Causal Inference Using Bayesian Methods

Hun Lee

Goal

Our initial goal is to impute the missing varialbe of X (1: positive for marijuana blood test, 0: negative) by using a proxy variable W (self-reported marijuana use) and vice versa, and the ultimate goal is to estimate the population average treatment effect of X on Y (1: homicide victim, 0: not a victim). Note that the three data sets are observational data. The imputation is done by estimating a latent variable using probit regression. For the estimatation of the causal effect, the g-formula (standardization) is going to be used through the Bayesian Bootstrap. All of our methods are done in Stan programming.

Generated Data Feature

Y: Homicide victim (outcome variable) - binary

X: Marijuana Blood test (exposure variable) - binary

W: Self-reported marijuana use (misclassified exposure variable) - binary

Z: Confounding Covariates (Demogrpahic variables) - discrete

NRS: Y, X, W, Z

NVDRS: Y, X, Z (W missing)

NSDUH: Y, W, Z (X missing)

Note that set.seed(1) is consistently used. Different seed values may or may not give different results.

Assumptions for causal inference

Assumption 1 (positivity).

There exist $\delta \in (0, 1)$ such that $0 < \delta < f(X|Z) < 1 - \delta < 1$

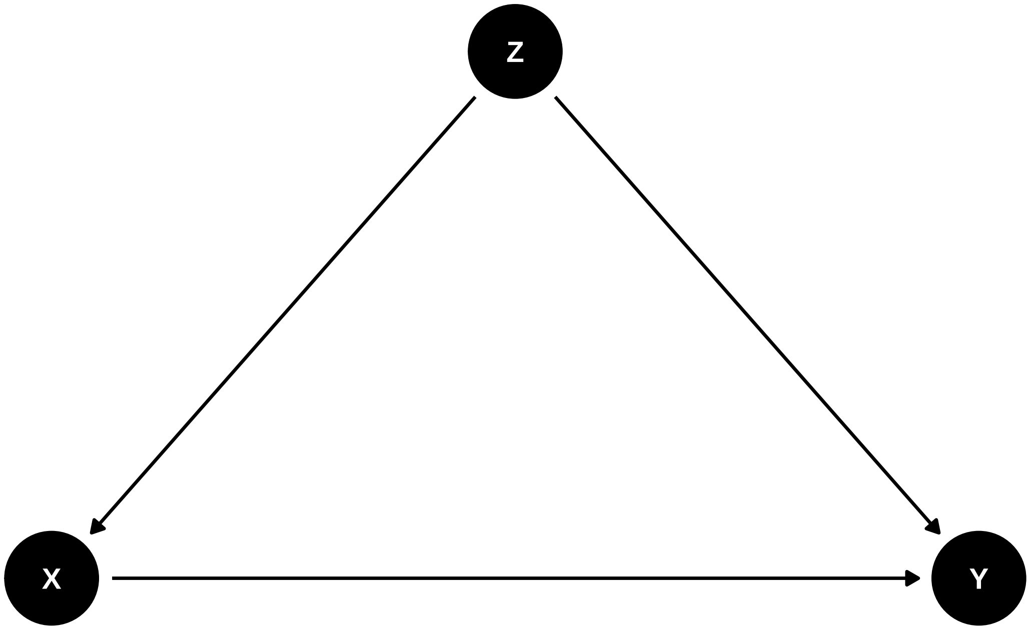
Assumption 2 (consistency).

If X = x, then Y(x) = Y with probability 1.

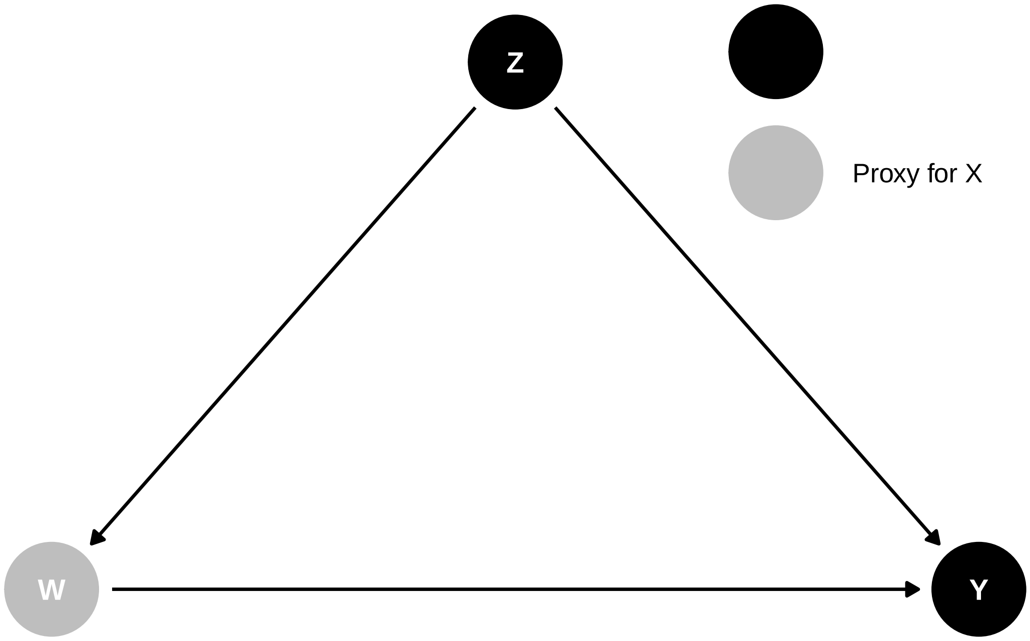
Assumption 3 (no unmeasured confounding between exposure and outcome).

