Capstone Project

Replicating "Intergenerational Mobility and Preferences for Redistribution" (AER 2018)

March 8, 2019

Overview

The following capstone project replicates some of the key results from the paper "Intergenerational Mobility and Preferences for Redistribution" by Alberto Alesina, Stefanie Stantcheva, and Edoardo Teso (*American Economic Review* 2018). In their paper, Alesina, Stantcheva, and Teso explore how overly optimistic or pessimistic beliefs about intergenerational mobility (as measured by the difference between true and perceived mobility) is related to support for redistribution.

In this replication, you will reproduce three of the main tables and two figures, either in part or in whole. **You have 10 days to complete the replication exercise - it is due by the end of the day on March 18th.** During the replication, you are expected to work on your own - do not collaborate with other students. If you have devoted substantial time to a problem and are still in need of assistance, you may contact me for suitable hints. But please try to use the course materials, R help files, and online documentation to figure out the solutions yourself as much as possible. In the future, it is important that you're able to troubleshoot problems and navigate resources in R on your own.

Preliminaries

i. In your project folder for the course, create a new folder for the capstone exercise.

ii. Download the paper and data for the paper from here and store them in the capstone folder.

iii. Within the capstone folder, create a new R Markdown file for the exercise that will produce an HTML file.

iv. Set the following options for the document:

- Set the data frame print method to "kable".
- Set the code highlighting to "zenburn".
- Set the theme to "readable".
- Set the document to use code folding, with the code hidden by default.

v. Reproduce the instructions in your RMarkdown document, with code entered between each instruction.

vi. Be sure to commit and push your work to Github after each section AT A MINIMUM (hence, at least after "Preliminaries", "Basic data set preparation", "Table 1","Table 2", "Table 3", "Figure 2", and "Figure 3").

Basic data set preparation

i. Import the data file "Data_Descriptive_Waves_ABC.dta".

ii. Keep observations where *flag_1* and *flag_2* are equal to 0.

iii. Generate a University indicator variable, as follows:

- The variable should equal 1 when any of the following conditions hold:

  - *education* is greater than or equal to 6 and *US* is equal to 1.
  - *education* is greater than or equal to 6 and *UK* is equal to 1.
  - *education* is greater than or equal to 5 and *Italy* is equal to 1.
  - *education* is greater than or equal to 7 and *France* is equal to 1.
  - *education* is greater than or equal to 5, *Sweden* is equal to 1, and *wave* is equal to "September".
  - *education* is greater than or equal to 6, *Sweden* is equal to 1, and *wave* is equal to "February".
- The variable should equal to NA_real_ (NA value specific to numeric values) if the value of education is missing.
- Zero otherwise (if using a case_when function, you can do this by setting the last case to be "TRUE ~ 0".)

iv. Turn married into an indicator variable by recoding values of 2 for *married* to 0.

v. Generate age bracket indicators as follows:

- *age_1*, equal to 1 if age is between 18 and 29 (inclusive).
- *age_2*, equal to 1 if age is between 30 and 39 (inclusive).
- *age_3*, equal to 1 if age is between 40 and 49 (inclusive).
- *age_4*, equal to 1 if age is between 50 and 59 (inclusive).
- *age_5*, equal to 1 if age is greater than or equal to 60.

vi. Generate a *country* variable, equal to:

- "US" if *US* is equal to 1.
- "UK" if *UK* is equal to 1.
- "France" if *France* is equal to 1.
- "Italy" if *Italy* is equal to 1.
- "Sweden" if *Sweden* is equal to 1.

Table 1 - Summary Statistics

i. Begin by creating a summary statistics data frame, with averages for selected variables by country (with missing values removed when calculating the mean). Be sure to create new variable names that match the description in Table 1 of Alesina, Stantcheva, & Teso. The variables should be as follows:

Variables	
age_1	inc_bracket_3
age_2	inc_bracket_4
age_3	married
age_4	born_in_country
age_5	employed
inc_bracket_1	unemployed
inc_bracket_2	university

ii. Rearrange the summary statistics dataframe to look more like Table 1 in Alesina, Stantcheva, & Teso.

- First, transpose the dataframe and convert it into a tibble - saving this to a new object.
- Set the country variable from the original summary statistics data frame as the column names of the new tibble.
- Remove the country names from the first row of the new tibble.
- Define a new variable, called *Variable*, equal to the column names of the original summary data frame (except for the first element of the column names vector).
- Rearrange the columns to match the layout of Table 1, by using the **select()** function.

