Matching directly on confounders

03.05.21 15:24

- we need to choose some metric of closeness e.g.
  a) mahalanobis distance
  b) robust mahalanobis distance

  to look at how similar two sets of covariates are from each other

Mahalanobis distance:

- X_j ... vector of covariates for subject j
- then the distance btw. two subjects i and j is:

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

Square-root of the vector product scaled by S^{-1}

↳ think of as "the square root of the sum of squared distances $(x_i - x_j)^2$ btw. each covariate x_i and x_j scaled by the covariate matrix $S = cov(X)$ where X is a random vector producing the samples X_i, X_j

→ multiplying $(X_i - X_j)^T (X_i - X_j)$ by S^{-1} allows us to scale each variable in X_i^k by its inverse-variance $var(X_i^k)$

Example using Mahalanobis distance:

- 3 covariates: X^1 = age, X^2 = COPD (yes = 1, no = 0), Female (yes = 1, no = 0)

Treated

Age	COPD	Female
78.17	0	1

Control

Age	COPD	Female	Computed distance
70.25	1	0	4.23
75.33	0	1	0.17
54.97	0	0	2.45
18.04	0	1	3.60

best match

Robust Mahalanobis distance:

- Motivation: Outliers can create large distances btw. 2 subjects i and j, even if their covariates are otherwise similar
- use ranks instead, to make the distance robust to outliers
  ↳ replace original values with ranks (e.g. X being age variable, then we give
  * rank = 1 for youngest subject &
  * rank = N for oldest subject in the dataset w/ N subjects

Method to compute Robust Mahalanobis distance:

(1) replace each covariate with its rank
(2) constant diagonal on covariance matrix S bc ranks should be on the same scale [1, N] now.
(3) compute the usual Mahalanobis distance on the ranks

Other distance measures:

- distance on propensity score
- if we need exact matches on some variables, then we could for example set $D = \infty$ if the variables' values don't match

Given that we have the distance measure, how do we select the matches?

(a) greedy (nearest neighbor) matching: → not as good but comp. fast

(b) optimal matching: → better but comp. demanding