$Y(x) \perp\!\!\!\perp X|Z$

Causal DAG with X



Causal DAG with W



Original X vs W Contingency Table

```
##    X
## W      0    1
##   0 5039  245
##   1  335  381
```

Stan GLM with original full data (true coefficient values: [-2, 1, 0.2, 0.2])

```
##              mean  mcse   sd    10%   50%   90% n_eff Rhat
## (Intercept) -1.98 0.0012 0.060 -2.058 -1.98 -1.90  2565    1
## X            0.99 0.0021 0.097  0.869  0.99  1.12  2178    1
## Z1           0.16 0.0017 0.078  0.061  0.16  0.26  2011    1
## Z2           0.18 0.0018 0.077  0.087  0.18  0.28  1886    1
```

With additional proxy variable W (true coefficient values: [-2, 1, 0.2, 0.2, 0])

```
##                mean    mcse     sd     10%     50%    90% n_eff Rhat
## (Intercept) -1.9806 0.00121 0.0605 -2.0587 -1.9810 -1.901  2519    1
## X            0.9808 0.00304 0.1156  0.8307  0.9822  1.129  1444    1
## Z1           0.1598 0.00190 0.0776  0.0560  0.1621  0.260  1672    1
## Z2           0.1808 0.00184 0.0774  0.0804  0.1806  0.282  1767    1
## W            0.0166 0.00305 0.1173 -0.1319  0.0151  0.171  1481    1
```

Imputing W in NVDRS using NRS data

Misclassificaiton probit model

```
## Inference for Stan model: c5e76aef3b559b9a3f8d7044d0724a1f.
## 2 chains, each with iter=2000; warmup=1000; thin=1;
## post-warmup draws per chain=1000, total post-warmup draws=2000.
##
##     mean se_mean   sd  2.5%   25%   50%   75% 97.5% n_eff Rhat
## a  -1.57       0 0.05 -1.68 -1.61 -1.57 -1.54 -1.47   954    1
## a1  1.67       0 0.09  1.50  1.62  1.67  1.72  1.84  1018    1
## a2  0.00       0 0.07 -0.14 -0.06  0.00  0.05  0.13   829    1
## a3  0.14       0 0.07  0.00  0.09  0.14  0.19  0.28   908    1
##
## Samples were drawn using NUTS(diag_e) at Sat Aug 13 17:59:57 2022.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

Table 1: Misclassification Model True Coefficients

a	a1	a2	a3
-1.7	1.8	0.1	0.2

True W vs Imputed W contingency table

```
##          True_W
## Imputed_W   0   1
##         0 713  45
##         1  74 128
```

Table 2: Imputation Accuracy

88%

Imputing X in NSDUH using NRS and NVDRS data sets

Exposure probit model

```
## Inference for Stan model: 86c95029a87333345e5183425e41e69f.
## 2 chains, each with iter=2000; warmup=1000; thin=1;
## post-warmup draws per chain=1000, total post-warmup draws=2000.
##
##      mean se_mean   sd  2.5%   25%   50%   75% 97.5% n_eff Rhat
## b  -2.12    0.00 0.07 -2.26 -2.17 -2.12 -2.07 -1.97   375 1.01
## b1  0.20    0.01 0.12 -0.02  0.12  0.20  0.28  0.42   480 1.01
## b2  0.16    0.01 0.11 -0.06  0.09  0.16  0.24  0.38   419 1.01
## b3  0.07    0.01 0.16 -0.24 -0.03  0.07  0.18  0.39   480 1.01
## b4  2.22    0.00 0.07  2.08  2.17  2.22  2.27  2.37  1002 1.01
##
## Samples were drawn using NUTS(diag_e) at Sat Aug 13 18:12:27 2022.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

Table 3: Blood Model True Coefficients

b	b1	b2	b3
-1.7	0.2	0.3	0.3

True X vs Imputed X contingency table

```
##           True_X
## Imputed_X    0    1
##         0 1730   68
##         1  104  114
```

Table 4: Imputation Accuracy

91%

Final X vs W table after imputation

```
##    X
## W      0    1
##   0 5152  103
##   1  186  559
```