- Convert the variables *US*, *UK*,*France*,*Italy*, and *Sweden* in the summary tibble into numeric and then round the values to two digits (using the **round()** function). Be sure to vectorize the data type conversion and round functions.

iii. Turn the work in part (ii) into a function, that takes a generic summary data frame and does the exact steps outlined above.

iv. Transform the original summary data frame now using the function you've created instead. View the resulting tibble, replicating the summary statistics of Table 1.

Table 2 - Perceived and Actual Transition Probabilities

i. Begin by taking the basic data set and restricting it to keep only observations where *Treated* is equal to zero.

ii. Create a table 2 summary statistics data frame, with averages again computed by country and missing values excluded. The summary statistics should comprise the averages of the following variables (again with new variable names that match the formatting of Table 2 in the paper):

Variables	
q1_to_q5	q1_to_q3
q1_to_q4	q1_to_q2
age_3	q1_to_q1

iii. Use your table rearrangement function to transform your table 2 summary data frame so that the variables are rows and the countries are columns.

iv. Create a new variable *type* equal to "perceived".

v. Import the actual probabilities from the csv file on the course website and convert it into a tibble.

vi. Append the actual probabilities to the perceived probabilities summary you create and then sort the resulting data frame by the variable *Variable* (in descending alphabetical order).

vii. Display your data frame for Table 2

Table 3 - Relation between Perceptions and Policy Preferences

Prepare the data for analysis

i. Import the table 3 data from the file "Data_Experiment_Waves_BC.dta".

ii. Keep only the observations where *flag_1*, *flag_2*, and *Treated* all equal zero.

iii. Repeat the creation of the *country* variable for the table 3 data set

iv. Generate political spectrum position indicators as follows:

- *left*, equal to 1 if *ideology_economic* is equal to 1 or 2.
- *right*, equal to 1 if *ideology_economic* is equal to 4 or 5.
- *center*, equal to 1 if *ideology_economic* is equal to 3.

v. Generate variables for specific policy support beliefs, as follows:

- *budget_opportunities*, equal to the sum of *budget_education* and *budget_health*.

- *support_estate_45*, an indicator equal to 1 if the value of *estate_tax_support* is greater than or equal to 4.
- *unequal_opp_problem_d*, an indicator equal to 1 if *unequal_opportunities_problem* is equal to 4.
- *tools_d*, an indicator equal to 1 if *tools_government* is greater than or equal to 1.

vi. Rename the following policy support variables:

- Rename *level_playing_field_policies* to *support_eq_opp_pol*
- Rename *income_tax_bottom50* to *income_tax_bot50*

vii. Generate an indicator for whether or not someone is "rich" (household income is above the 75$^{\text{th}}$ percentile for the country.)

- First, create a summary data frame with a new variable *income_p75*, the 75th percentile of *household_income* by country. You may need to use the **quantile()** function with **summarize()**.
- Then merge these values into the table 3 data frame.
- Finally, create the *rich* indicator, equal to 1 if *household_income* is greater than the 75th$^{\text{th}}$ percentile of income.

viii. Generate the following further indicator variables:

- *young*, equal to 1 if *age* is less than 45.
- *moved_up*, equal to 1 if *job_prestige_father* is greater than 3.
- *immigrant*, equal to 1 if *parents_born_in_country* is equal to zero.

ix. Create a *country_survey* variable, which is a concatenation of the *country* and *round* variables. Convert this variable so that it is a factor.

Perform Table 3 Regressions

i. To reproduce Panels A and B Table 3:

- Create four basic model specifications, with explanatory variables comprising the controls mentioned in the next step and primary explanatory variables as follows:
  1. **Panel A, Q1 to Q1 Specification:** The main explanatory variable should be the *q1_to_q1* variable.
  2. **Panel A, Q1 to Q5 Specification:** The main explanatory variable should be the *q1_to_q5* variable.
  3. **Panel B, Q1 to Q1 Specification:** The main explanatory variables should be *q1_to_q1* $\times$ *left*, *q1_to_q1* $\times$ *right*, and *q1_to_q1* $\times$ *center*.
  4. **Panel B, Q1 to Q5 Specification:** The main explanatory variables should be *q1_to_q5* $\times$ *left*, *q1_to_q5* $\times$ *right*, and *q1_to_q5* $\times$ *center*.
     - To specify just the interaction effect and exclude the main effects of interaction terms in panel B, use the interaction notation x:y instead of x*y (supposing x and y are interacted).
- For every specification, include the following control variables:

