Matching and Propensity Score Matching

Intermediate Data Analytics

Dr. Alex Marsella

April 12th, 2024

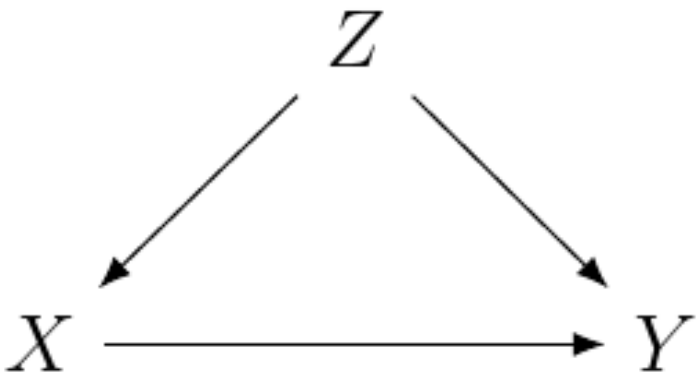
First, we'll start with the basic concept of "Matching", then we'll talk about "Propensity Score Matching".

Matching

- A family of statistical techniques that involve finding units that are treated and untreated in some observational study and "matching" them so that untreated units are very similar to treated units across observable characteristics, aside from treatment status itself.
- Basically involves subsetting your control (untreated) group so that it's more similar to the treated group, dealing with potential selection into treatment.
- It's one of many ways we can try to mimic randomization and deal with confounding.

Matching

- Imagine a job training program that affects your wage and employment prospects.
- Suppose the pool of eligible people is 50/50 men and women.
- Suppose it's mostly advertised to men, the program enrollment is 80% men.
- If gender also affects job outcomes as well, then it's a confounder.
- Z = Gender, X = Training Program, Y = Job Outcome



What would matching do for us?

- Looks at the eligible pool of untreated people (50% men / 50% women) and reconstructs it to be 80% male, 20% female.
- Now, gender is the same between both groups.
- No variation in gender –> no correlation between gender and treatment –> no longer a confounder!
- We could now run a linear regression of Y on X without even controlling for gender!

But then why not just control for gender??

Here's why

- Linear regression assumes linear relationships between things.
- If the variables you would match on have non-linear relationships to the outcome, controlling for them in linear regression would not work anyway.
- For a simple example like gender, it might not matter so much, but if you have many potential complex confounders, it may be better to match.

What's happening behind the scenes in our example?

- Treated Group: 80 men, 20 women
- Of the 80 men, 60 get a job and 20 do not. Of the women, 12 get a job and 8 do not.
- Control Group: 500 men, 500 women. 350 men get a job, 150 do not. 275 women get a job, 225 do not.
- Raw comparison: $(60+12/100) = 72\%$ of treated group get a job.
  - 62.5% of control group get a job.
- The problem: if men are more likely to be hired anyway, then the estimated effect of the job program has [what kind of] bias?
- Just this are comparison of 72% - 62.5% is an estimate of the effect! And it's biased by the confounder of gender.

What matching does behind the scenes

- Gives a weight of $80/500 = .16$ to all untreated men
- Gives a weight of $20/500 = .04$ to all untreated women
- Now, recalculating employment for untreated group, given 350/500 men got jobs, and 275/500 women got jobs.
- $\frac{.16 \times 350 + .04 \times 275}{.16 \times 500 + .04 \times 500} = 67\%$.
- So instead of 72% - 62.5%, it's 72% - 67% when it's reweighted to account for the bias toward males.
- Job training program unbiased estimate is 5ppt increase, whereas biased estimate was a 9.5ppt increase!

Propensity Score Matching

Introduction to Propensity Score Matching

- Propensity Score Matching (PSM) is a statistical technique used to reduce confounding in observational studies, where random assignment is not possible.
- It involves pairing individuals from treatment and control groups with similar **propensity** scores.
- **Propensity Score:** The predicted probability (0-1) of a unit being assigned to treatment based on a set of variables (we call them "covariates") we believe predict *selection/assignment into treatment*.
- Instead of removing all variation in matching variables (like in the previous example), we model treatment likelihood (propensity) based on a slew of variables we think affect treatment and then match on that likelihood.

