DSAI Assignment on Econometrics Data

Subhasis Ray*

<2022-11-20 Sun>

1. This is an individual assignment, and the Plaksha Academic Integrity Policy applies: in short, do it by yourself: you can look up documentation, lecture notes, and textbooks, but do not take the solutions from somebody else.
2. This assignment uses data related to this published study (Miguel, Edward and Kremer, Michael, 2004)[1].
3. Save your work in a file named `DSAI_econometrics.py`. In the instructions, wherever I ask a question (not codable), insert your answer as a comment. Submit the file to codePost.
4. You can upload your figures as image files and the code on the LMS in addition.

1 Startup

1. Import the usual libraries for data analysis

```
#%% imports
import os
import numpy as np
import pandas as pd
from scipy import stats
import matplotlib.pyplot as plt
```

*subhasis.ray@plaksha.edu.in

[1] Credit goes to Dr. Kriti Khanna for sharing the research article, data, its explanation, and part of the problem statements.

2 Load the data

1. This data is in DTA format from STATA, a popular general-purpose statistical software. Pandas has `read_stata()` function for loading data in this format. Read the data into a variable named `data`.

2. Print the column names in the dataframe to see what is there. In the steps below, if some columns get omitted when printing, you can use `pd.set_option('max_columns', None)` to print every column.

3. Use `DataFrame.head()` to look at the first few entries in the dataframe.

4. You should notice a lot of `NaN` entries. These are cases where data was not available. Some of the columns are integers (like school id), but because `nan` is of type `float`, these columns have been coerced into floating point numbers.

   Keep the presence of NaNs in mind when doing the later parts of this assignment. Operations with NaN produce NaN, thus propagating through your results.

5. Summarize the data: use `DataFrame.describe()` to get some insight into the data.

6. Among these columns `pupid` seems to be pupil ID, `test96_a` and `test98` are the test scores in 1996 and in 1998, `schid98` is school ID, `t98` indicates if the pupil belongs to the treatment group in 1998 (i.e., 1 if treated with deworming medicine in 1998, 0 otherwise). `yob` is the year of birth, `prs991` indicates school participation (1 if pupil was present when the NGO visited the school, 0 otherwise). `sex` is 1 for male and 0 for female.

7. Note that sex is not a quantitative variable, and presence in school is not a continuous variable. Still we shall coerce these into our statistical tests. See `https://www.statology.org/dummy-variables-regression/`

3 Data exploration

1. Create a histogram of the test scores in '98. Specify the number of bins as 15.

2. Look at the mean, the range of the values, and also the actual numbers when you printed the data above, paying attention to the number of significant digits. Store the mean as `mean_test98`, minimum as `min_test98`, maximum as `max_test98`, and standard deviation as `std_test98`. Print all of them.

3. What do you think is going on here?

4. Compare the test scores from '96 (column `test96_a`). Apply on the '96 scores the same transformation that you think was carried out with '98 scores and add it to `data` as a new column named `test96`.

5. Create a box plot for visually comparing the '98 scores and the '96 scores (with the transformed column so that they are actually comparable).

4 Treatment effect

1. Consider the effect of treatment on school participation. Select the data rows for treated pupils in a variable named `treated` and those of the controls in `control`.

2. Create a boxplot comparing these groups in terms of observed class particpation (`prs991`). Did you face any problems? What was the solution? After working around it, how do the box plots look? Why do you think it is so?

3. What is a suitable test for checking if school participation is higher for treated students, assuming that the observations are independent? Conduct this test using the appropriate function from the `scipy.stats` module and store in a variable named `result_participation`. You may want to omit the `nan` values before passing the variables to the the scipy function.

4. Put a comment with your conclusion about the result.

5. What is a suitable test to see if the treatment had any effect on test score (comparing the scores in 1998 to those in 1996)?

6. Conduct the test using the suitable function from `scipy.stats` module and store the results in a variable named `result_scores`.

5 Regression

1. Conduct a linear regression of the dummy for school participation on the dummy for being treated using the appropriate function from `scipy.stats` module. Store the results in a variable called `result_regression`. Based on your belief about the treatement's effect, specify the suitable `alternative` parameter to the function. Which component of the result changes when you do not specify it?

2. Plot the data points as scatterplot and overlay the regression line on it.

3. Yes, the data plot looks weird, because our dummy variables can only take 0 or 1: so only four possible (x, y) pairs. But the line does tell us about the trend effected by the treatment.