Control Variables	
country_survey	young
left	children_dummy
right	rich
inc_bracket_1	university_degree
support_eq_opp_pol	immigrant
male	moved_up

- For every specification, perform a separate regression for each of the following dependent variables (corresponding to the model titles of Table 3):

Dependent Variables	
budget_opportunities	unequal_opp_problem_d
support_estate_45	budget_safetynet
support_eq_opp_pol	income_tax_top1
inc_bracket_1	income_tax_bot50
support_eq_opp_pol	tools_d
government_intervention	

- Save each of the regressions into a named regression object. You can either run perform these regressions using a for-loop, which is much more concise but more difficult, or manually write each of the 4 sets of 9 regressions. If you use a loop, you may need to use the **get()** function with the iterated dependent variables (and store the regressions in a list).

ii. If you did not place the regression objects in lists in the previous step, place the regressions for each of the 4 specifications in their own list now.

iii. Use stargazer with the each of the 4 lists to produce Panels A and B of Table 3. You will need to specify options for the stargazer function:

- First, create a *table_columns* vector, using the following code to get the column titles as written in the paper:

```
table_columns <- c("Budget opp",
                   "Support estate tax",
                   "Support equality opp. policies",
                   "Government interv",
                   "Unequal opp. very serious problem",
                   "Budget safety net",
                   "Tax rate top 1",
                   "Tax rate bottom 50",
                   "Govt. tools")
```

- Then use the following options in the stargazer function call (in quotes if appropriate):

Option	Value
type:	html
object.names	FALSE
style	aer
omit.stat	all
column.sep.width	2pt
font.size	footnotesize
digits	3
column.labels	table_columns

- Finally, you will also need to manually specify the values for the following options: *title*, *keep*, *covariate.labels*. Choose these such that the regressions looks like the respective panels of Table 3.

Figures

Figure 2, Panel B: Actual and Perceived Mobility Across Countries

i. Create a data set for Figure 2, by modifying the basic data set to keep only the observations where *Treated* is equal to zero.

ii. Create a figure 2 summary statistics data frame, which computes the average by country for the following variables:

- **Perceived Q1 to Q5 Transition Probability**: *q1_to_q5*
- **True Q1 to Q5 Transition Probability**: *true_q1_to_q5*

iii. Recode the country variable so that:

- Sweden is abbreviated to "SE"
- Italy is abbreviated to "IT"
- France is abbreviated to "FR"

iv. Generate a scatterplot using text labels for country instead of points, with the following formatting:

- The true Q1 to Q5 probability is on the x-axis, with range 6 to 12.
- The perceived Q1 to Q5 probability is on the y-axis, with range 6 to 12.
- Each data point (or rather text label) is colored according to *country*.
- There is a dotted reference line, using **geom_abline()**, with an intercept of 0 and slope of 1.
- Suitable titles are added for the overall graph and each axis.
- The following annotations are added to the ggplot:

Figure 3 - Accuracy of Individual-Level Perceptions

i. Create a data set for Figure 3, by modifying the basic data set to keep only the observations where *Treated* is equal to zero. *q1_to_q1* is not equal to 100, and *q1_to_q5* is less than 80.

ii. Generate the following misperception variable:

- *misperception_q1*, equal to the negative absolute value of *q1_to_q1* minus *true_q1_to_q1*
- *misperception_q5*, equal to the negative absolute value of *q1_to_q5* minus *true_q1_to_q5*

iii. Generate two different data sets:

- A *figure3_US* dataset, keeping only the observations where country is equal to *US* from the figure 3 dataset.
- A *figure3_Europe* dataset, keeping only the observations where country is *not* equal to *US* from the figure 3 dataset.

iv. Use **ggplot** to reproduce the plots of the CDF of the negative absolute error between perceived and actual transition probabilities by country.

- For each of the two graphs (US and Europe), plot both *misperception_q1* and *misperception_q5* by two seperate **stat_ecdf()** geometries to the same graph.
- In each **stat_ecdf()**, you will need to specify the x-variable in the aesthetic, as well as 'geom = "step" ' and a color to that particular ECDF. Choose 'col = "blue" ' for *misperception_q1* and 'col = "red" ' for *misperception_q5*.
- For each graph, add a suitable overall title, axis titles, and set the range of the x axis to between -80 and 0.
- Finally, apply the **theme_hc()** theme from **gg_themes()**.