Gov 2002: 6. Posttreatment Bias and Weighting

Matthew Blackwell

October 8, 2015

Propensity score weighting

Post-treatment bias

Where are we? Where are we going?

- Discussed randomized experiments, started talking about observational data.
- Last week: matching under no unmeasured confoudners.
- This week: propensity score weighting, posttreatment bias.
- Coming weeks: regression for causal inference, what happens when n.u.c. doesn't hold.

1/ Propensity score weighting

Weighting

- Next of the ways to estimate the ATE under no unmeasured confounders.
- **Intuition**
  - Treated and control samples are unrepresentative of the overall population.
  - Leads to imbalance in the covariates.
  - Reweight them to be more representative.

Survey samples

- Useful to review survey samples to understand the logic
- Finite population: $\{1, \ldots, N\}$
- Suppose that we wanted estimate the population mean of Y_i:

$$\bar{Y}_N = \frac{1}{N} \sum_{i=1}^{N} Y_i$$

- We have a sample of size n, where $Z_i = 1$ indicates that i is included in the sample.
- Unequal sampling probability: $\mathbb{P}(Z_i = 1) = \pi_i$
  - $\rightsquigarrow$ sample is not representative.
  - $\sum_{i=1}^{N} \pi_i = n$

Survey weights

- Sample mean is biased:

$$\mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{N} Z_i Y_i\right] = \frac{1}{n}\sum_{i=1} \pi_i Y_i$$

- **Inverse probability weighting**: To correct, weight each unit by the reciprocal of the probability of being included in the sample: Y_i/π_i.
- **Horvitz-Thompson estimator** is unbiased:

$$\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N} \frac{Z_i Y_i}{\pi_i}\right] = \frac{1}{N}\sum_{i=1}^{N} \frac{\mathbb{E}[Z_i] Y_i}{\pi_i} = \frac{1}{N}\sum_{i=1}^{N} \frac{\pi_i Y_i}{\pi_i} = \bar{Y}_N$$

- Reweights the sample to be representative of the population.

Back to causal effects

- With a completely randomized experiment, we can just use the simple differences in means:

$$\mathbb{E}[Y_i|D_i = 1] - \mathbb{E}[Y_i|D_i = 0] = \mathbb{E}[Y_i(1)] - \mathbb{E}[Y_i(0)]$$

- With no unmeasured confounders, we need to adjust for X_i.

$$\begin{aligned}
\mathbb{E}[Y_i(d)] &= \mathbb{E}\left[\mathbb{E}[Y_i(d)|X_i]\right] \\
&= \sum_{x \in \mathcal{X}} \mathbb{E}[Y_i(d)|X_i = x]\mathbb{P}(X_i = x) \\
&= \sum_{x \in \mathcal{X}} \mathbb{E}[Y_i(d)|D_i = d, X_i = x]\mathbb{P}(X_i = x) \\
&= \sum_{x \in \mathcal{X}} \mathbb{E}[Y_i|D_i = d, X_i = x]\mathbb{P}(X_i = x)
\end{aligned}$$

- With subclassification, we binned X_i, calclulated within-bin differences and then averaged across the bins, just like this.

Searching for the weights

$$\mathbb{E}[Y_i(d)] = \sum_{x \in \mathcal{X}} \mathbb{E}[Y_i | D_i = d, X_i = x] \mathbb{P}(X_i = x)$$

- Compare this to the the within treatment group average:

$$\begin{aligned} \mathbb{E}[Y_i | D_i = d] &= \sum_{x \in \mathcal{X}} \mathbb{E}[Y_i | D_i = d, X_i = x] \mathbb{P}(X_i = x | D_i = d) \\ &= \sum_{x \in \mathcal{X}} \mathbb{E}[Y_i | D_i = d, X_i = x] \frac{\mathbb{P}(D_i = d | X_i = x) \mathbb{P}(X_i = x)}{\mathbb{P}(D_i = d)} \end{aligned}$$

- How should we reweight the data from an observational study?
- If we were to reweight the data by $W_i = 1/\mathbb{P}(D_i = d | X_i)$, then we would break the relationship between D_i and X_i.

