ECMA 31360, PSet 3: Causal Inference with Data From a RCT

Melissa Tartari, University of Chicago

Objective: Use experimental data to estimate the effect of the offer of training on earnings. Mimic the steps of empirical analysis: 1) describe control and treated groups; 2) test balance in predetermined characteristics to ascertain whether randomization was carried out successfully; 3) estimate impact using the treated-control comparison estimator and regression adjustment; 4) understand the conceptual difference b/w the impact of the offer of training and that of undergoing training.

Background: The National Supported Work (NSW) demonstration project was a transitional, subsidized work experience program for people with long-standing employment problems. NSW was implemented in 1976-77 as a RCT. Eligible applicants were randomly assigned by the "flip of a coin" either to treatment or control. Treated individuals were offered participation in the NSW employment *cum* training program, controls were not. The outcome of interest, earnings, was measured in 1978, i.e., about one year post-intervention. You work with Dehejia and Wahba (1999, 2002)'s extract of the NSW original data.[1] Treated sample contains 185 males, control sample contains 260 males. The treatment is the offer of employment *cum* training.

Part 1: Describe the Data (10 p)

1. (10 p) Combine the NSW data in nswre74_control.csv and nswre74_treated.csv and complete Table 1. Note that variables 1-10 are predetermined, i.e., capture characteristics determined at or before treatment assignment; some of these variables are background characteristics (e.g., `edu`), others capture a subject's pre-RCT labor market experience (e.g., `u75`). `re78` is the observed outcome variable. `treat` is the indicator of treatment status. **Programming Guidance**: To load data you may use `utils::read.delim( )` where `utils` is the package and `read.delim( )` is a function included in the package; another option is `read.csv( )` from the same package. There are other packages/functions that accomplish the same task, feel free to use whichever you prefer/are familiar with. To combine (stack rows) dataframes `df1` and `df2` into one dataframe `df` you may use `df <- rbind(df1,df2)` (mnemonic for row bind) available in base R; see Example 4 here. To count how many units are included in each sample you may use `dplyr::tally(dplyr::group_by(df, treat))` which employs the `dplyr` package and the function `group_by( )` to group by the treatment column `treat` and then `dplyr::tally( )` to produce the counts. To summarize multiple columns (e.g., compute averages) you may use `dplyr::summarise_all( )` in piped format[2], i.e. `dplyr::group_by(df, treat) %>% dplyr::summarise_all(list(mean))`, or without piping `dplyr::summarise_all(dplyr::group_by(df, treat), list(mean))`.

Variable Counter	Variable Name	Variable Definition	Sample Average	
			Treated	Control
1	`age`	Age in years		
2	`edu`	Education in years		
3	`nodegree`	1 if education < 12		
4	`black`	1 if Black		
5	`hisp`	1 if Hispanic		
6	`married`	1 if married		
7	`u74`	1 if unemployed in '74		
8	`u75`	1 if unemployed in '75		
9	`re74`	Real earnings in '74 (in '82 $)		
10	`re75`	Real earnings in '75 (in '82 $)		
11	`re78`	Real earnings in '78 (in '82 $)		
12	`treat`	1 if received offer of training	1	0
Sample Size			185	260

Table 1: Descriptive statistics for the NSW data by group.

[1] Dehejia and Wahba (1999) Causal Effects in Nonexperimental Studies: reevaluating the Evaluation of Training Programs, *JASA*, pp. 1053-1062 and Dehejia and Wahba (2002) Propensity-score Matching Methods for Nonexperimental Causal Studies, *ReStat*, pp. 151-161. The original data (not used in the pset) is available at the ICPSR page.

[2] Pipes let you take the output of one function and send it directly to the next, which is useful when you need to apply multiple transformation to the same data set. Pipes in R look like %>% and are made available via the `magrittr` package installed as part of `dplyr`.

Part 2: Test Balance (55 p)

Background on Covariate Balance: Proper random assignment of treatment *balances* all the subjects' characteristics, including all determinants of the outcome (but for treatment status), both observed and unobserved. Thus, an implication of proper randomization is that there should be no systematic differences (i.e., no "imbalance") between control and treatment groups in terms of their observed predetermined variables (OPVs).[3] You always want to check this implication in your data, because what you find informs how you setup estimation. Below you carry out the check in two ways: (Q1-2) test the hypothesis that the two groups have the same means for all OPVs; (Q3) test the hypothesis that OPVs do not predict treatment status.

