Exam2RMD

Devika Kumar

6/26/2020

R Markdown

Exam 2 GOV 355M

Importing the inequality data set:

```
#clear the environment, load the rio package, and import the inequality dataset
rm(list=ls(all=TRUE))

library(rio)
inequality_data = import("inequality.xlsx")
```

This is a cross-sectional data set since the observations of data all occur within one specific point in time, in this case, the year 2015:

```
##
## Please cite as:

##  Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.

##  R package version 5.2.2. https://CRAN.R-project.org/package=stargazer

##
## ===============================================================================
## Statistic        N    Mean    St. Dev.  Min   Pctl(25) Pctl(75)  Max
## -------------------------------------------------------------------------------
## inequality_gini 80   36.814    7.941   25.400  31.550   41.125  59.100
## year            203 2,015.000  0.000   2,015   2,015    2,015   2,015
## -------------------------------------------------------------------------------
```

The year summary using the Stargazer table shows us that the standard deviation is 0, meaning that all observations in `year` are for the year 2015, proving that this is cross-sectional data.

Next, subset the `inequality_gini` for Denmark and Sweden, then Brazil:

```
#provide the subset for the inequality Gini index for Denmark and Sweden
subset(inequality_data, country == "Denmark")
```

```
##    iso2c country inequality_gini year
## 40    DK Denmark            28.2 2015
```

```r
subset(inequality_data, country == "Sweden")
```

```
##     iso2c country inequality_gini year
## 174    SE  Sweden            29.2 2015
```

```r
#provide the subset for the inequality Gini index for Brazil
subset(inequality_data, country == "Brazil")
```

```
##    iso2c country inequality_gini year
## 13    BR  Brazil            51.9 2015
```

Seeing the results, it's better to have a Gini coefficient that is lower, since that means you are closer to 0, which represents perfect income equality, like we see in these Northern European countries. Next, we take a quick peek at the data:

```r
#quick peek at the data:
head(inequality_data)
```

```
##   iso2c country inequality_gini year
## 1    AL Albania            32.9 2015
## 2    AM Armenia            32.4 2015
## 3    AT Austria            30.5 2015
## 4    BY Belarús            25.6 2015
## 5    BE Belgium            27.7 2015
## 6    BZ  Belize              NA 2015
```

Next, we will create a `remove.accent` function to remove the accent on "Belarus."

```r
#create an accent.remove function and apply it to Belarus
# define the function
remove.accents <- function(s) {
  #1 character subs
  old1 <- "áéú"
  new1 <- "aeu"
  s1 <- chartr(old1, new1, s)

  #2 character subs (hinted that it'll be here for the exam)
  old2 <- c("ß")
  new2 <- c("ss")
  s2 <- s1

  for(i in seq_along(old2)) s2 <- gsub(old2[i], new2[i], s2, fixed = TRUE)

  s2

}
# apply it to inequality_data
inequality_data$country = remove.accents(inequality_data$country)

#peek at the data after the change
head(inequality_data)
```

```
##   iso2c country inequality_gini year
## 1    AL Albania            32.9 2015
## 2    AM Armenia            32.4 2015
## 3    AT Austria            30.5 2015
## 4    BY Belarus            25.6 2015
## 5    BE Belgium            27.7 2015
## 6    BZ  Belize              NA 2015
```

Next, we will `Sort` the data by the countries with the lowest `inequality_gini` scores and then run the `head` command again.

```
#sort by inequality_gini and run head() again
inequality_data = inequality_data[order(inequality_data$country), ]

head(inequality_data)
```

```
##   iso2c country inequality_gini year
## 1    AL Albania            32.9 2015
## 2    AM Armenia            32.4 2015
## 3    AT Austria            30.5 2015
## 4    BY Belarus            25.6 2015
## 5    BE Belgium            27.7 2015
## 6    BZ  Belize              NA 2015
```

Here is the mean `inequality_gini` score for the countries with Gini scores present, i.e. with missing data removed:

```
#mean inequality_gini score
mean(inequality_data$inequality_gini, na.rm = TRUE)
```

```
## [1] 36.81375
```

Create the dummy variables for `high_inequality` and `low_inequality` to measure how countries compare to the mean and cross-tabulate these data:

```
high_inequality <- ifelse(test = inequality_data$inequality_gini <= 36.8, yes = 0 , no = 1)
low_inequality <-  ifelse(test = inequality_data$inequality_gini >= 36.8, yes = 0 , no = 1)

library(doBy)
summaryBy(high_inequality ~ low_inequality, data=inequality_data, FUN=c(mean,length))
```

```
##   high_inequality.mean high_inequality.length
## 1                   NA                    203
```