Weights

- Single binary covariate. Define the weight function:

$$w(d, x) = \frac{1}{e(x)^d (1 - e(x))^{1-d}}$$

- To get the weight for i, plug in observed treatment, covariate: $W_i = w(D_i, X_i)$
- If $(D_i, X_i) = (1, 1)$,

$$W_i = \frac{1}{e(1)} = \frac{1}{\mathbb{P}(D_i = 1 | X_i = 1)}$$

- If $(D_i, X_i) = (0, 0)$:

$$W_i = \frac{1}{1 - e(0)} = \frac{1}{\mathbb{P}(D_i = 0 | X_i = 0)}$$

Example

	$X_i = 0$	$X_i = 1$
$D_i = 0$	4	3
$D_i = 1$	4	9

- $\mathbb{P}(D_i = 1 | X_i = 0) = 0.5$
- $\mathbb{P}(D_i = 1 | X_i = 1) = 0.75$
- Weights:

	$X_i = 0$	$X_i = 1$
$D_i = 0$	$1/0.5$	$1/0.25$
$D_i = 1$	$1/0.5$	$1/0.75$

- Weighted data (the pseudo-population):

	$X_i = 0$	$X_i = 1$
$D_i = 0$	8	12
$D_i = 1$	8	12

- $\mathbb{P}_W(D_i = 1 | X_i = x) = 0.5$ for all x

Properties of reweighted data

- Let's calculate the **weighted probability** that $D_i = 1$.

$$\begin{aligned}
&\mathbb{P}_W[D_i = 1 | X_i = x] \\
&= \frac{w(1,x) \cdot \mathbb{P}[D_i = 1 | X_i = x]}{\omega^*} \\
&= \frac{\frac{1}{\mathbb{P}[D_i = 1 | X_i = x]} \cdot \mathbb{P}[D_i = 1 | X_i = x]}{\omega^*} \\
&= \frac{1}{\omega^*}.
\end{aligned}$$

- ω^* is a normalization factor to make sure probabilities sum to 1.
- Important point: $\mathbb{P}_W(D_i = 1 | X_i = 1) = \mathbb{P}_W(D_i = 1 | X_i = 0) = \frac{1}{\omega^*}$
- $\rightsquigarrow$ D_i independent of X_i in the reweighted data.

Overall mean

- What is the weighted mean for the treated group?
- Use a similar approach to survey weights, where D_i is the "sampling indicator":

$$\bar{Y}_i^w = \frac{1}{N}\sum_{i=1}^{N} D_i W_i Y_i$$

- $W_i Y_i$ is the weighted outcome, D_i is there to select out the treated observations.
- We want to see what the conditional weighted mean identifies:

$$\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N} W_i D_i Y_i\right] = \frac{1}{N}\sum_{i=1}^{N}\mathbb{E}[W_i D_i Y_i] = \mathbb{E}[W_i D_i Y_i]$$

Proving unbiasedness

- Weighted mean of treated units is mean of potential outcome:

$$
\begin{aligned}
\mathbb{E}[W_i D_i Y_i] &= \mathbb{E}\left[\frac{D_i Y_i}{e(X_i)}\right] && \text{(Weight Def.)} \\
&= E\left[\frac{D_i Y_i(1)}{e(X_i)}\right] && \text{(Consistency)} \\
&= E\left[E\left[\frac{D_i Y_i(1)}{e(X_i)}\middle| X_i\right]\right] && \text{(Iterated Expectations)} \\
&= E\left[\frac{E[D_i|X_i]E[Y_i(1)|X_i]}{e(X_i)}\right] && \text{(n.u.c.)} \\
&= E\left[\frac{e(X_i)E[Y_i(1)|X_i]}{e(X_i)}\right] && \text{(Propensity Score Definition)} \\
&= E[Y_i(1)] && \text{(Iterated Expectations)}
\end{aligned}
$$

Putting it all together

- The same logic would give us the mean potential outcomes under control:
$$E\left[\frac{(1-D_i)Y_i}{1-e(X_i)}\right] = E[Y_i(0)]$$
- These two facts provide an estimator for the average treatment effect:

$$\hat{\tau} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{D_iY_i}{e(X_i)} - \frac{(1-D_i)Y_i}{1-e(X_i)}\right)$$

- The above two results give us that this esimator is unbiased.
- This is sometimes called the **Horvitz-Thompson** estimator due to the close connection to the survey sampling estimator.

