16:954:597 **Homework 8** Due: Tuesday, March 31, 2020

- **NO** late submission will be accepted, except under special circumstances.
- Homework must be done individually and not in groups. Discussion of problems with others is permitted (and encouraged!), but you must write your own work in your own words.
- Submit your answers (via Canvas) as a single RMarkdown file that can be run on anyone's machine (i.e., that doesn't refer to your local files or directories). Your file name should have the following format: `lastname_NetID_week08.Rmd`. Make sure that your Rmarkdown file has yourself as author and has `output:html_document`.
- Be sure to include detailed explanatory text and remarks of what you are doing—don't just show a lot of `R` code and computer generated output. Use commands from the `tidyverse` and pipes whenever you can.

1. Read Chapter 25, "Many Models", in `R` *for Data Science* and review the `intro-to-sql.html` and `nested.html` slides from Week 8.
2. Look at the API documentation at `https://dev.socrata.com/foundry/healthdata.nj.gov/9hse-wixk`.
   (a) Write a function that will use the API and then generate a plot of the rate of heart disease over time by year. Set the default to all races, but include an option to specify the race. To get `ggplot2` to generate a plot from within a function, wrap the final object in the `print()` function. Be sure to include in your `RMarkdown` a plot generated by your function, using the default of all races and another using one specific race.
3. The *New York Times* has a nice set of APIs, described at `https://developer.nytimes.com/apis`.
   (a) Get yourself an API key.
   (b) Make 2 barplots, one with the most common non-stop-words in the titles of the Most Popular articles by views for the past week (`https://developer.nytimes.com/docs/most-popular-product/1/overview`) and another of the most common non-stop-words in the titles of the "world" Top Stores articles (`https://developer.nytimes.com/docs/top-stories-product/1/overview`).
   (c) IMPORTANT: Do **not** include your API key in your `RMarkdown` file. Instead, create a file called `api-keys.R` to store the key as a string called `api.key.nytimes`, and include `source("../api-keys.R")` in the preamble to load the file.
4. The following code will set up a connection to a demo database. You will probably need to install the `pool`, `RSQLite`, and `RMySQL` packages.

```
library(dplyr)
library(pool)
library(RSQLite)
library(RMySQL)
my_db <- dbPool(
RMySQL::MySQL(),
dbname = "shinydemo",
host = "shiny-demo.csa7qlmguqrf.us-east-1.rds.amazonaws.com",
username = "guest",
password = "guest"
)
```

This database has three tables: `City`, `Country`, and `CountryLanguage`. For example, `city <- tbl(my_db, "City")` will give you access to the `City` table.

(a) By joining the `City` and `Country` tables on the code variable (slightly different names in each table, i.e., `Code` and `CountryCode`), create a new table containing only the name of the city, the continent, the region, and the population of the city (not the population of the country).

(b) Restricting to North America, make boxplots of population by region.

(c) Taking advantage of the `nest()` function, show the 5 largest cities in each region in North America.

5. `R` has a built-in dataframe called `ChickWeight`.

(a) Using `nest()`, construct a 50×3 dataframe with columns corresponding to the chick id, the chick diet, and dataframes of each chick's data.

(b) Create a new column consisting of an `lm` model for each chick, where the regression is `weight ~ Time`.

(c) Add columns to the dataframe giving the slope and intercept for each chick (the `broom` package might be helpful, but it's not the only way).

(d) Make a scatterplot of slope (vertical axis) and intercept (horizontal axis) of the linear models colored by `Diet`.