SOC 4930/5050: PS-07 - A Data Cleaning Puzzle

Christopher Prener, Ph.D.

October 16th, 2017

Directions

This assignment creates the `childMortality` data set that is included in the latest update to Chris's `testDriveR` package. Complete as much of the puzzle as you can. Your R Notebook source (the `.Rmd` file) and `html` output should be uploaded to your GitHub assignment repository by 4:15pm on Monday, October 30th, 2017.

The Problem

The data set at the center of this puzzle originates from United Nations Children's Fund (UNICEF), a United Nations organization dedicated to improving outcomes for children and mothers around the world. These data are included in a data release on their Child Mortality website. You can download them by going to www.childmortality.org and downloading the spreadsheet linked to under "Estimates for under-five, infant and neonatal mortality". The data have a number of problems. They are included in an Excel file that includes metadata in the first few rows. These data also are formatted as wide data.

Tools

Much of this puzzle can be completed with the `dplyr` and `tidyr` functions we've learned so far this semester. You will also need two other tidyverse packages, `readxl` and `stringr`. I used the `stringr::read_xlsx()` function to import the spreadsheet. I used the `base::is.na()` and the `dplyr::slice()` functions as part of the subsetting process. I also used the following functions as part of `dplyr::mutate()` calls:

- `base::as.integer()`
- `base::as.numeric()`
- `stringr::str_detect()`
- `stringr::str_sub()`

Finally, I used the following base `R` function to convert the row in the data set containing variable names into actual variable names: `colnames(`*`dataFrame`*`) = `*`dataFrame`*`[1,]`

The Challenge

Without using any outside tools (including Microsoft Excel), import the spreadsheet from the website and clean it so that it is a tibble with 28,982 observations and 5 variables. The variables should be ordered and formatted as follows:

1. `countryISO` - `chr` vector containing three-letter country codes
2. `countryName` - `chr` vector containing country names
3. `category` - `chr` vector containing three values: `under5_MR` (under-5 mortality rate), `infant_MR` (infant mortality rate, or `neonate_MR` (neonatal mortality rate)
4. `year` - `int` vector containing years for valid estimates (no `NA` data should be included)
5. `estimate` - `num` vector containing *median* mortality rate estimates (as opposed to the upper and lower bounds also included in the data release)

The rows should be arranged by country name, category, and year in descending order. All of your code, except your initial `library()` calls and the standalone function for renaming the variables, should be built around `tidyverse` functions.

My code to complete this challenge was fairly compact: 4 lines dedicated to `library()` for loading packages, 2 standalone lines (to import the data and rename the variables), and 2 pipes that had a combined 19 lines of code. You do not need to complete the puzzle in 25 lines of code, but this should give you a sense of how efficient your process is.

Preview

Here is a preview of what you final tibble should look like:

```
> print(childMortality)
# A tibble: 28,982 x 5
   countryISO countryName  category  year estimate
        <chr>       <chr>     <chr> <int>    <dbl>
 1        AFG Afghanistan infant_MR  1961    240.5
 2        AFG Afghanistan infant_MR  1962    236.3
 3        AFG Afghanistan infant_MR  1963    232.3
 4        AFG Afghanistan infant_MR  1964    228.5
 5        AFG Afghanistan infant_MR  1965    224.6
 6        AFG Afghanistan infant_MR  1966    220.7
 7        AFG Afghanistan infant_MR  1967    217.0
 8        AFG Afghanistan infant_MR  1968    213.3
 9        AFG Afghanistan infant_MR  1969    209.8
10        AFG Afghanistan infant_MR  1970    206.1
# ... with 28,972 more rows
```