Matching Estimators of Causal Effects

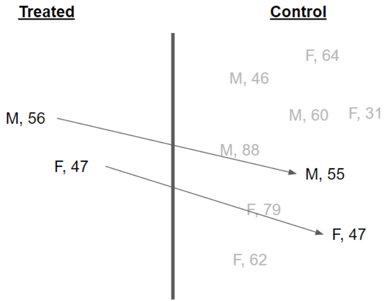
Hans Jarett Ong

September 8, 2022

Contents

Overview of Matching

Example of Matching



Example of matching copied from *A Crash Course in Causality (Coursera)*

- In this example, assume we're controlling for gender and age.
- We're looking for units in the control group that are similar to the treatment group.

Causal effect of treatment on the treated

- Since we are making the covariate distribution in the control look like the treated, we are effectively measuring the *causal effect of treatment on the treated*.
- This is also known as the **Average Treatment Effect on the Treated (ATT)** which is different from ATE.

$$\text{ATT} = E[Y^1 - Y^0 | D = 1]$$

- In most cases, we are under sampling the control group.

Fine Balance

- A less strict form of matching where we accept non-ideal matches as long as the final distributions of the treated and control have the same marginal distributions.
- e.g. we might accept

$$P(\text{Gender}, \text{Age}|D = 1) \neq P(\text{Gender}, \text{Age}|D = 0)$$

as long as

$$\begin{gathered} P(\text{Gender}|D = 1) = P(\text{Gender}|D = 0) \quad \text{and} \\ P(\text{Age}|D = 1) = P(\text{Age}|D = 0) \end{gathered}$$

Number of matches

- One to one (pair matching)
  - Simplest form of matching; does not over sample the treated units.
  - But this discards a lot of available data (controls with no matches).
- Many to one
  - Match each treatment unit with a *fixed number* of controls.
  - e.g. 5 control units per treatment unit
- Variable
  - The number of matches per treatment unit will be variable depending on the availability of good matches.

Matching Using a Distance Metric

Matching using a Distance Metric

- This is the most straightforward matching technique.
- Match covariates using some distance metric.
- We'll explore 2 metrics:
    - Mahalanobis Distance
    - Robust Mahalanobis Distance
- and 2 matching strategies:
    - Greedy (nearest neighbor) matching
    - Optimal matching

Mahalanobis Distance

- The Mahalanobis Distance D is defined as:

$$D(X_i, X_j) = \sqrt{(X_i - X_j)^T S^{-1} (X_i - X_j)}$$

  - X_i is the covariate vector for subject i.
  - S is the covariance matrix
- Intuition: If there are no covariances between the covariates, then this is equivalent to scaling (using std. dev.) and using Euclidean distance.
- This just makes sure that axes with naturally large variances are not over represented in the distance computation.
- Another intuition: This is equivalent to using Euclidean distance on the scaled (using std. dev.) PCA-transformed data.

Mahalanobis Distance

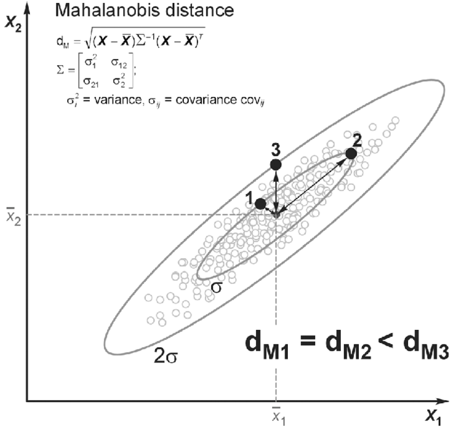

Mahalanobis distance
$d_M = \sqrt{(\boldsymbol{X} - \overline{\boldsymbol{X}})\Sigma^{-1}(\boldsymbol{X} - \overline{\boldsymbol{X}})^T}$
$\Sigma = \begin{bmatrix} \sigma_1^2 & \sigma_{12} \\ \sigma_{21} & \sigma_2^2 \end{bmatrix};$
σ_i^2 = variance, σ_{ij} = covariance cov_{ij}
X_2
$\bar{x}_2$
3
2
1
σ
2σ
$d_{M1} = d_{M2} < d_{M3}$
$\bar{x}_1$
X_1

Illustration of Mahalanobis Distance (Source)

Robust Mahalanobis Distance

- Replaces the covariate values with their **ranks** before using Mahalanobis distance.
- This makes the metric more robust against outliers (which could otherwise greatly affect the variance/covariance)

Greedy (nearest-neighbor) Matching

Steps:

1. Randomize the order of treated and control units.
2. Start with the first treated subject and match it to control units with the smallest distance.
3. Remove the match control units from the list of matches.
4. Move to the next treated subject and repeat the process until all treated subjects have been matched.