1. (5 p) Test balance for each of the 10 OPVs in Table (1), i.e., test that each variable's mean is the same in the control and treated groups. Do so by running 10 simple linear regressions (SLR) specifications. Use a 5% significance level and look at the relevant 10 t-tests, comment on your findings. **Hint**: Each OPV is the dependent variable in its regression equation. All 10 SLR models have the same covariates. **Programming Guidance**: To run a SLR use `stats::lm( )` where `stats` is a package and `lm( )` is a function that estimates linear-in-parameter models. It is convenient to first create the formula for the model then pass it to `lm ()`. Example: Declare `formula <- stats::formula(paste(age, ''~treat''))` to create the formula for a SLR that has `age` as the dependent variable regressed on a constant and the treatment indicator (printing `formula` to standard output (stout) yields `age` $\sim$ `treat` because the constant is left implicit and present by default). Then pass the formula to `lm( )` by typing `lm_model <- lm(formula = formula, data = df)`. To retrieve regression output use `summary(lm_model)`, where `summary( )` is available in base R. Example: Retrieve regression coefficients with `summary(lm_model)$coefficients`. To repeat for each OPV, employ a `for` loop or e.g., `lapply( )`, see here for an example.

2. (40 p) Testing balance as done in **Q1** suffers from the so called "multiple comparisons" or "multiple testing" problem which occurs when one considers a set of statistical inferences simultaneously. The problem emerges because as more variables are compared, it becomes more likely that the treatment and control groups appear to differ on at least one attribute *by random chance alone*. To deal with this problem we use an estimation methodology called SUR estimation and then test just one hypothesis, the *joint* hypothesis that all OPVs are balanced, i.e., their means are the same in the two groups. SUR stands for seemingly unrelated regression and it is a special case of feasible Generalized Least Squares (GLS) estimation. Instead of estimating the coefficients *equation-by-equation* by OLS (and done in **Q1**) you combine the 10 equations in a system of equations and estimate the coefficients present in all the equations jointly, accounting for the fact that the unobservables may be correlated across equations within an individual (we continue to assume that they are uncorrelated across units). After estimation, you use standard testing procedures to test the *joint* hypothesis that the slope coefficients in all the equations of the SUR system are zero. This joint test is a test of covariate balance that does not suffer from the "multiple testing" problem.[4]

(a) (25 p) Estimate the SUR system. Are the estimated coefficients and their SEs different from those obtained in **Q1**? Comment. **Hint**: In 2 situations there is no efficiency payoff to GLS versus OLS: 1) the unobservables are uncorrelated across equations within an individual; and 2) the equations have identical covariates. **Programming guidance**: Use `systemfit::systemfit`, where `systemfit` is both the R package and of the function. The vignette is here, go directly to Section 4.1 and 4.2. Implementation takes 3 steps: 1) collect the 10 formulas in a list (create each formula with `stats::formula()`); 2) pass the list to `systemfit::systemfit`, specifying `method = ''SUR''`; 3) summarize the output using `summary( )`. Example: If your list of formulas is named `sur_system`, you type `sur_fit <- systemfit::systemfit(formula = sur_system, data = df, method = ''SUR")` then `summary(sur_fit)`.

(b) (15 p) Test *joint* balance by testing the joint hypothesis that the coefficients of `treat` are zero in all the equations of the system. Spell out null and alternative hypotheses, comment on findings, and verify the test's value and p-value "manually." **Hint**: Use the Likelihood Ratio (LR) test where the unrestricted model is the system of equations with `treat` as covariate, and the restricted model is the system with no regression covariates, i.e., with only the constant term. To verify the value of the test use its algebraic expression. Recall that the test has a χ^2_{df} distribution with df = number of restrictions. **Programming guidance**: Implementation of the test takes 3 steps: 1) create a list of formulas named `null_system` collecting specifications with only the constant; 2) pass the list to `systemfit::systemfit`: `null_fit <- systemfit::systemfit(null_system, data = df, method = ''SUR")`; 3) run the LR test `lrtest_obj <- lmtest::lrtest(null_fit,sur_fit)`. Check the attributes of `lrtest_obj`. Use `stats::pchisq()` to verify the test's p-value.

3. (10 p) Test that the OPVs do not predict treatment assignment. Spell out null and alternative hypotheses, comment on findings, and verify the test's value and p-value "manually." Why would scientists carry out this test? **Hint:** Use a MLRM and the F-test for the overall significance of the regression. **Programming guidance**: Use `summary()` after estimation, e.g., `summary(lm_fit)$fstatistic` returns the test's value and degrees of freedom. Use `stats::pf()` to verify the test's p-value.

[3] OPVs are often called "covariates," the name stems from the traditional approach to causal inference which uses OPVs as regression covariates, i.e., as right-end-side variables in a regression equation.