Why Propensity Score Matching?

- In observational studies, treatment assignment is not random, leading to potential confounding.
- PSM helps to mimic a randomized controlled trial by ensuring that matched pairs have similar covariate distributions.
- This reduces the bias in estimating the treatment effect.

Estimating Propensity Scores

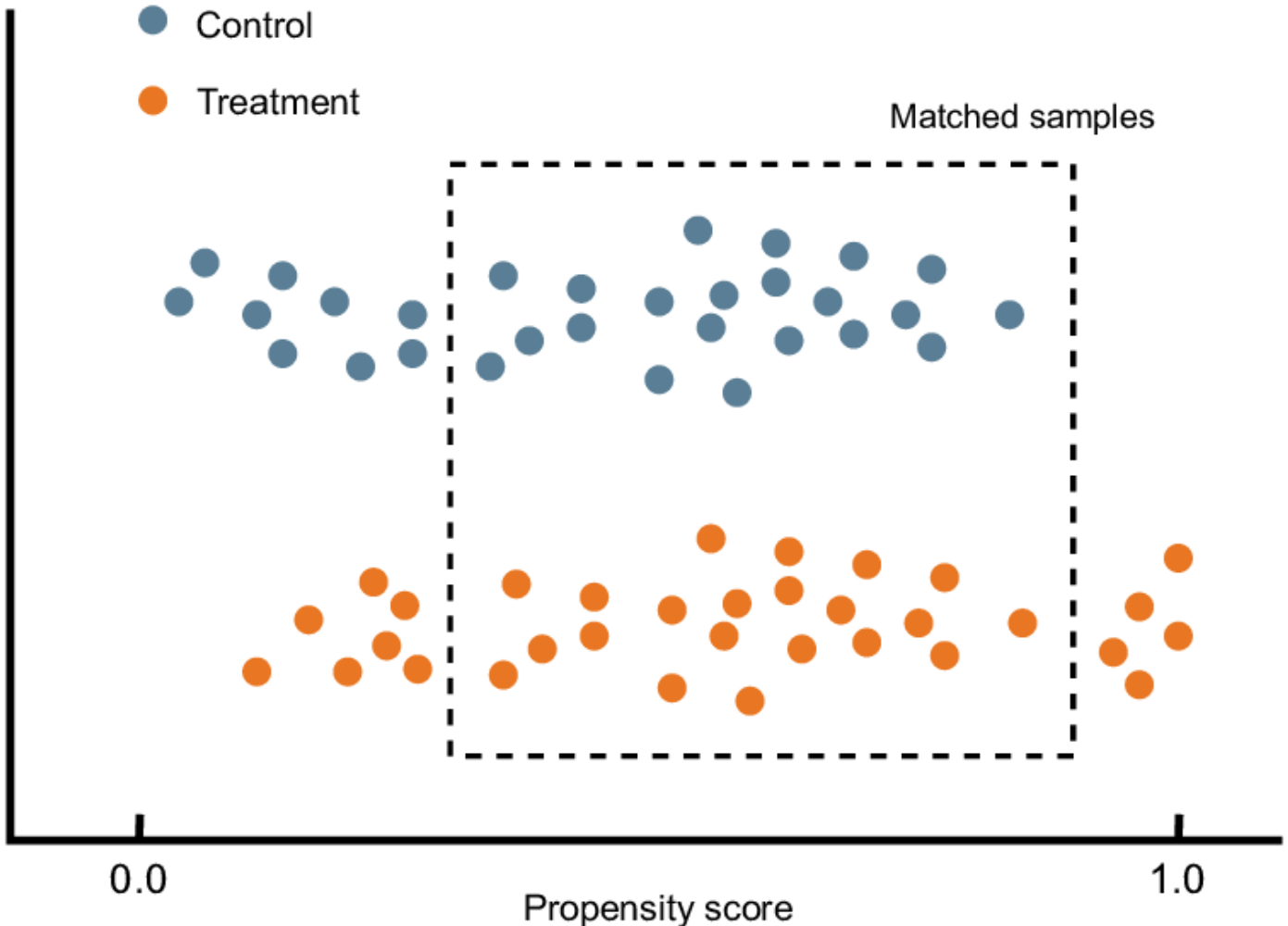
- Propensity scores are typically estimated using logistic regression.
- The model predicts the probability of receiving the treatment based on observed covariates.
- For our code-along next Wednesday, we'll examine the effect of Catholic schools.
- For today's lecture, I'm using data on individuals who entered into a job training program compared to those who did not.
  - From Lalonde, R. (1986), "Evaluating the Econometric Evaluations of Training Programs," American Economic Review, 76, 604-620.