Next, we will print the names of the organizations that are working on inequality in Africa, using a for loop:

```
#create a for loop that names the IO actors
#create an organization vector
orgs <- c('World Bank', 'The African Development Bank', 'The Bill and Melinda Gates Foundation')
# Create the for statement
for ( i in orgs){
  print(i)
  }
```

```
## [1] "World Bank"
## [1] "The African Development Bank"
## [1] "The Bill and Melinda Gates Foundation"
```

Finding a variable from the WDI

Here I am choosing the indicator on **Income share held by the highest 10%** of the population. I chose this because I believe that if the highest 10% of the population holds a majority of the country's wealth, there is a high income inequality. Now I will import this directly into RStudio.

```
#importing directly into RSTudio
library(WDI)
highten_data = WDI(country = "all",
                       indicator = c("SI.DST.10TH.10"), # indicator from web
                       start = 2015, end = 2015, extra = FALSE, cache = NULL)
```

Now, we can rename the varible to be something that we actually understand, like "Highest 10% Share" :

```
#rename the var
#installing and loading the data.table package
library(data.table)
setnames(highten_data,"SI.DST.10TH.10", "Highest 10% Share")
```

Next, we will merge the two data sets together with a `left_join`. Then we will check the names to make sure there are no repeats, and if there are, we will resolve this:

```
#merge the datasets
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------
```

```
## v ggplot2 3.3.1     v purrr   0.3.4
## v tibble  3.0.1     v dplyr   1.0.0
## v tidyr   1.1.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

```
## -- Conflicts ------------------------------------------------------------------------------------
## x dplyr::between()   masks data.table::between()
## x dplyr::filter()    masks stats::filter()
## x dplyr::first()     masks data.table::first()
## x dplyr::lag()       masks stats::lag()
## x dplyr::last()      masks data.table::last()
## x dplyr::order_by()  masks doBy::order_by()
## x purrr::transpose() masks data.table::transpose()
```

```
merged_df = left_join(x=inequality_data,
                       y=highten_data,
                       by =c("country", "year"))
#however, we have repeats of the country code iso2c, so we will remove them:

#check the names of the merged data
names(merged_df)
```

```
## [1] "iso2c.x"           "country"           "inequality_gini"
## [4] "year"              "iso2c.y"           "Highest 10% Share"
```

```
# create countries spelling match variable using mutate
# note: mutate is a tidyverse command for create/generate # we include it here so you know how to use i
merged_df <-
  merged_df %>%
  mutate(cc_match = ifelse(iso2c.x == iso2c.y, "yes",
                                  "no"))
merged_df <-
  merged_df %>%
  select(-c("iso2c.x")) %>% # drop iso2c.x
  rename("iso2c" = "iso2c.y")
```

Next, `merged_df`, we will remove the missing data on the basis of `inequality_gini` and Highest 10% Share that we took from the World Development Indicators:

```
#now remove all the missing data
no_NA_df <- na.omit(merged_df, select=c("inequality_gini", "Highest 10% Share"))
```

Use a filter to only keep `inequality_gini` scores greater than 30. Save the new data frame as `data_greater_30`:

```
#filter the data
library(tidyverse)
#pipe the data through the filter like a function
data_greater_30 <-
  merged_df %>%
  dplyr::filter(inequality_data$inequality_gini > 30)
```

Next, count how many countries have the sequence "ai" in their name:

```
#which countries have "ai"?
grep(data_greater_30$country, pattern = "ai")
```

```
## [1] 53 56
```

From the results, we can see that 2 countries have "ai" in their name. Next, we will apply the sum function to the `inequality_gini` variable:

```
#apply function to sum inequality_gini
sapply(data_greater_30$inequality_gini, sum)
```

```
##  [1] 32.9 32.4 30.5 47.8 46.7 53.3 51.9 38.6 42.4 44.4 38.6 51.1 48.4 41.5 31.1
## [16] 34.0 45.2 46.0 31.8 40.6 32.7 35.0 32.7 35.9 36.5 31.7 36.0 49.6 30.4 41.0
## [31] 39.5 31.8 35.4 40.8 34.2 37.4 33.8 41.0 39.0 38.1 59.1 35.6 33.5 50.8 47.6
## [46] 43.4 44.4 31.8 35.5 35.9 37.7 40.5 36.2 32.3 34.0 36.0 43.1 37.6 32.8 42.9
## [61] 33.2 40.1 57.1
```

Now label the data:

```r
# label the variables of merged_data
#install.packages("labelled")
library(labelled)
var_label(data_greater_30) <- list(`country` = "Country",
                                   `inequality_gini` = "Gini Index Score",
                                   `year` = "year",
                                   `iso2c`= "ISO-2 Country Code",
                                   `Highest 10% Share` = "Highest 10% Share"
                                    )
```

Finally, save the data as a Stata data set!

```r
#stata data set
#save the dataset in Stata format with the labels
library(rio)
export(data_greater_30, file = "final_data.xlsx")
```

NOTE: My GitHub username is devikakumar99