Module 4 - Answers

Olle Törnquist

22/04/2021

In the next assignment we want to replicate some plots from the paper "Female Socialization: How Daughters Affect Their Legislator Fathers' Voting on Women's Issues" (Washington, 2008). The paper explores whether having a daughter makes politicians more sensitive to women's rights issues and how this is reflected in their voting behavior. The main identifying assumption is that after controlling for the number of children, the gender composition is random. This might be violated if families that have a preference for girls keep having children until they have a girl. In this assignment we will prepare a dataset that allows us to test whether families engage in such a "female child stopping rule".

Setup

- Load the libraries "Rio" and "tidyverse"

Nothing to comment except turning messages and warnings off.

```r
library("rio")
library("tidyverse")
```

- Change the path of the working directory to your working directory.

Here's my directory:

```r
setwd("C:/Users/ollet/OneDrive/Dokument/SSE/R/Module_4")
```

- import the data sets *basic.dta* and *genold108.dta*

Importing using import from the rio package:

```r
basic <- import("basic.dta")
genold <- import("genold108.dta")
```

- Create a subset of the 108th congress from the *basic* dataset

Easily done by piping the dataset into a filter function.

```r
basic <- basic %>% filter(congress==108)
```

- join this subset with the *genold* dataset

I used inner_join here but the observations are the same so left_join would do the exact same operation. The names are unique identifiers and common across both datasets.

```r
total <- inner_join(basic, genold, by = "name")
```

Data preparation

- check table 1 in the appendix of the paper and decide which variables are necessary for the analysis (check the footnote for control variables). Drop all other variables.

Comparing with the paper, I use select to choose the control variables, the dependent variables and the variable of interest, while at the same time dropping all other variables.

```r
total <- total %>% select(party, white, female, age, srvlng, region, rgroup, ngirls,
                          totchi, genold)
```

- Recode *genold* such that gender is a factor variable and missing values are coded as NAs.

I do this using factor and making empty levels into NAs.

```r
total$genold <- factor(total$genold)
levels(total$genold)[levels(total$genold)==""]<- NA
```

- Recode *party* as a factor with 3 levels (D, R, I)

Again using factor, this time specifying labels.

```r
total$party <- factor(total$party, levels=c(1,2,3), labels=c("D", "R", "I"))
```

- Recode *rgroup* and *region* as factors.

And again, the factor command.

```r
total$rgroup<-factor(total$rgroup)
total$region<-factor(total$region)
```

- generate variables for age squared and service length squared

Easily done by piping the dataset into mutate.

```r
total <- total %>% mutate(agesq=age^2)
total <- total %>% mutate(srvlngsq=srvlng^2)
```

- create an additional variable of the number of children as factor variable

Done by using mutate and as.factor simultaneously.

```r
total <- total %>% mutate(chifac=as.factor(totchi))
```

Replicating Table 1 from the Appendix

We haven't covered regressions in R yet. Use the function *lm()*. The function takes the regression model (formula) and the data as an input. The model is written as $y \sim x$, where x stands for any linear combination of regressors (e.g. $y \sim x_1 + x_2 + female$). Use the help file to understand the function.

- Run the regression $total.children = \beta_0 + \beta_1 gender.oldest + \gamma' X$ where γ stands for a vector of coefficients and X is a matrix that contains all columns that are control variables.[1]

I'm choosing to code Sanders and Goode as Democrats already before the first regression to get the same regression coefficient as in the paper. The regression itself is just a long addition.

```r
total$party[total$party== "I"] <- "D"

regression <- lm(totchi~genold+white+female+party+age+agesq+srvlng+srvlngsq+rgroup+region,
                 data=total)
```

- Save the main coefficient of interest (β_1)

Since the summary command constructs a matrix coefficients, it's easy to extract coefficients as scalars by simply selecting their coordinates. I extract the standard error as well as it will be used later.

[1] This is just a short notation instead of writing the full model with all control variables $totchi = \beta_0 + \beta_1 genold + \gamma_1 age + \gamma_2 age^2 + \gamma_3 Democrat + ... + \epsilon$ which quickly gets out of hand for large models.

```
beta1 <- summary(regression)$coefficients[2, 1]
std1 <- summary(regression)$coefficients[2, 2]
```

- Run the same regression separately for Democrats and Republicans (assign the independent to one of the parties). Save the coefficient and standard error of *genold*

Same as above, but the regressions are only performed on subsets. Of course party affiliation is then removed from the controls.

```
demreg <- lm(totchi~genold+white+female+age+agesq+srvlng+srvlngsq+rgroup+region,
             data=total, party=="D")

repreg <- lm(totchi~genold+white+female+age+agesq+srvlng+srvlngsq+rgroup+region,
             data=total, party=="R")
```

- Collect all the *genold* coefficients from the six regressions, including their standard errors and arrange them in a table as in the paper.

First, I extract the scalars from the coefficient matrices. Then I couple the betas, standard errors and numbers of observations as vectors, one per regression. In the next step I add these together to form a data frame. The final steps are done to make the layout as similar to the paper as possible: I add row names, round down the decimals and add parentheses to the standard errors.

```
beta2<- summary(demreg)$coefficients[2, 1]
std2 <- summary(demreg)$coefficients[2, 2]
beta3<- summary(repreg)$coefficients[2, 1]
std3 <- summary(repreg)$coefficients[2, 2]
All <- c(beta1, std1, nobs(regression))
Democrats <- c(beta2, std2, nobs(demreg))
Republicans <- c(beta3, std3, nobs(repreg))
coefs <- data.frame(All, Democrats, Republicans)
row.names(coefs)<- c("Coefficient", "Standard error", "Observations")
roundcoefs <- round(coefs, digits=2)
roundcoefs[2,] <- paste0("(", format(unlist(roundcoefs[2,])),")")
```

- Print the table

Here it is:

```
roundcoefs
```

	All	Democrats	Republicans
Coefficient	-0.09	0.07	-0.28
Standard error	(0.15)	(0.18)	(0.23)
Observations	227	105	122