processing
- All steps should be recorded

The Components of Tidy Data

The tidy data:

1. Each variable you measure should be in one column
2. Each different observation of that variable should be in a different row
3. There should be one table for each "kind" of variable
4. If you have multiple tables, they should include a column in the table that allows them to be linked

Some important tips:

- Include a row at the top of each file with variable names
- Make variable names human readable AgeAtDiagnosis insted of AgeDx
- In general data should be saved in one file per table.

The Code Book

Should include:

1. Information about the variables (with units) in the data set not contained in the tidy data

2. information about the summary choices you made
3. Information about the experimental study design you used

Some other important tips:

- A common format for this document is a Word/text file
- There should be a section called "Study Design" that has a thorough description of how you collected the data.
- There must be a section called "Code Book" that describes each variable and its units.

Downloading files

Checking for and creating directories

- *files.exists("directoryName")* will check to see if the directory exists.
- *dir.create("directoryName")* will create a directory if it doesn't exist

Download a file from the web

```
fileURL <- "https://data.baltimorecity.gov/api/views/dz54-2aru/rows.csv?accessType=DOWNLOAD"
download.file(fileURL, destfile = "./data/cameras.csv", method = "curl")
list.files("./data")
```

Reading Local Flat Files

read.table()

- This is the main function for reading data into R
- Flexible and robust but requires more parameters
- Reads data into RAM - big data can cause problems
- Important parameters *file*, *header*, *sep*, *row.names*, *nrows*
- Related: *read.csv()*, *read.csv2()*

Reading Excel Files

```
library(xlsx)

cameraData <- read.xlsx("./data/cameras.xlsx", sheetIndex = 1, header = TRUE)
```

Reading XML

- Extensible Markup Language
- Frequently used to store structured data
- Particularly widely used in internet applications
- Extracting XML is the basis for most web scrapping
- Components
  - Markup

– Contents

```
library(XML)
fileURL <- "http://www.w3schools.com/xml/simple.xml"
doc <- xmlTreeParse(fileURL, userInternal = FALSE)
rootNode <- xmlRoot(doc)
xmlName(rootNode)
```

Reading JSON

```
library(jsonlite)
jsonData <- fromJSON("https://api.github.com/users/jtleek/repos")
names(jsonData)
names(jsonData$owner$login)

## Writing Data to JSON
myjson <- toJSON(iris, pretty = TRUE)
```

The data.table Package

- Inherets from data.frame
- Written in C so it is much faster
- Much, much faster at subsetting, group, and updating