PC project 1: Experiments

You are expected to solve this PC-Project until and including Exercise 2b until 28.02.2022. You may complete the remaining exercises during the session. Submit the PDF with your answers as well as your Python files with a reproducible code in pc1.py format. All functions should be submitted in a file called pc1_functions.py. Upload your solution in a zip file named pc1_yournames.zip to the designated module in Canvas.

General information

The National Supported Work Demonstration (NSW) was a program designed to analyze the effects of temporary employment programs for disadvantaged workers (women receiving social welfare, ex-addicts, school dropouts, criminals etc.) in the United States. In particular, the NSW randomly assigned applicants to the training program. While the treatment group received 9 to 18 months of subsidized employment and some additional support, the control group did not receive any assistance from the program. In 1978 the earnings of the participants in the treatment and control group (*re78*) were collected and compared. Some pre-treatment variables for both groups were measured as well. LaLonde (1986) evaluates the effects of this training program. You received a random sample of the experimental part of this dataset. The following table contains some description of this data.

Variable name	Description
treat	Dummy: 1 if participant was assigned to the training program
age, age2	Age in years (age2 = age**2)
ed	Education in years
black	Dummy: 1 if participant is black
hisp	Dummy: 1 if participant is Hispanic
married	Dummy: 1 if participant is married
nodeg	Dummy: 1 if participant did not obtain a schooling degree
re74	Annual earnings in 1974 (in 1982 dollars)
re75	Annual earnings in 1975 (in 1982 dollars)
re78	Annual earnings in 1978 (in 1982 dollars)

Part 1: Data preparation

Download the dataset *data_pc1.csv* and save it in a folder on your computer. Download *pc1.py* and *pc1_functions.py* and save them in the same folder on your computer.

a. Open *pc1.py* in Spyder. Specify the path variable accordingly and make sure that *pc1_functions* is imported. Load *data_pc1.csv* into your environment as a Pandas DataFrame object.

b. Create an output with descriptive statistics for the variables in the dataset (use the in-built functions from the Pandas module for this).

c. Additionally, open *pc1_functions.py* and code your own function that gives back adjusted summary statistics including the mean, variance, standard deviation, maximum and minimum, the number of missing and unique values and number of observations as well as the variable names in a single object. Call your function from *pc1.py* and print the results. Do you detect any missing or implausible values?

d. Drop *age2* from the DataFrame (use the Pandas module for this).

e. Plot the histograms of all continuous variables in the data. Code your own function that automatically produces nice plots including the variable name as the plot title and the option to save the plot in a pre-specified directory. Add your function to the *pc1_functions.py* file and call it from your main *pc1.py* script. What do you learn about the data? Was the treatment assignment balanced in this experiment? Do you observe any issues with the variables measuring the annual earnings?

f. Based on your descriptive analysis in exercise b, c and e, are there any anomalies in the data? Can you use the dataset in its current form for a statistical analysis? Save your final dataset in a **.csv* file (use the Pandas module for this).

g. In the *pc1_functions.py* file you find the function *balance_check()* that returns the mean differences of the variables in the two subsamples of the treated ($i = 1 \dots N_{treat}$) and the untreated ($j = 1 \dots N_{control}$) populations, its standard deviation, t-value and p-value, and additionally the so called standardized difference. The standardized difference is defined as

$$std.diff = \left| \frac{\frac{1}{N_{treat}} \sum_{i=1}^{N_{treat}} X_i - \frac{1}{N_{control}} \sum_{j=1}^{N_{control}} X_j}{\sqrt{\frac{s_{treat}^2 + s_{control}^2}{2}}} \right| * 100$$

Where the s^2 terms are the respective sample variance terms. Values above 10 are generally considered as “large”. Call this function from your main file to investigate the differences of the covariates among treated and controls. What do you conclude from the results? Are the covariate values balanced?

Part 2: ATE estimation

Continue further with your script *pc1.py* and the function file *pc1_functions.py*.

a) Estimate the ATE by simple mean differences and provide the standard error. You can use the *balance_check()* function for this or the *ate_md()* function provided in *pc1_functions.py*. Call your preferred function from the main script. How do you interpret your results? Discuss the underlying identification assumptions and their validity.

b) Code your own function, or use a function from a Python module *statsmodels* or *scikit-learn* (import the *statsmodels.api* or the *sklearn.linear_model*), to estimate OLS, providing beta coefficients, their standard errors, t-values and p-values as output. Use this function to estimate the ATE without covariates. Compare the results of exercise 2a to the results obtained now. If you notice any differences, can you explain them?

c) Use your own or the *Statsmodels* or the scikit-learn OLS function to estimate the ATE in a model that includes all pre-treatment covariates. Compare your results to those obtained in exercise 2b. What do you conclude regarding the role of covariates in experimental settings? Relate your answer to your findings in exercise 1g. Additionally, how do you judge the validity of the linear model assumption?

Part 3: CATE estimation

Proceed with the main script *pc1.py* and the loaded dataset.

a. Suppose that the policy maker wants to refine the programs according to the criterion whether a potential participant obtained a schooling degree or not.

Estimate the effect for participants with a degree and without a degree. Are the effects in the subgroups significantly different from zero? Are the effects for both groups significantly different? (Hint: The test statistic for mean differences in independent samples is given by $t = \frac{\acute{x}_1 - \acute{x}_2}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}}$ where $\acute{x}_s$ is the sample mean, σ_s^2 is the sample variance and n_s is the sample size for subsample $s = 1,2$. You can use the *ate_md()* and *test_diff()* functions provided in *pc1_functions.py* for this.)

b. Suppose that you find a significantly positive effect for a subgroup at the 5% level, how do you interpret the result? What is the probability of finding at least one significant effect at the 5% level with the two subgroups from the previous exercise when in truth the effects are all the same? With 20 subgroups? Code your own function to determine the probability. (Hint: Notice that $Pr(at\ least\ one\ significant\ effect) = 1 - Pr\ (no\ significant\ effect)$. For example, the probability of finding at least one significant effect at the 5% level with 10 subgroups would be $1 - (1 - 0.05)^{10}$).

References

LaLonde, R. (1986). Evaluating the Econometric Evaluations of Training Programs with Experimental Data. The American Economic Review, 76(4), 604-620.