Regression II
Vanderbilt University
Prof. Sean P. Corcoran

Lecture 2 In-Class Exercises

1.1 **Exact and nearest neighbor matching.** *See Lecture 2 matching with simulated data* do file for code.

(a) Create a simulated dataset with 750 observations on age (18-40), education (0-1), treatment, and the outcome. In this dataset, y is the outcome (e.g., earnings or self-reported health) which is related to age, education, and a treatment. Treatment status is related to both age and education, and there is a constant treatment effect of +30. Multiple regression would work fine with this data, but we are going to use it for matching.

(b) Estimate the ATE and ATT after conducting an exact match on age using `teffects`. After matching, check the balance of your covariates using `tebalance summarize`.

(c) Repeat part (b) but do an exact match on age and education. Include the `gen()` option to store the observation numbers of the exact matches.

(d) For the next examples, keep a subsample of 200 observations. This will reduce the number of exact matches.

(e) Estimate the ATE and ATT after conducting a nearest neighbor match on age and education using `teffects`. First use the Euclidean distance metric, then use Mahalanobis distance. Use 5 nearest neighbors. Use `tebalance summarize`, `tebalance box`, and `tebalance density` to check balance on the matching variables.

(f) After estimating the ATT in (e) using nearest neighbors and Mahalanobis distance, use the `gen()` option to store the observation numbers of the exact matches, and the post-estimation command `predict` to get the distance to the nearest neighbors.

(g) Now use the `predict` command to get "potential outcomes" and individual "treatment effects." Note the counterfactual (unobserved) potential outcome and treatment effect are imputed based on the matched nearest neighbors.

1.2 **Propensity score estimation and matching.** This example uses a dataset with 4,727 community college students enrolled in an English 1 course on their first attempt. The goal here is to estimate the impact of enrollment in a writing center course which provides supplemental supports to English 1 students. We will use propensity score matching to attempt to create balance between the treated (course takers) and untreated (non-course takers) and then estimate the ATT. See *Lecture 2 propensity score example* do file for code and more detail.

(a) There are two course outcomes of interest: a binary measure of "success" (passing with a C or better) and a 0-4 grade point measure. Estimate the naive regression of each outcome on the treatment variable (enrollment in the writing center). Look descriptively at the characteristics of students enrolled in the writing center and those who are not. Do you have concerns about selection bias?

(b) Use `teffects psmatch` to estimate propensity scores and find nearest neighbor matches. Use the default logit model and 1 nearest neighbor (note ties will be retained). Estimate the ATT on the course grade. For now use a parsimonious set of matching variables: age, gender, race/ethnicity, first generation, GPA at the beginning of the term, and first time student. Check the balance using `tebalance summarize` and the distribution of propensity scores using `teffects overlap`. It's worth noting that this sample is pretty balanced from the outset; enrollees in the writing center don't look *that* different from those who do not enroll.

(c) You can use `predict` post-estimation to get the estimated propensity scores and distance to nearest neighbor(s). Look descriptively at distance to see how far away the typical neighbor is, and whether there are outliers that you might deem "too far" away.

(d) Add more predictor variables to the propensity score model: high school GPA, financial aid recipient, units attempted to date, units attempted this term, nonresident tuition, international student, online course, and dummies for course term. Do the same checks for balance as in part (b). Estimate ATTs for both course grade and "success."

(e) Now try the older `psmatch2` command to estimate propensity scores and nearest neighbor matches. This command has some nice features not found in `teffects`. For comparability with `teffects`, you will need to use the options `logit` and `ties` to use a logit model and keep ties. Use the same predictor variables as (d).

(f) Use `pstest` after post-estimation to check balance in covariates and `psgraph` to compare the distributions of propensity scores. These are comparable to `tebalance summarize` and `teffects overlap`.

(g) See the do file for an exploration of variables created by `psmatch2` that begin with `_*`. These are quite useful for identifying observations used in the matched sample, and their weights (if matched more than once). You can use these to create your own plots showing the distribution of propensity scores in the matched sample.

(h) Rather than matching, use the estimated propensity scores to do inverse probability weighting, estimating the ATT with `teffects ipw`. Use the same predictor variables as (d). You can verify the ATT calculation yourself by calculating the IPWs (1 for treated cases and $p/(1-p)$ for untreated cases) and taking the weighted average.