R Project - Descriptive Stats and Weighting in ADH

Ulrich Bergmann Lachlan Deer

Overview

In this exercise we are going to replicate summary statistics from Autor, Dorn, Hansen (AER, 2003): "The China Syndrome: Local Labor Market Effects of Import Competition in the United States" that can be found in Appendix Table 2.

We are going to work the the 'micro' data directly from ADH. Luckily, some of our coding friends at NYU Stern have done a tonne of the heavy lifting for us and merged all of ADH's essential data together into one file.

The trickiest thing to understand is the timing of the data and the variable names.

Here is some info:

Details about timing is as follows.

The start of the period is 1991 and then end is 2007. This is then divided into two periods. The first periods is 1991-2000, thus this is a 9 year time period. They convert stuff into a "comparable decadal scale" see Footnote 22. Thus, for values for this period, they multiply them by 10/9. The same issue arises for the second period which is 2000-2007. The values for this are again converted to "decadal scales" so they are multiplied by 10/7.

The Appendix Table 2, reports the income variable and the decadal adjustments. In the summary statistics for the stuff that we care about, the ADH data is adjusted in this way described above.

That is, variables starting with 'l' are in levels whereas variables starting with 'd' are the decadal equivalents.

As necessary, we will tell you which variable to use, so that $*_s omevariable$ means to choose the appropriate level or decadal equivalent. We leave you to figure out which of the l or d variables to use. Do ask us if you are confused.

Understanding the Essence of the Paper and What Comes Next.

Read Section 1 of ADH, so that you build an understanding of there main measure 'IPW' and what the paper is about. This will help you understand the context behind the remaining exercises in this notebook and those to follow.

Your first task will be to compute some descriptive statistics from the data. To be more precise, you will replicate some of the key numbers in Appendix Table 2 of ADH. (On a side note, at least one of us thinks this table should be in the main text!)

Load Necessary Packages

Install the ones you do not have yet.

```r
library("readr")
library("tibble")
library("dplyr")
```

```
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library("Hmisc")
```

```
## Loading required package: lattice
## Loading required package: survival
## Loading required package: Formula
## Loading required package: ggplot2
##
## Attaching package: 'Hmisc'
## The following objects are masked from 'package:dplyr':
##
##     src, summarize
## The following objects are masked from 'package:base':
##
##     format.pval, units
```

Load Data

Like always, we are going to load the data and save it as a tibble

```r
df = read_csv("data/adh_data.csv") %>% as_tibble
```

```
## Parsed with column specification:
## cols(
##   .default = col_double(),
##   city = col_character()
## )
## See spec(...) for full column specifications.
```

Compute Simple Grouped Mean

1. Find which years (`yr`) are reflected in the data.

```r
unique(df$yr)
```

```
## [1] 1990 2000
```

2. Compute the average number of chinese imports per worker (`l_tradeusch_pw` & `d_tradeusch_pw`) for each "year".

```
df_yr = group_by(df, yr)

df_yr %>% summarise(l_tradeusch_pw_avg = mean(l_tradeusch_pw),
          d_tradeusch_pw_avg           = mean(d_tradeusch_pw)
          )
```

```
## # A tibble: 2 x 3
##      yr l_tradeusch_pw_avg d_tradeusch_pw_avg
##   <dbl>              <dbl>              <dbl>
## 1  1990              0.364               1.18
## 2  2000              1.12                2.64
```

Computed Weighted Group Means and Standard Deviations

For the rest of the exercise, weight the mean by population count per region instead (`l_popcount`) and compare it with the numbers in the table.

3. Repeat step 2 with weights.

```
df_yr %>% summarise(l_tradeusch_pw = weighted.mean(l_tradeusch_pw),
                    d_tradeusch_pw = weighted.mean(d_tradeusch_pw))
```

```
## # A tibble: 2 x 3
##      yr l_tradeusch_pw d_tradeusch_pw
##   <dbl>          <dbl>          <dbl>
## 1  1990          0.364           1.18
## 2  2000          1.12            2.64
```

4. Now also compute the weighted standard deviations for both variables. Hint: Use the `Hmisc` package and find the relevant function.

```
df_yr %>% summarise(l_tradeusch_pw_avg = weighted.mean(l_tradeusch_pw),
                    d_tradeusch_pw_avg = weighted.mean(d_tradeusch_pw),
                    l_tradeusch_pw_sd  = sqrt(wtd.var(l_tradeusch_pw, l_popcount)),
                    d_tradeusch_pw_sd  = sqrt(wtd.var(d_tradeusch_pw, l_popcount))
                    )
```

```
## # A tibble: 2 x 5
##      yr l_tradeusch_pw_avg d_tradeusch_pw_avg l_tradeusch_pw_sd d_tradeusch_pw_~
##   <dbl>              <dbl>              <dbl>             <dbl>            <dbl>
## 1  1990              0.364               1.18             0.325            0.992
## 2  2000              1.12                2.64             0.897            2.01
```

5. Now compute the mean and standard deviation of the average household wage and salary (`l_avg_hhincwage_pc_pw`, `d_avg_hhincwage_pc_pw`)

```
df_yr %>% summarise(l_avg_hhincwage_pc_pw_avg = weighted.mean(l_avg_hhincwage_pc_pw),
                    d_avg_hhincwage_pc_pw_avg = weighted.mean(d_avg_hhincwage_pc_pw),
                    l_avg_hhincwage_pc_pw_sd  = sqrt(wtd.var(l_avg_hhincwage_pc_pw, l_popcount)),
                    d_avg_hhincwage_pc_pw_sd  = sqrt(wtd.var(d_avg_hhincwage_pc_pw, l_popcount))
)
```

```
## # A tibble: 2 x 5
##      yr l_avg_hhincwage_p~ d_avg_hhincwage_~ l_avg_hhincwage_~ d_avg_hhincwage_~
```

```
##   <dbl>              <dbl>             <dbl>             <dbl>             <dbl>
## 1  1990             18093.             3618.             4697.             1568.
## 2  2000             21711.             1077.             5445.             2621.
```

6. And once more for share not in labor force (`l_sh_nilf`, `d_sh_nilf`)

```
df_yr %>% summarise(l_sh_nilf_avg = weighted.mean(l_sh_nilf, , l_popcount),
                    d_sh_nilf_avg = weighted.mean(d_sh_nilf, , l_popcount),
                    l_sh_nilf_sd  = sqrt(wtd.var(l_sh_nilf, l_popcount)),
                    d_sh_nilf_sd  = sqrt(wtd.var(d_sh_nilf, l_popcount))
)
```

```
## # A tibble: 2 x 5
##      yr l_sh_nilf_avg d_sh_nilf_avg l_sh_nilf_sd d_sh_nilf_sd
##   <dbl>         <dbl>         <dbl>        <dbl>        <dbl>
## 1  1990          26.7        -0.344         4.33         2.55
## 2  2000          26.4        -1.51          4.39         2.56
```

How well do your numbers line up with those reported in the paper?