SPEC Lab R Resources: Data Management 2 - Homework

Ben Graham & Jasmine Chu

February 2022

Manipulating Data using the Tidyverse

This homework assignment requires you to download data from the World Bank Ed Stats dataset and tidy it into country year format. In addition to pivot_wider() and pivot_longer() functions from this module, you will also need to use some dplyr commands to rename variables, make character variables numeric and to filter out excess rows from the raw data. So this homework requires all your chops from both Data Management 1 and Data Management 2. If you can complete this assignment, you are well on your way to being able to wrangle data effectively for social science research.

If you are a SPEC Lab students, please save your R script and merged dataset into your homework submission folder. The R script should be titled HW_DM2_[YOUR INTIALS]

Exercise 1:

Download the following variables for all countries and all years from the World Bank Ed Stats database: Barro and Lee's measures of the number of 15-19 year-olds that have completed secondary school for both the whole population and for females only. https://databank.worldbank.org/reports.aspx?source=Education%20Statistics

```
#To complete this homework, please download data on your own and save
# the csv file to your working directory.

#The code in this answer key works with data we downloaded,
  # which we named world_bank_education.csv
#Both .csv files referenced in this answer key are available in the
#Training Data folder you downloaded from the SPEC website.

# Students should download the data themselves. Note that the code may
# look a little different if your raw data is a little different.

rm(list=ls()) #clear everything
library(dplyr) #load the correct library

dt <- read_csv("world_bank_data_education.csv") #load the csv file
names(dt)[5:55] <- c(1970:2020) #renaming year columns
```

A Reshape the data to a tidy dataset with a country-year unit of analysis

Helpful Hint: When working with variables that have more than one word, and are separated by spaces, use backticks or back quotes ().

```
# tidy and clean the World Bank data to country-year format
df <- dt %>%
mutate_at(5:55, as.numeric)%>% #all data values should be numeric
  filter(!row_number()%in%537:541)%>% #take out non-country values
```

```
  filter(!`Country Name`== "")%>% #filter out other blank rows
  rename(country = `Country Name`)%>%
  select(-`Series Code`)%>% #take out unnecessary variables
  pivot_longer(names_to = "year",
               values_to = "value",
               c("1970":"2020"))%>% #pivot long (creates year variable)
  mutate(year = as.numeric(year))%>% #make year values numeric
  pivot_wider(names_from = Series,
              values_from = value, id_cols=c(year, country))#variables as columns
```

B Rename the two education variables with concise, informative variable names and an appropriate suffix

```
#We can do this using the dplyr rename() function
df2 <- df %>%
  rename("secondary_total_BL" = "Barro-Lee: Average years of secondary schooling, age 15-19, total",
         "secondary_female_BL" = "Barro-Lee: Average years of secondary schooling, age 15-19, female")

#OR using base R and just specifying the column number -- a bit simpler

names(df)[3] <- "secondary_total_BL" #renaming education variable
names(df)[4] <- "secondary_female_BL" #renaming education variable
```

Exercise 2:

Download information on literacy rates and child (under age 5) mortality rates from the World Development Indicators.

A If these data are not already tidy, make them so.

```
dt1 <- read_csv("world_bank_data_literacy_rates.csv") #load the csv file
names(dt1)[5:65] <- c(1960:2020) #renaming year columns

dt1 <- dt1 %>%
mutate_at(5:65, as.numeric)%>% #all data values should be numeric
  filter(!row_number()%in%533:537)%>% #take out non-country values
  rename(country = `Country Name`)%>%
  select(-`Series Code`)%>% #take out unnecessary variables
  pivot_longer(names_to = "year",
               values_to = "value",
               c("1960":"2020"))%>% #pivot long (creates year variable)
  mutate(year = as.numeric(year))%>% #make year values numeric
  pivot_wider(names_from = 'Series Name',
              values_from = value, id_cols=c(year, country))#variables as columns
```

B Rename the literacy rate and child mortality variables with concise informative variable names and an appropriate suffix.

```
names(dt1)[3] <- "literacy_rate_total_WDI" #renaming variable
names(dt1)[4] <- "mortality_rate_underfive_WDI" #renaming variable
```

Exercise 3:

Merge these two datasets together on the basis of country and year, using `full_join()`. Note: SPEC Lab best-practice is to merge by Gleditsch-Ward Country Code and year, but because these two datasets both come from the World Bank, we can get away in this homework with merging by country and year. Please complete our Data Management 2A module to learn how to append Gleditsch-Ward numbers.

```
merged<- full_join(dt1, df, by=c("year", "country")) #merge both dataframes
```

OPTIONAL BONUS Complete Data Management 2A. Then append gwno numbers to both the Barro & Lee and the WDI data, check carefully for duplicate observations, evaluate which observations, if any, did not receive gwno numbers, and then merge these two datasets together on the basis of gwno and year.