Assignment 2 - Visualizations

Math 154, Computational Statistics
Fall 2015

Due: Tuesday, Sep 15, 2015, noon

Summary

In this assignment, you will think carefully about graphics and visualization. You will work with some of R's sophisticated graphical tools (shiny and ggplot) as well as working to create a new and improved visualization from a graphic you have found elsewhere. The assignment will be to collect and clean a dataset, and then use that dataset to construct an interactive graphic using R's ggplot2 and Shiny packages. Skills you will learn include more detailed cleaning techniques and how to construct and interactive graphic.

Additionally, we will focus on graphics that are not particularly good or convincing. By trying to communicate details of and then re-plot information that has not been conveyed well, you will practice the art of good graphics.

Requisites

There are many `ggplot` tutorials. One that I particularly like is `http://www.ceb-institute.org/bbs/wp-content/uploads/2011/09/handout_ggplot2.pdf`. You will be able to do much more if you learn ggplot instead of relying on qplot.

Using R's package manager, download `shiny` and load it into the library. As much as needed, walk through the shiny 'Learn Shiny' tutorial, and for more detail you can consult: `http://shiny.rstudio.com/tutorial/`.

The Assignment

1. Using data you find online, create a figure using ggplot (**Extra Credit Bonus Karma Points:** and shiny). (Turn in one static plot and one plot via a video, see more below.)

   - The plot must have at least 2 aesthetics.
   - The plot must use at least two different geoms.
   - **Extra Credit Bonus Karma Points:** Demonstrate how to change the plot (or the aesthetics) using Shiny. Take (and post to Sakai!) a very short video demonstrating how your Shiny app works.

2. Exercise 7 in MDS: The `bbteams` data set in the `mdsr` package contains information about Major Leauge Baseball teams in the past four seasons. There are several quantitative and a few categorical variables present. See how many variables you can illustrate on a single plot in R. The current record is 7. [Note: this is *not* good graphical practice – it is merely an exercise to help you understand how to use visual cues and aesthetics!]

```
> # uncomment and run the lines below in your console, then recomment them to
> # knit the markdown file:
>
> #install.packages("devtools")
> #require(devtools)
> #devtools::install_github("beanumber/mdsr")
> require(mdsr)
> data(bbteams)
> names(bbteams)

 [1] "yearID"     "teamID"     "lgID"       "W"          "L"
 [6] "WPct"       "attendance" "normAttend" "payroll"    "metroPop"
[11] "name"

>
> # if you have a PC, you probably need Rtools 3.3 ...
> # download it here: https://cran.r-project.org/bin/windows/Rtools/
```

3. Check out the website Information Is Beautiful `http://www.informationisbeautiful.net/data/`. Find one plot on Information Is Beautiful (save the image to your computer and upload it into your assignment, be sure to give the URL and citation associated with the plot) that violate the concepts of effective data visualization. Write a few sentences about each plot, with a critique of what aspects of the plotting could be improved. Imagine you were going to correspond with the people who designed the plot, and give them guidance about how to make a more effective depiction of the data.

4. Using the data provided from Information is Beautiful, improve the plot in some way. (You may not be able to improve the plot overall.)