[4] This approach is appropriate when the number of tests is relatively small. For large scale multiple testing, e.g., in genetics association studies where millions of tests are carried out, you would employ different methodologies. For example, you may use an an approach based on the family-wide error rate: to ensure that the probability of a type 1 error (i.e., of erroneously concluding that at least one characteristic is different between two groups) is less than 5% you use 5% divided by the number of tests as the significance level that you compare to the p-value of each individual test.

Part 3: Estimate the Effect of the Offer of Training (35 p)

Objective: Your target is the average treatment effect (ATE) of receiving a training offer on post-intervention earnings (`re78`). Here you estimate this estimand using the approaches listed in Table 2.

Spec	Estimation Approach	Description
0	T-test	Compute group-specific sample averages, take difference.
1	OLS	Estimate coefficient of SLRM.
2	OLS	Add to spec. 1 OPVs `nodegree` and `edu` in linear form.
3	OLS	Add to spec. 2 the other 8 OPVs in linear form.
4	OLS	Add to spec. 3 the interaction b/w `age` (in deviation from its sample mean) and `treat`

Table 2: Specifications used to estimate the ATE of offer of training

1. (5 p) Implement specs 0 (3 p) and 1 (2 p). Describe your findings. **Programming Guidance:** To test H_0 that ATE is zero in spec 0 use `stats::t.test()` which implements Welch's Two Sample t-test. To test H_0 that ATE is zero in spec 1 use `summary()` on the object returned by `stats::lm()`.

2. (20 p) Implement specs 2 through 4.

(a) (9 p) Report the estimates of the ATE and test H_0 that ATE is zero (3 p each spec). **Hint:** You consider spec 2 in light of the imbalance in educational attainment documented in Part 2. This approach to account for the presence of observable confounders (i.e., imbalance in OPVs) is called the regression adjustment approach. Spec 3 has 11 regression covariates. Spec 4 has 12 regression covariates.

(b) (5 p) Are there reasons to add OPVs as covariates when they are balanced across treatment and control groups? If so, what are they? Comment on the estimation results from specs 2 and 3.

(c) (1 p) With reference to spec 3, do you think that it is problematic to use lagged / past values of the dependent variable as regression covariates? Explain.

(d) (5 p) Are there reasons to add interactions between OPVs and the treatment indicator? If so, what are they? With reference to spec 4, test the following two hypothesis: i) the ATE is zero; ii) the effect of the treatment does not vary by the age of the subject. **Note:** If additional assumptions are needed to carry out i) and ii), state them. **Programming Guidance**: Use `car::linearHypothesis()`.

3. (5 p) Brainstorm on the possible mechanisms for the estimated ATE of being offered training. **Hint:** Can you think of the possible pathways through which on-the-job training may cause an increase in post-intervention earnings?

4. (5 p) In this question you brainstorm on why the ATE of the offer of training may be different from the ATE of undergoing training. Terminology-wise, the ATE of the offer of training is called an Intent to Treat Effect (ITT). To think about the relationship between these two treatment effects in a systematic way you consider the following setup. Let Z_i denote a binary variable that takes the value 1 if individual i is offered training, and zero otherwise. Let D_i be a binary variable that takes the value 1 if individual i undergoes training, and zero otherwise. Thus, for instance, and individual who is offered training but does not take it up has $(Z_i, D_i) = (1, 0)$. Assume: 1) when an individual receives an offer of training he flips a coin, if it comes up Head he enrolls in the training program while if it comes up Tail he does not; 2) individuals not offered training cannot take it up; 3) the offer of training does not *per se* affect future earnings.[5] Each individual has **two pairs** of potential outcomes, one pair is associated with treatment defined as being offered training and the other pair is associated with treatment defined as undergoing training. Specifically, let (y_{1i}, y_{0i}) denote potential 1978 earnings *with* training and *without* training. Similarly, let (y^o_{1i}, y^o_{0i}) denote potential 1978 earnings *with, and respectively without, the offer of training in hands*. In **Q1**-**Q3** your goal was to estimate the ATE of being offered training, let us denote it by $ATE^o \equiv E[y^o_{1i} - y^o_{0i}]$ (which is the ITT effect). In this question you are asked to compare ATE^o to $ATE \equiv E[y_{1i} - y_{0i}]$, and show/discuss why they may be different. **Hint #1**:To answer this question start by relating the two pairs of potential outcomes, specifically, express (y^o_{1i}, y^o_{0i}) in terms of (y_{1i}, y_{0i}). Go from there. **Hint #2**: You have all the information necessary to obtain an exact relationship b/w ATE^o and ATE. **Hint #3**: The key lesson to take from this question is that ATE^o is typically different from ATE even in the absence of self-selection into the training program.

[5] This assumption rules out e.g., a case in which, by virtue of receiving a training offer, an individual becomes more optimistic about his future labor market prospects and searches for a well paying job harder than he would have had he not received the offer.