Estimation of the propensity score

$$\hat{\tau} = \frac{1}{N} \sum_{i=1}^{N} \left(\frac{D_i Y_i}{e(X_i)} - \frac{(1 - D_i) Y_i}{1 - e(X_i)} \right)$$

- Need to know or estimate the propensity score, $e(X_i)$. How do we do that?
- **Discrete covariates** estimate the within-strata propensity scores
  $$\hat{e}(x) = \frac{N_{xd}}{N_x}$$
  - Non-parametric estimate of the propensity score in each stratum of the data.
- **Continuous covariates** $\rightsquigarrow$ Logistic regression of D_i on X_i.

Estimated versus known pscores

```r
ht.est <- function(y, d, w) {
    n <- length(y)
    (1/n) * sum((y * d * w) - (y * (1 - d) * w))
}
n <- 200
x <- rbinom(n, size = 1, prob = 0.5)
dprobs <- 0.5 * x + 0.4 * (1 - x)
d <- rbinom(n, size = 1, prob = dprobs)
y <- 5 * d - 10 * x + rnorm(n, sd = 5)

true.w <- ifelse(d == 1, 1/dprobs, 1/(1 - dprobs))
pprobs <- predict(glm(d ~ x))
est.w <- ifelse(d == 1, 1/pprobs, 1/(1 - pprobs))
ht.est(y, d, est.w)
```

```
## [1] 5.1
```

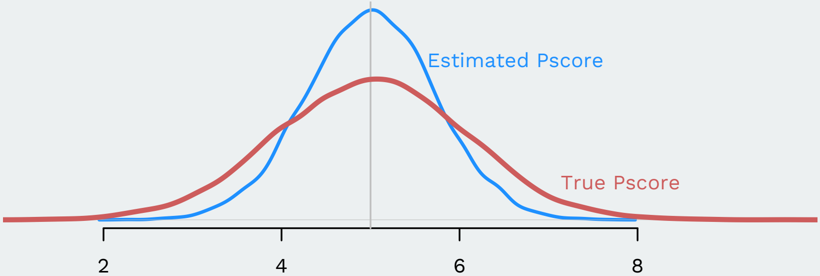
```r
ht.est(y, d, true.w)
```

```
## [1] 5.5
```

Sampling distribution of the HT estimators

```r
sims <- 10000
true.holder <- rep(NA, sims)
est.holder <- rep(NA, sims)
for (i in 1:sims) {
    x <- rbinom(n, size = 1, prob = 0.5)
    dprobs <- 0.5 * x + 0.4 * (1 - x)
    d <- rbinom(n, size = 1, prob = dprobs)

    y <- 5 * d - 10 * x + rnorm(n, sd = 5)
    true.w <- ifelse(d == 1, 1/dprobs, 1/(1 - dprobs))
    pprobs <- predict(glm(d ~ x))
    est.w <- ifelse(d == 1, 1/pprobs, 1/(1 - pprobs))
    est.holder[i] <- ht.est(y, d, est.w)
    true.holder[i] <- ht.est(y, d, true.w)
}
```

Sampling distribution of the HT estimators



```
var(est.holder)
```

```
## [1] 0.52
```

```
var(true.holder)
```

```
## [1] 1.2
```

Why use estimated pscores?

- Why does the estimated propensity score do better than the true propensity score?
- **Removing chance variations** using $\hat{e}(X_i)$ adjusts for any small imbalances that arise because of a finite sample.
- The true p-score only adjusts for the **expected** differences between samples.