Greedy (nearest-neighbor) Matching

- This is "greedy" because we immediately match the current treated unit with the closest control.
- Advantages:
  - Intuitive
  - Computationally Inexpensive – this method can still be fast even for large data sets
- Disadvantages:
  - Matching varies depending on order of the training units.
  - The matching isn't optimal – it doesn't minimize the total distance. i.e. it is possible that another treatment unit down the line is a better match for the selected control.

Greedy (nearest-neighbor) Matching

- For many to one matching, just run the algorithm through the treated units k times. e.g. make sure that all treated units have 1 match before starting the second loop.
- We could set a maximum allowable distance for cases where there aren't any good matches. Is these cases, treated units with no close matches will be excluded.

Optimal Matching

- Minimized some *global* distance measure. e.g. Total Distance
- Computationally demanding – this is usually only possible for small data sets.
- Network flow optimization problem
- Sparse Optimal Matching
    - Do the optimal matching for certain features only. e.g. optimal matching per disease category, age group, gender, etc.
    - Aim for fine balance only.
- R packges:
    - optmatch
    - rcbalance

Matching in Practice

Matching Bias

- With matching, the assumption is

$$X_i \approx X_j \implies Y_i^0 \approx Y_j^0$$

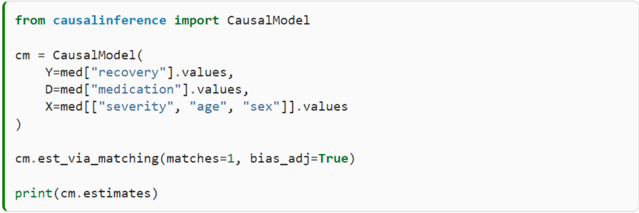
  where Y_j^0 is factual since it is from the control group.
- We say that Y_j^0 (the matching) is an unbiased estimator iff

$$\sqrt{N_{D=1}}(E[Y^0|D=1] - E[Y^0|D=0])$$

  converges to 0 as $N_{D=1} \to \infty$
- But this doesn't turn out to be the case. Here, $\sqrt{N_{D=1}}$ grows faster than $(E[Y^0|D=1] - E[Y^0|D=0])$ shrinks.
- Intuitively, increasing the treatment units makes it more likely to get good (closer) matches, but the overall decrease in distance doesn't converge fast enough.

Adjusting for Bias

- We won't go through it here, but adjusting for the matching bias involves estimating $E[Y|X, D = 0]$ using some model (e.g. linear regression on the control samples only).
- In practice, we can just use the library causalinference:

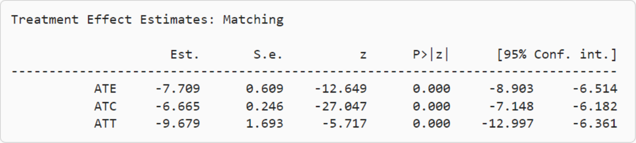


```python
from causalinference import CausalModel

cm = CausalModel(
    Y=med["recovery"].values,
    D=med["medication"].values,
    X=med[["severity", "age", "sex"]].values
)

cm.est_via_matching(matches=1, bias_adj=True)

print(cm.estimates)
```



```
Treatment Effect Estimates: Matching

                     Est.       S.e.          z       P>|z|      [95% Conf. int.]
--------------------------------------------------------------------------------
           ATE     -7.709      0.609    -12.649      0.000     -8.903     -6.514
           ATC     -6.665      0.246    -27.047      0.000     -7.148     -6.182
           ATT     -9.679      1.693     -5.717      0.000    -12.997     -6.361
```

Taken from *Causal Inference for the Brave and True*.