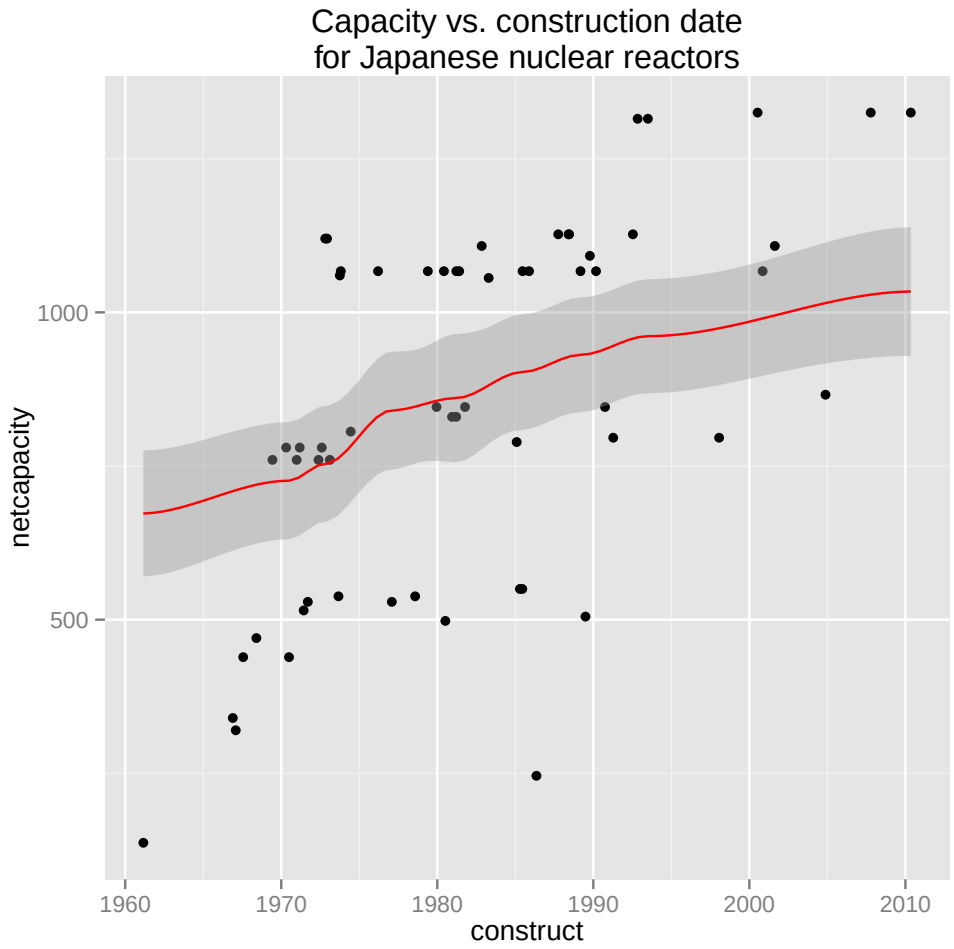
5. Describe (at least) 4 substantial ways that the poster winner "Congestion in the sky" (from the Data Expo 2009 poster competition results, `http://stat-computing.org/dataexpo/2009/posters/`) could be improved, using the concepts of effective data visualization. Write a constructive criticism that gives suggestions for improvement on each aspect that you criticize. (Note that a poster is different from one image in a paper or talk.)

An Example

ggplot example

The following is a list of all power station nuclear reactors in Japan (source: Wikipedia). The first reactors were built in the late 1960s, and the last is still under construction.

```
> library(XML)
> library(RCurl)
> library(mosaic)
> library(lubridate)
> library(dplyr)
> library(ggplot2)
> wikipedia = getURL("https://en.wikipedia.org/wiki/List_of_nuclear_reactors")
> result = readHTMLTable(wikipedia,
+                        stringsAsFactors=FALSE)
> table = data.frame(result[[22]][-1,])  # change to appropriate table number
> finaltable = mutate(table,
+                     netcapacity = as.numeric(V6),
+                     status = V5,
+                     construct = dmy(V8))  # from lubridate
> ggplot(finaltable, aes(x=construct, y=netcapacity)) +
+         geom_point() +
+         geom_smooth(colour="red", lwd=.5, degree=0, span=2/3) +
+         ggtitle("Capacity vs. construction date\nfor Japanese nuclear reactors")
>
```



Interpretation: the average net capacity of nuclear power plants in Japan tended to increase over time (but then possibly plateaued in recent years). Note that the plot here is simply exploratory, and more analysis would need to be done to truly understand the relationship.

Shiny Example

The Shiny example is provided in two files: server.R and ui.R. Once you have the two files in the appropriate directory, you can run the file using:

```
runApp()
```

Screencasting

To turn in your shiny app, please demo the capabilities of the visualization using a screencast. Then submit the screencast as a video.

Recommended Software Open Broadcaster Software, commonly known as OBS, is the absolute best free screencasting and streaming tool. Downloads for Mac, Linux, and Windows are available at `https://obsproject.com/`.