Distribution of X in the weighed data

```
ht.est(x, d, est.w)
```

```
## [1] 8.1e-16
```

```
ht.est(x, d, true.w)
```

```
## [1] -0.2
```

Positivity violations

- Remember the positivity assumption:

$$0 < p(D_i = 1|X_i) < 1$$

- What happens to the weights if this is violated? Then, $\hat{e}(x) = 0$ or $\hat{e}(x) = 1$ and

$$\frac{1}{\hat{e}(x)} = \frac{1}{0} = \infty$$

- **Structural** $\rightsquigarrow$ population probability is 0.
- **Random** $\rightsquigarrow$ sample probability is 0.
  - Need to “borrow” information from other values of X_i to estimate $e(X_i)$
  - $\rightsquigarrow$ modeling via logit, etc.

Automated approaches

- Challenge: specifying the propensity score model.

$$\hat{e}(X_i) = \text{logit}^{-1}(X_i'\beta)$$

- What terms should we include?
- Big problem for weights: small changes to PS model lead to big changes in the weights.
- Entropy balancing (Hainmueller 2012):
  - Choose weights for each observation that maximize the balance between treatment and control groups.
- Covariate Balancing Propensity Scores (Imai and Ratkovic):
  - Estimate the propensity score subject to the additional constraint of maximizing balance.

Boostrapping to get the SEs

- How to get the standard error for $\hat{\tau}$?
- Variance estimators are messy $\rightsquigarrow$ use the bootstrap!

  1. Draw a sample of the data with replacement, call this, S_b.
  2. Estimate the propensity scores in this sample, $\hat{e}_b$ and create weights, W_b.
  3. Use the weights to get an estimate of the average treatment effect, τ_b in the sample S_b.
  4. Repeat.

- The distribution of the estimates, $\hat{\tau}_b$, will give us the bootstrapped standard errors and confidence intervals.

Bootstrap in R

```
mydata <- data.frame(y, d, x)
boots <- 1000
b.holder <- rep(NA)
for (i in 1:boots) {
    S.b <- sample(1:n, size = n, replace = TRUE)
    data.b <- mydata[S.b, ]
    pprobs <- predict(glm(d ~ x, data = data.b))
    est.w <- ifelse(data.b$d == 1, 1/pprobs, 1/(1 -
        pprobs))
    b.holder[i] <- ht.est(data.b$y, data.b$d, est.w)
}
```

- Compare bootstrapped variance to true sampling variance:

```
var(b.holder)
```

```
## [1] 0.51
```

```
var(est.holder)
```

```
## [1] 0.52
```

Reducing weight variation

- $e(X_i)$ close to 0 or 1 lead to very large weights, high standard errors.
- Potential solutions:

1. **Trimming/Windsorizing the weights**
   - Pick some value w' and create trimmed weights which are:

$$W_i' = \begin{cases} W_i & \text{if } W_i < w' \\ w' & \text{if } W_i \geq w' \end{cases}$$

2. **Stabilized weights**
   - We can actually put any other function of the treatment vector in the numerator, which can reduce the variation in the weights.
   - We call these stabilized weights:

$$sw(d, x) = \frac{\mathbb{P}[D_i = 1]^d (1 - \mathbb{P}[D_i = 1])^{1-d}}{e(x)^d (1 - e(x))^{1-d}}$$

Stablized weights

- With a binary treatment, we can implement the stabilized weight by normalizing the weights:

$$SW_i = \frac{W_i}{\sum_{i=1}^{N} W_i}$$

- This leads to the following estimator:

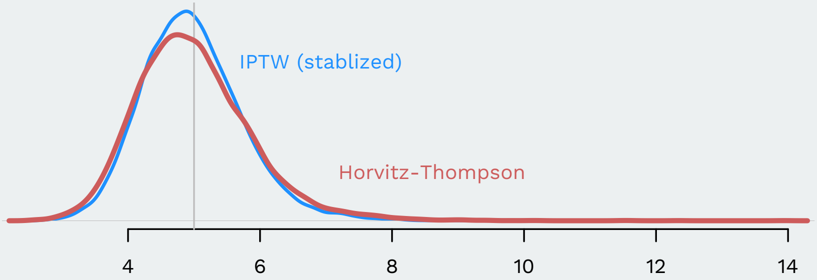
$$\begin{aligned}\hat{\tau}_{IPTW} =& \frac{1}{\sum_{i=1}^{N} W_i D_i} \sum_{i=1}^{N} W_i D_i Y_i - \frac{1}{\sum_{i=1}^{N} W_i (1 - D_i)} \sum_{i=1}^{N} W_i (1 - D_i) Y_i \\ =& \frac{1}{\sum_{i=1}^{N} D_i / \hat{e}(X_i)} \sum_{i=1}^{N} \frac{D_i Y_i}{\hat{e}(X_i)} \\ & - \frac{1}{\sum_{i=1}^{N} (1 - D_i)/(1 - \hat{e}(X_i))} \sum_{i=1}^{N} \frac{(1 - D_i) Y_i}{1 - \hat{e}(X_i)}\end{aligned}$$

