Getting and Cleaning Data

# Raw data -> Processing Script -> tidy data -> data analysis -> data communication

# Four Files to have:

1. The raw data
2. A tidy data set
3. A code book describing each variable and its values in the tidy data set
4. An exact recipe how you do these steps

# Raw Data

* Raw confusing data, when NO SOFTWARE is run on it
* JSON, XML, Binary

# Tidy Data

* Each variable in one column
* Each observation in a row
* Make variables names human readable

# The Code Book

* Information about the variables, including units
* Summary choices made
* About the expiremental data design study used
* Can be in word/text file

# The Instruction List

* A computer script (R, Python, etc)
* INPUT: Raw data OUTPUT: Tidy data
* Provide instructions to break up the scripts if need be!

# Get/Set Working Directory

* Relative – setwd(“./data”)
* Absolute – setwd(“/Users/drose/data”)
* Windows uses backslash, OSX uses forward
* download.file()
* Set method to curl on mac HTTPS

# Internet

  download.file(url, destfile= “directory/filename.extension”, method  = “curl”)

\*curl for Mac with HTTPS

# Reading R

  **read.table()**

  **read.csv()**  \*automatically set sep  = “.” and header = TRUE

         na.strings   = set the character that represents missing vale (if coming from SAS: NULL)

         nrows         = how many rows to read (make it quicker)

         skip             = how many lines to skip

         col.names  = go ahead and name columns

# Reading Excel

  xlsx package**: read.xlsx(path, sheetIndex = 1, ...)**

         colIndex     = read certain columns

         rowIndex   = read certain rows

  **write.xlsx()** – create an excel file

  **read.xlsx2()** – faster, but may be unstable if subsetting

  XLConnect package has more options for altering Excel files

# Reading XML

  Frequently used in internet apps

  Extracting XML is basis for most web scraping

  components:

o   markup – labels that give text structure

o   content – actual text of the document

  tags (**Sections in document**) = <section> </section> <line-break />

  elements = <Greeting> test </Greeting>

  attributes = <image src ="a.jpg" alt = "b">

  reading file into R

o   library(XML)

* file <- xmlTreeParse(fileUrl, useInternal = TRUE)