Our observational data

- We know **after the fact** that some of these people went into a job training program, and we know their wages in 1974, 1975, and 1978, along with other characteristics.
- They were not randomly assigned to the program, they chose to enter. This is a confounder problem! If you CHOOSE to enter a job training program, you probably have higher motivation then people who don't. Motivation affects wage.
- Sadly, we can't observe motivation, so we opt for a bunch of other factors we think help us predict selection into the program.

Nearest neighbor propensity score matching

- For every treated unit, it finds one untreated unit that is most similarly likely to be treated.
- Results in a control group of the same size as the treated group.



What propensity score matching in R is going to do

- We use a package called `MatchIt` that does it all for us, but basically, it will

1. Estimate a logistic regression model with the treatment as the outcome.
2. Calculate "propensity scores" by essentially using the `predict()` function like we did for the titanic survivors.
3. Find, within, the control group, units that have similar propensity scores to our treated units.
4. Creates a new dataframe for it, that we can then run analyses on.

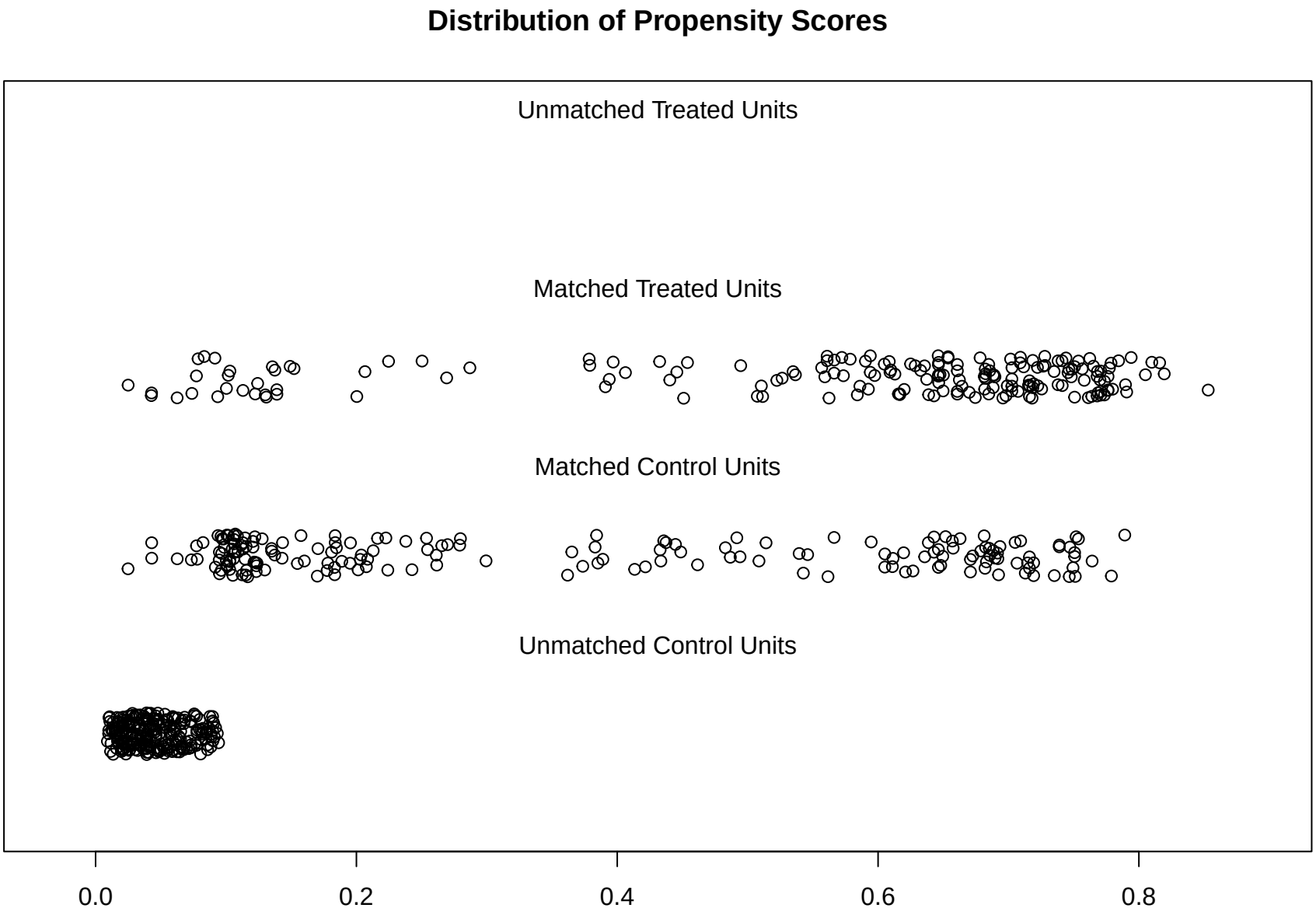
```
##
## Call:
## matchit(formula = treat ~ age + educ + race + married +
##     re74 + re75, data = lalonde, method = "nearest")
##
## Summary of Balance for All Data:
##              Means Treated Means Control Std. Mean Diff. V
## distance            0.5774        0.1822           1.7941
## age                25.8162       28.0303          -0.3094
## educ               10.3459       10.2354           0.0550
## raceblack           0.8432        0.2028           1.7615
## racehispan          0.0595        0.1422          -0.3498
## racewhite           0.0973        0.6550          -1.8819
## married             0.1892        0.5128          -0.8263
## nodegree            0.7081        0.5967           0.2450
## re74             2095.5737     5619.2365          -0.7211
## re75             1532.0553     2466.4844          -0.2903
##              eCDF Max
## distance       0.6444
## age            0.1577
```