- These are the means that the `weighted.mean()` function in R calculates. It normalizes the weights before calculating the mean.

Stablized weights

```
n <- 1000
sims <- 10000
est2.holder <- rep(NA, sims)
sw.holder <- rep(NA, sims)
for (i in 1:sims) {
    x <- rnorm(n)
    dprobs <- boot::inv.logit(-1 + x)
    d <- rbinom(n, size = 1, prob = dprobs)
    y <- 5 * d - 10 * x + rnorm(n, sd = 5)

    pprobs <- glm(d ~ x, family = binomial())$fitted
    est.w <- ifelse(d == 1, 1/pprobs, 1/(1 - pprobs))
    est2.holder[i] <- ht.est(y, d, est.w)
    sw.holder[i] <- weighted.mean(y[d == 1], est.w[d ==
        1]) - weighted.mean(y[d == 0], est.w[d == 0])
}
```

Stabilized weights



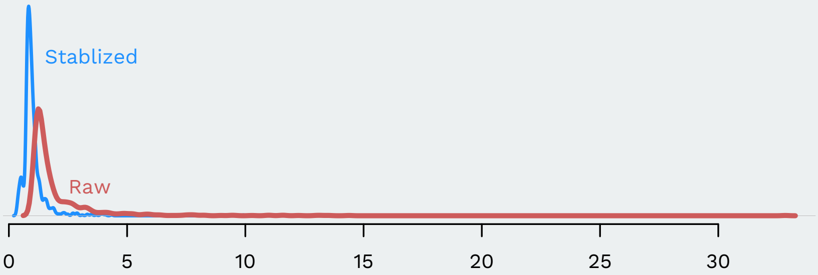
```r
var(est2.holder)
```

```
## [1] 0.78
```

```r
var(sw.holder)
```

```
## [1] 0.59
```

Distribution of the weights



```
tail(est.w[order(est.w)])
```

```
## [1] 12 13 13 14 14 33
```

```
tail(est.sw[order(est.sw)])
```

```
## [1] 3.9 3.9 4.0 4.1 4.3 9.9
```

2/ Post-treatment bias

Post-treatment bias

- Rule of matching/weighting/regression: **don't condition on posttreatment variables**.
- Usual intuition:
  - You might "control away" part of the effect of D_i on Y_i that "flows through" Z_i where Z_i is the posttreatment variable.
  - Can be misleading.
- Two big problems with conditioning on these:
  - Changes the quantity of interest (see above).
  - Induces selection bias.
- We'll go through Rosenbaum (1984) logic.

Setup

- Posttreatment variable Z_i
- Has potential outcomes because it is affected by treatment: $(Z_i(1), Z_i(0))$.
- Consistency for the posttreatment variable:

$$Z_i = D_i Z_i(1) + (1 - D_i) Z_i(0)$$

- Example:
  - Effect of campaign negativity (D_i) fixing polling later in the campaign (Z_i)

Assumptions and estimators

- Assume no unmeasured confounders:

$$\Big(Y_i(1), Y_i(0)\Big) \perp\!\!\!\perp D_i | X_i$$

- Usually estimate the CATE:

$$\tau(x) = E[Y_i | D_i = 1, X_i = x] - E[Y_i | D_i = 0, X_i = x]$$

- Average to get the ATE: $\tau = E[\tau(X_i)]$.