Recording a Video Follow the installation instructions and open OBS. The software's display isn't entirely intuitive. Add a scene in the 'Scenes' section by clicking the green plus sign and naming your scene. You can specify the inputs to the capture in the Sourcesfield. Add a source by clicking the plus sign, and select either Window Capture (for recording the activity in only one or more windows of your choice) or Display Capture (for recording everything on the screen). You can narrate your movie or turn off the mic. Press Start Recording, demo your app, then hit Stop Recording. The file will save automatically in your Movies Directory if you are using mac. If you cannot find the videos, under File at the top of the screen choose Show Recordings. The recordings will be named by the time they took place.

Submitting Your Video Compress your video using zip or gzip. You can do this by right clicking on the video and choosing 'Compress' from the dropdown menu. Now the file is small enough to be submitted to Sakai.

Information is Beautiful

Consider the plot at `http://www.informationisbeautiful.net/visualizations/caffeine-and-calories/`. Note that the origin is at the point (150,150). While we can get over this hurdle, it is not what is expected when looking at a graph.

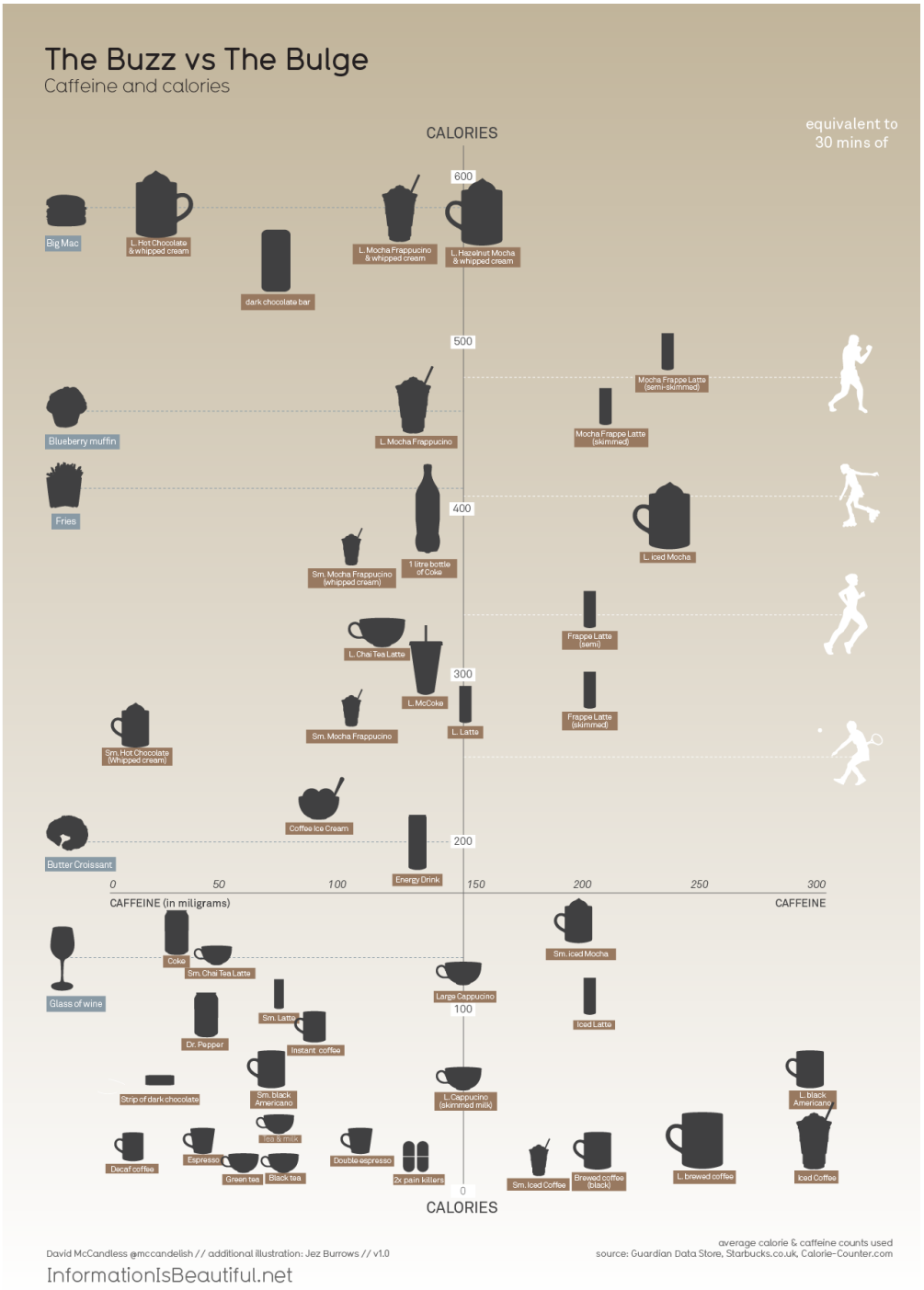


Figure 1: `http://infobeautiful3.s3.amazonaws.com/2013/01/1276_buzz_v_bulge.png`

```
> require(ggplot2)
> require(googlesheets)
> require(tidyr)
```

```
> require(dplyr)
> require(httpuv)
```

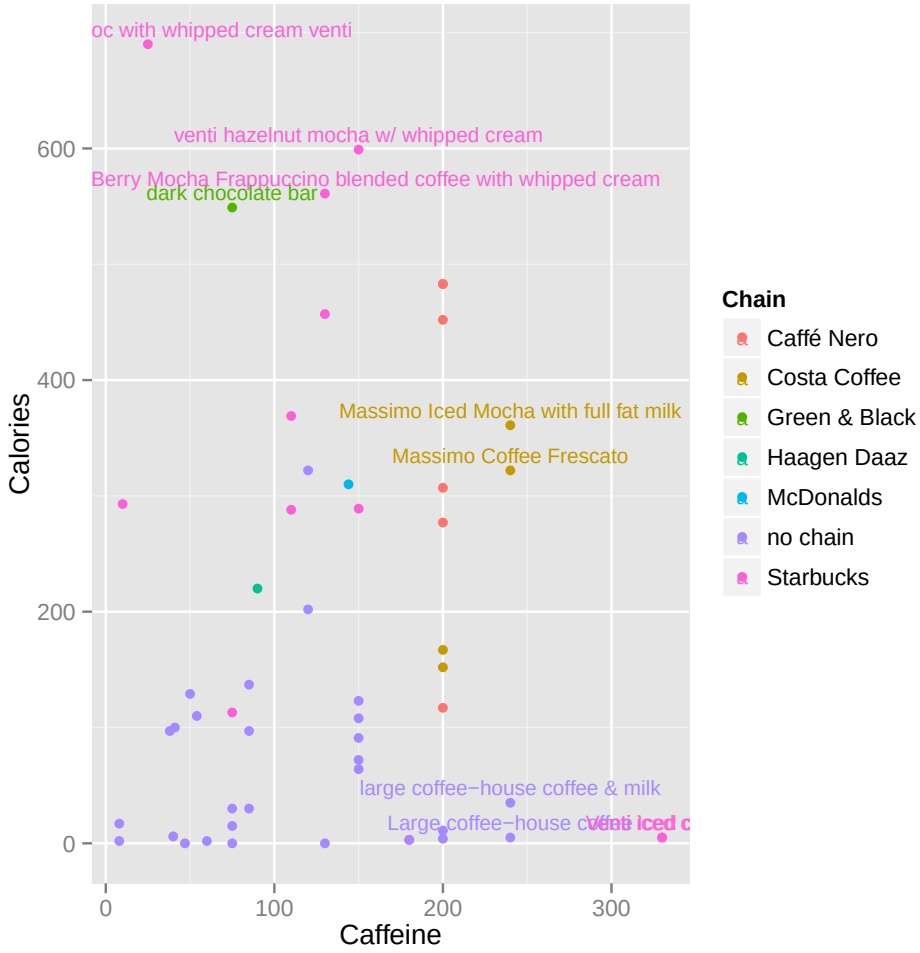
The next googlesheet requires authentication. My authorization won't work for you, but hopefully running the same code (`gs_auth()`) will work for you. See `https://github.com/jennybc/googlesheets/issues/165` for more help.

```
> # do something that will verify you are in or will get you into an
> # authorized state, such as gs_auth()