Performing Matching

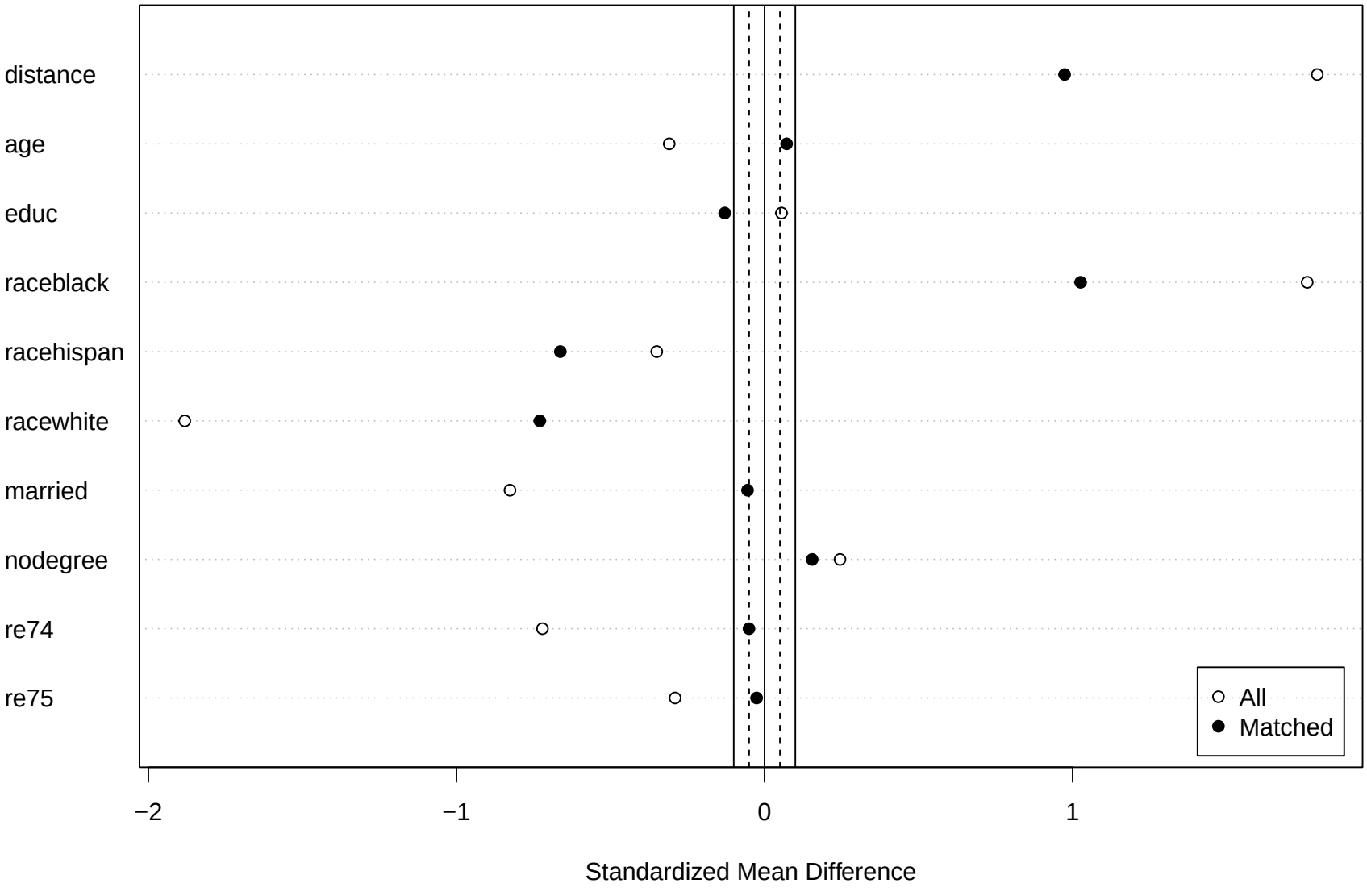
- After estimating propensity scores, we match individuals from the treatment and control groups.
- Many methods are available: nearest neighbor, caliper matching, stratification, etc. We will focus on "nearest neighbor", which is what I have been describing.

Assessing Match Quality

- After matching, it's important to assess the balance of covariates between treatment and control groups.



Love Plot


distance
age
educ
raceblack
racehispan
racewhite
married
nodegree
re74
re75
All
Matched
−2
−1
0
1
Standardized Mean Difference

Estimating Treatment Effects

- The final step is to estimate the treatment effect using the matched sample.
- We use `match.data()` on our object that we made using `matchit`, to create our matched dataset.

```
##
## Call:
## lm(formula = re78 ~ treat, data = matched)
##
## Residuals:
##    Min      1Q  Median      3Q     Max
##  -6349   -5455   -2106    3239   53959
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   5454.8      516.4  10.563   <2e-16 ***
## treat          894.4      730.3   1.225    0.221
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1
```

Exercise

- Create two more models, one with no controls, one with all the controls, and then put them all in a stargazer table using my code below.
- We can compare our estimates and think about potential biases.