Condition on a posttreatment variable

- What happens when we control for the post-treatment variable:

$$\begin{aligned}\Delta(x,z) &= E[Y_i|D_i = 1, Z_i = z, X_i = x] - E[Y_i|D_i = 0, Z_i = z, X_i = x] \\ &= E[Y_i(1)|D_i = 1, Z_i = z, X_i = x] - E[Y_i(0)|D_i = 0, Z_i = z, X_i = x] \\ &= E[Y_i(1)|D_i = 1, Z_i(1) = z, X_i = x] - E[Y_i(0)|D_i = 0, Z_i(0) = z, X_i = x]\end{aligned}$$

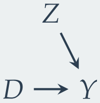
- Average these over the distribution of (X, Z): $\Delta = E[\Delta(X, Z)]$.
- Compare this estimator Δ to the average treatment effect τ.

Controlled direct effect

- Define the **net treatment difference** $\nu(x,z)$:

$$\nu(x,z) = E[Y_i(1)|Z_i(1) = z, X_i = x] - E[Y_i(0)|Z_i(0) = z, X_i = x]$$

- Similar to the **controlled direct effect**, or the effect of D_i fixing $Z_i(1) = Z_i(0) = z$, removing the arrow from D_i to Z_i:



- Intuitively (if not precisely): if $\nu(x,z) = 0$ and $\tau > 0$, the effect of D_i on Y_i flows entirely through Z_i.
- Again, we'll take the average over (X_i, Z_i): $\nu = E[\nu(X_i, Z_i)]$.

Posttreatment bias decomposition

$$\Delta - \tau = \underbrace{(\Delta - \nu)}_{\text{bias for NTD}} + \underbrace{(\nu - \tau)}_{\text{change in QoI}}$$

- The bias of Δ is two terms.
- $(\Delta - \nu)$ measures our inability to estimate the net treatment difference.
- Why? Maybe Z_i is a collider. If we condition on Z_i, it opens a backdoor path between D_i and Y_i:

$$\begin{tikzcd}
U \arrow[r] \arrow[d] & Z \\
D \arrow[r] \arrow[ur] & Y \arrow[u]
\end{tikzcd}$$

- In this case, conditioning on Z opens the backdoor path from $D \leftarrow U \rightarrow Z \leftarrow Y$. Thus, $(\Delta - \nu)$ represents the bias due to unmeasured confounding between D_i and Z_i.

Posttreatment bias

$$\Delta - \tau = \underbrace{(\Delta - \nu)}_{\text{bias for NTD}} + \underbrace{(\nu - \tau)}_{\text{change in QoI}}$$

- $(\nu - \tau)$: difference between the net treatment difference and the average treatment effect.
- The change in the quantity of interest.
- Might call this the effect of intervening on Z_i.
- Under some conditions, this difference can be thought of as the indirect effect of D_i on Y_i through Z_i, but not always.
  - $\rightsquigarrow$ Causal mediation/mechanisms
  - Very tricky assumptions, we'll talk about later.

Conditions that eliminate post-treatment bias

- When will there be no posttreatment bias?
- Under two assumptions:

1. **No unmeasured confounders for post-treatment variable**:

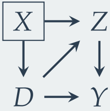
$$(Y_i(0), Z_i(0), Y_i(1), Z_i(1)) \perp\!\!\!\perp D_i | X_i$$

2. **No effect of treatment on the post-treatment variable**: $Z_i(1) = Z_i(0) = Z_i$ for all units.

No unmeasured confounders, II

$$\big(Y_i(0), Z_i(0), Y_i(1), Z_i(1)\big) \perp\!\!\!\perp D_i | X_i$$

- This extends no unmeasured confounders to the post-treatment variable.
- Most likely satisfied under randomization.
- Implies that $\Delta = \nu$. Why?
  - No unblocked backdoor paths from D_i to Z_i
  - $\rightsquigarrow Z_i$ cannot be a collider on a back-door path.
  - No collider bias for NTD
- Still could change the quantity of interest.