> # webpage: https://docs.google.com/spreadsheets/d/1KYMUjrCulPtpUHwep9bVvsBvmVsDEbucdyRZ5uHCDx
> # (see the markdown file for the entire URL)
>
> gs_auth()
> drinks_key <- "1KYMUjrCulPtpUHwep9bVvsBvmVsDEbucdyRZ5uHCDxw"
> drinks_ss <- gs_key(drinks_key, lookup = FALSE, visibility = "private")
> drinks_ss

                 Spreadsheet title: Caffeine and Calories
                Spreadsheet author: david.mccandless
  Date of googlesheets registration: 2015-09-11 18:50:42 GMT
    Date of last spreadsheet update: 2010-07-19 10:01:24 GMT
                         visibility: private
                        permissions: rw
                            version: new

Contains 1 worksheets:
(Title): (Nominal worksheet extent as rows x columns)
Coffees compared: 71 x 14

Key: 1KYMUjrCulPtpUHwep9bVvsBvmVsDEbucdyRZ5uHCDxw
Browser URL: https://docs.google.com/spreadsheets/d/1KYMUjrCulPtpUHwep9bVvsBvmVsDEbucdyRZ5uHCDx

> drinks <- gs_read(drinks_ss)
> glimpse(drinks)

Observations: 55
Variables: 8
$ Coffee
$ Coffee.chain
$ Calories
$ Caffeine
$ sources..World.Cancer.Research.Fund..Starbucks.Beverage.Nutrition.Guide..Calorie.Counter.Data
$ http...www.starbucks.com.menu.catalog.nutrition.drink.all.view_control.nutrition
$ http...www.wcrf.uk.org
$ http...caloriecount.about.com.
```

```
> # the glimpse function shows me the data aren't quite right... so I use the
> # command View(drinks) as another way of looking at the data.
> # Seems like what we need is:
> drinks = drinks[-1,1:4]
> glimpse(drinks)

Observations: 54
Variables: 4
$ Coffee       (chr) "Venti Dark Berry Mocha Frappuccino blended coffee wit...
$ Coffee.chain (chr) "Starbucks", "Starbucks", "Starbucks", "Starbucks", "S...
$ Calories     (dbl) 561, 369, 457, 288, 3, 5, 483, 483, 452, 307, 277, 117...
$ Caffeine     (dbl) 130, 110, 130, 110, 180, 330, 200, 200, 200, 200, 200,...

> # ahh, much better
```