Outcome Logistic Model

```
## Inference for Stan model: 166aed1770e8dace5bb9ee98e29b03b4.
## 2 chains, each with iter=2000; warmup=1000; thin=1;
## post-warmup draws per chain=1000, total post-warmup draws=2000.
##
##     mean se_mean   sd  2.5%   25%   50%   75% 97.5% n_eff Rhat
## c  -1.99       0 0.06 -2.11 -2.03 -1.99 -1.95 -1.87  1311    1
## c1  0.91       0 0.10  0.72  0.85  0.91  0.97  1.10  1648    1
## c2  0.18       0 0.08  0.03  0.13  0.19  0.24  0.33  1363    1
## c3  0.19       0 0.08  0.03  0.14  0.19  0.24  0.34  1603    1
##
## Samples were drawn using NUTS(diag_e) at Sat Aug 13 18:16:03 2022.
## For each parameter, n_eff is a crude measure of effective sample size,
## and Rhat is the potential scale reduction factor on split chains (at
## convergence, Rhat=1).
```

Table 5: Outcome Model True Coefficients

c	c1	c2	c3
-2	1	0.2	0.2

Stan GLM with original full data

```
##               mean   mcse    sd    10%   50%   90% n_eff Rhat
## (Intercept) -1.98 0.0012 0.060 -2.058 -1.98 -1.90  2565    1
## X            0.99 0.0021 0.097  0.869  0.99  1.12  2178    1
## Z1           0.16 0.0017 0.078  0.061  0.16  0.26  2011    1
## Z2           0.18 0.0018 0.077  0.087  0.18  0.28  1886    1
```

It is to be observed the estimated coefficients in the outcome model after imputation is not far from the true coefficients used for data simulation. Now, let's proceed to use this imputed data to estimate the average treatment using standardization and the Bayesian bootstrapping to take into account the combination of covariates that are more likely to be resampled with replacement.

Causal Inference

Estimating the True Population Average Treatment Effect on Y

Table 6: True ATE

0.168

Estimating ATE of X on Y using Standardization (g-formula)

```
##          mean      se_mean         sd      2.5%       25%    50%       75%
## ATE 0.1497856 0.0004862832 0.01977635 0.1114958 0.1368333 0.1495 0.1626667
##         97.5%    n_eff     Rhat
## ATE 0.1898375 1653.916 1.000483
```

Estimating ATE of W on Y using Standardization (g-formula)

```
##          mean      se_mean         sd   2.5%       25%       50%       75%
## ATE 0.1151402 0.0004527707 0.01813712 0.0805 0.1028333 0.1151667 0.1271667
##         97.5%    n_eff     Rhat
## ATE 0.1513375 1604.648 1.000249
```

As expected, the estimation of the population average treatment effect with imputed data is biased. It is also to be observed there is a huge difference between ATE of X and ATE of W. This suggests that the self-reported variable W is not a good proxy for X for a causal estimation.

How to improve the accruacy of ATE estimation?

Population average treatment effect estimates the counterfactual outcome, "everyone treated vs everyone untreated." In this counterfactual world, it is also to be assumed that there is no misclassification. Namely, everyone's self-reported status also should align with their treatment status, that is X = W. Thus, X will only contribute to estimating the outcome Y, and the coefficient of W is expected to be 0 even if we use both variables in the outcome model. However, that's not the case with our imperfect data and models as well as imperfect imputation. However, we can use this assumption and the fact that W is a proxy for X in order to reduce bias in ATE estimator. To this end, we fit two models: an outcome model without W and and an outcome model with W.

$$Y \sim a_0 + a_1 X + a_2 Z$$

$$Y \sim b_0 + b_1 X + b_2 Z + b_3 W$$

Since the imputed X variable is still somewhat misclassified, its coefficient should be also biased and hence the proxy variable W should have non-zero coefficient in the outcome model to compensate the biased coefficient of X. Thus, we use the following model to estimate the population average treatment effect:

$$Y \sim a_0 + a_1 X + a_2 Z + b_3 W$$

Thus,

Y^1[n] = bernoulli_logit_rng(Z[bootstrap_i] * alphaZ + alphaA + betaW);

Y^0[n] = bernoulli_logit_rng(Z[bootstrap_i] * alphaZ);

With the imputed data adding W

```
##          mean      se_mean        sd     2.5%       25%       50%       75%
## ATE 0.1706713 0.0008634187 0.0388037 0.096825 0.1443333 0.1690833 0.1968333
##     97.5%    n_eff      Rhat
## ATE 0.248 2019.777 0.9991521
```

With the original data adding W

```
##          mean     se_mean         sd      2.5%       25%       50%   75%
## ATE 0.1709071 0.000715881 0.03424504 0.1053333 0.1477917 0.1699167 0.194
##         97.5%    n_eff      Rhat
## ATE 0.2405042 2288.304 0.9991558
```

With the original data without W

```
##          mean      se_mean         sd      2.5%       25%       50%       75%
## ATE 0.1654007 0.0004861023 0.02039699 0.1253333 0.1513333 0.1653333 0.1793333
##         97.5%    n_eff     Rhat
## ATE 0.2026667 1760.666 1.000849
```

Adding W variable to the model reduces bias in the population ATE estimation. In fact, this method results in less biased estimation than when using the full original data without W. It also appears that the posterior median estimator for ATE is less biased than the posterior mean estimator. One thing to note is that the fact that this ad hoc approach works here doesn't mean it would work under other circumstancese.