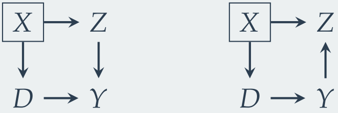


No effect on Z

- **No effect of treatment on the post-treatment variable**: $Z_i(1) = Z_i(0) = Z_i$ for all units.
- Under this condition, we have NTD = ATE.
  - The effect of D_i cannot go through Z_i since it doesn't affect Z_i:

$$\begin{aligned} \nu(x,z) &= \mathbb{E}[Y(1)|Z(1) = z, X = x] - \mathbb{E}[Y(0)|Z(0) = z, X = x] \\ &= \mathbb{E}[Y(1) - Y(0)|Z = z, X = x]. \end{aligned}$$

- So that when we take the average over (X_i, Z_i), we get $\nu = \tau$. In this case the above DAGs would be:

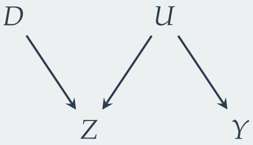


- Essentially assumes Z_i is pretreatment.

Posttreatment bias overview

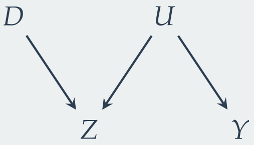
- Found two assumptions under which condition on Z_i doesn't matter.
- But, these two assumptions buy us nothing:
  - Requires no unmeasured confounders $\rightsquigarrow$ could have estimated the ATE in the usual way.

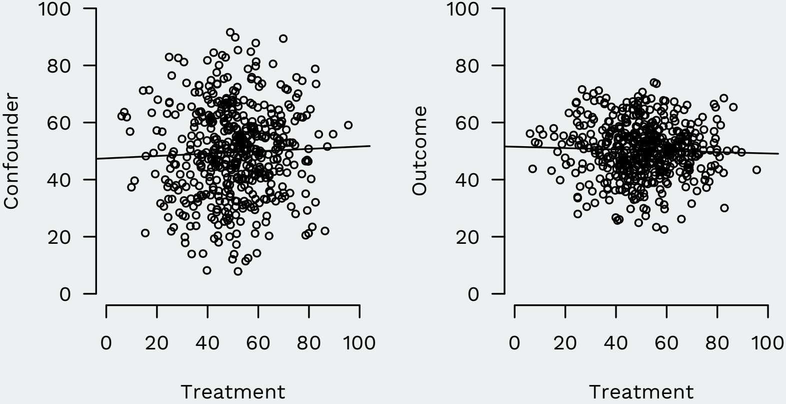
Simulation



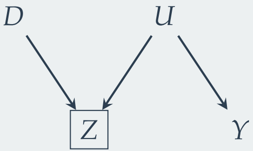
```
## Post-treatment bias simulation
set.seed(14627)
d <- rnorm(500, 50, 15)
u <- rnorm(500, 50, 15)
z <- rnorm(500, 0.5 * d + 0.5 * u, 5)
y <- rnorm(500, 75 + -0.5 * u, 5)

sub <- z > 60 & z < 70
```

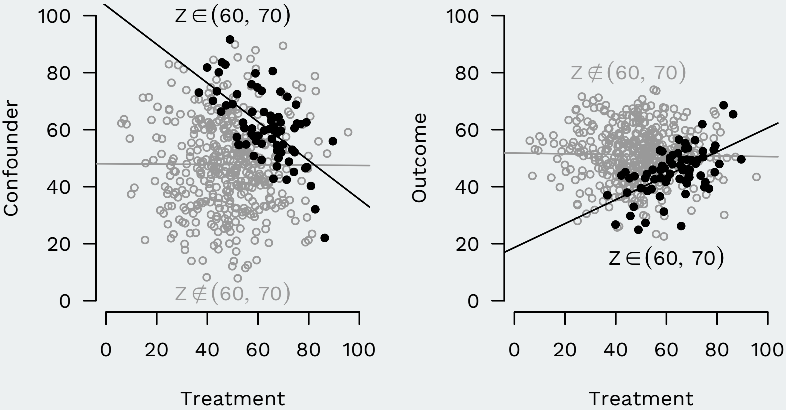
Posttreatment bias example





Posttreatment bias example




Confounder
Z ∈ (60, 70)
Z ∉ (60, 70)
Treatment
Outcome
Z ∉ (60, 70)
Z ∈ (60, 70)
Treatment