Plotting the data I have removed the vertical and horizontal lines which detracted from the idea of an origin. I have also added additional information (color) to describe the chain from which the drink comes from. Notice that a different difference between my plot and the plot above is that I have many more observations than she did.

```
> drinks = drinks %>% mutate(dr.label = substr(Coffee,1,10)) %>%
+   mutate(Chain = ifelse(is.na(Coffee.chain), "no chain",
+                  ifelse(Coffee.chain == "Starbucks" | Coffee.chain == "starbucks",
+                         "Starbucks",
+                  ifelse(Coffee.chain == "mcdonalds", "McDonalds",
+                  ifelse(Coffee.chain == "haagen daaz", "Haagen Daaz",
+                  ifelse(Coffee.chain == "green & black", "Green & Black",Coffee.chain))))))

> ggplot(drinks, aes(x=Caffeine, y=Calories, label=Coffee)) +
+   geom_point(aes(color=Chain)) +
+   geom_text(size=3, vjust=-.5, hjust=0.5,
+             aes(label=ifelse(Caffeine > 200 | Calories > 500, as.character(Coffee),''),
+                 color=Chain)) #+
```



Calories and Caffeine for drinks from various drinks and other items. Data source is: World Cancer Research Fund, Starbucks Beverage Nutrition Guide, Calorie Counter Database. Seemingly, the observational units (rows) are not a random sample of anything. As such, we should be careful of summarizing the data in any way - what would the "average" calories even mean? Note, from the entire dataset give, the average calories is 179.81 and the average caffeine is 134.43. How do those numbers compare to the original plot?