Propensity Score Matching

Motivation: The Curse of Dimensionality

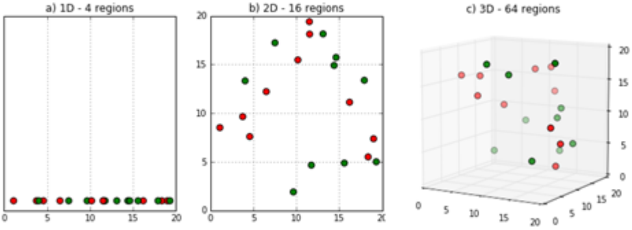


Image from DeepAI.

- Adding more covariates makes it (exponentially) more difficult to satisfy the *positivity assumption*.
- e.g. If you have 10 binary covariates, then there are 1024 possible states of X, and you need to make sure that you have a good sample size per state to find reasonable matches.

The Propensity Score

- The propensity score of an individual i is given by

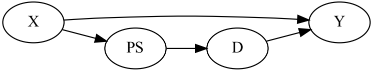
$$\pi_i = P(D = 1|X_i)$$

  i.e. it is the *probability of receiving treatment*.

- Instead of controlling for X, **it is sufficient to control for π to satisfy ignorability**. More formally,

$$(Y^1, Y^0) \perp\!\!\!\perp D|\pi(x)$$

- There's a formal way of proving this, but the intuition is:



- Notice that controlling for the Propensity Score (PS) is sufficient to satisfy the backdoor path criterion.

Estimated Propensity Score

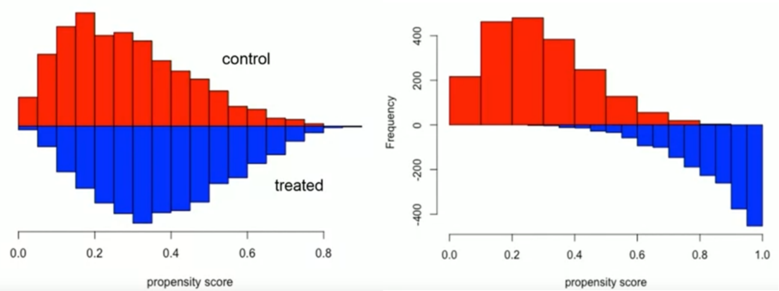
- Unless we're designing the experiment, we won't know the true propensity scores.
- So we have to resort to estimating $\pi(x) = P(D = 1|X)$.
- We can use any model to do this, but the most common way is to use **logistic regression**.

Propensity Score Matching (PSM)

- Propensity Score Matching (PSM) is just the same matching procedure, but we match on $\hat{\pi}(x)$ instead of on the covariates X.
- This makes the matching problem easier! It also gets around the curse of dimensionality problem.
- In practice, people usually match on the **logit** (log-odds) because it "stretches out" the distribution while preserving the rank. This is done because $0 <= \pi(x) <= 1$ making most values appear similar.

Assessing PSM Results

- We can easily check the Propensity Score distributions per treatment group to assess the quality of matching:



Taken from *A Crash Course in Causality*.

- The left figure shows good overlap while the right one shows poor overlap.

References

References

1. Hernan, Miquel A., and James M. Robins. Causal Inference. CRC Press, 2019.
2. Roy, Jason. “A Crash Course in Causality: Inferring Causal Effects from Observational Data — Coursera.” Coursera, https://www.coursera.org/learn/crash-course-in-causality. Accessed 15 Aug. 2022.
3. Matheus, Facure. “Causal Inference for The Brave and True.” Matheus Facure, https://matheusfacure.github.io/python-causality-handbook/landing-page.html. Accessed 15 Aug. 2022.
4. Morgan, Stephen L., and Christopher Winship. Counterfactuals and Causal Inference. Cambridge University Press, 2014.
5. Pearl, Judea, et al. Causal Inference in Statistics. John Wiley & Sons, 2016.
6. Yao, Liuyi, et al. “A Survey on Causal Inference.” ACM Transactions on Knowledge Discovery from Data, no. 5, Association for Computing Machinery (ACM), Oct. 2021, pp. 1–46. Crossref, doi:10.1145/3444944.