Data retrieved from: `https://docs.google.com/spreadsheets/d/1KYMUjrCulPtpUHwep9bVvsBvmVsDEbucdyRZ5edit?hl=en_GB#gid=0`

More hints, help, and extensions

Shiny Help

Walk through a shiny example using their tutorial. Shiny comes with 11 fully implemented examples for you to use as inspiration, each example comes with the code used to create it. Typing `runExample(01_hello)` will launch one of these examples, note:

> Your R session will be busy while the Hello Shiny app is active, so you will not be able to run any R commands. R is monitoring the app and executing the app's reactions. To get your R session back, hit escape or click the stop sign icon (found in the upper right corner of the RStudio console panel).

In order to create your Shiny file, you need need to make two files: `ui.R` and `server.R` (see my examples on the course website as well as the examples in the Shiny tutorial). Also, don't forget that the first step is installing the shiny package onto your computer.

```
install.packages("shiny")
```

Shiny webinars are available at `https://www.rstudio.com/resources/webinars/`.

Choosing A DataSet

Feel free to get data from anywhere. There are lists of datasets on my website `http://research.pomona.edu/johardin/datasources/`. One fun place for scraping data is GapMinder. Be sure to provide the source of your data and as much information on the variables as possible.

```
> require(mosaic)
> require(ggplot2)
> require(googlesheets)
> require(tidyr)
> require(dplyr)
> litF_url = "https://docs.google.com/spreadsheets/d/1hDinTIRHQIaZg1RUn6Z_6mo12PtKwEPFIz_mJVF6
> #pulling in the URL & keeping track of how big it is
> litFurl = gs_url(litF_url, visibility="public")
> litF_nrow = litFurl$ws$row_extent[1]
> litF_ncol = litFurl$ws$col_extent[1]
> #reading in the dataset
> litF = gs_read(litFurl, range=cell_limits(c(1,1), c(litF_nrow,litF_ncol)))
> glimpse(litF)

Observations: 260
Variables: 38
$ Adult..15...literacy.rate......Female (chr) "Afghanistan", "Albania", "Al...
$ X1975                                 (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X1976                                 (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X1977                                 (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X1978                                 (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X1979                                 (chr) "4.987460442", NA, NA, NA, NA...
$ X1980                                 (chr) NA, NA, NA, NA, NA, NA, NA, "...
$ X1981                                 (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X1982                                 (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X1983                                 (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X1984                                 (chr) NA, NA, NA, NA, NA, "95.71493...
$ X1985                                 (chr) NA, NA, NA, NA, NA, NA, NA, N...
```

```
$ X1986                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X1987                                  (chr) NA, NA, "35.83991508", NA, NA...
$ X1988                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X1989                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X1990                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X1991                                  (chr) NA, NA, NA, NA, NA, NA, NA, "...
$ X1992                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X1993                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X1994                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X1995                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X1996                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X1997                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X1998                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X1999                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X2000                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X2001                                  (chr) NA, "98.25227398", NA, NA, "5...
$ X2002                                  (chr) NA, NA, "60.07508212", NA, NA...
$ X2003                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X2004                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X2005                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X2006                                  (chr) NA, NA, "63.91878516", NA, NA...
$ X2007                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X2008                                  (chr) NA, "94.68181425", NA, NA, NA...
$ X2009                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X2010                                  (chr) NA, NA, NA, NA, NA, NA, NA, N...
$ X2011                                  (chr) "13", "95.69147996", NA, NA, ...
```

```
> litF = litF %>% select(country=starts_with("Adult"), starts_with("X")) %>%
+         gather(year, litRateF, -country) %>%
+         mutate( year = extract_numeric(year)) %>%
+         filter(!is.na(litRateF))
> glimpse(litF)

Observations: 571
Variables: 3
$ country  (chr) "Burkina Faso", "Central African Rep.", "Kuwait", "Turkey"...
$ year     (dbl) 1975, 1975, 1975, 1975, 1975, 1975, 1976, 1976, 1976, 1976...
$ litRateF (chr) "3.182765695", "8.399575854", "48.01521406", "45.0989206",...
```

Bibliography Thanks much to Tessa Barton for putting together many of these resources!