# Propensity Score Weighting

## using machine learning

Young Geun Kim
ygeunkim.github.io

2019711358, Department of Statistics

03 Dec, 2020

Introduction

Evaluation

Related Contents

# Introduction

# Reviewed Paper

### Estimation

Reviewed and apply Lee et al. (2010): estimate propensity score using

- Logistic regression: `glm()`
- Random forests: `randomForest::randomForest()`
- SVM (Pirracchio et al., 2014): `e1071::svm()`

### Evaluation

- Average standardized absolute mean distance
- Emprical distribution of IPTW
- IPW and SIPW

# Custom Package

```r
# remotes::install_github("ygeunkim/propensityml")
library(propensityml)
```

# Simulation study

Simulation setting by Setoguchi et al. (2008):

- 10 covariates: confounders, exposure predictors, outcome predictors
- Treatment, (true) treatment probability
- Continuous outcome

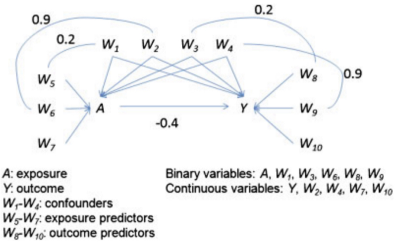


**Figure 1:** Simulation Data

# Correlation Matrix

of covariates:

$$
\begin{bmatrix}
1 & 0 & 0 & 0 & 0.2 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0.9 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0.2 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0.9 & 0 \\
0.2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.9 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0.2 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0.9 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
$$

# Scenarios

1. Additivity and linearity:

$$P(Z=1 \mid X_i) = \frac{1}{1+\exp(-(\beta_0+\beta_1 X_1+\cdots+\beta_7 X_7))}$$

2. Moderate non-linearity: *3 quadratic term*

$$P(Z=1 \mid X_i) = \frac{1}{1+\exp(-(\beta_0+\beta_1 X_1+\cdots+\beta_7 X_7+\beta_2 X_2^2))}$$

3. Moderate non-linearity: *10 two-way interaction terms*
4. Moderate non-additivity and non-linearity: *10 two-way interaction terms and 3 quadratic terms*

Here,

$$(\beta_0, \beta_1, \ldots, \beta_7)^T = (0, 0.8, -0.25, 0.6, -0.4, -0.8, -0.5, 0.7)^T$$

# Outcome

$$Y = \alpha_0 + \alpha_1 X_1 + \cdots + \alpha_4 X_4 + \alpha_5 X_8 + \cdots + \alpha_7 X_{10} + \gamma Z$$

where

- $(\alpha_0, \alpha_1, \ldots, \alpha_7)^T = (-3.85, 0.3, -0.36, -73, -0.2, 0.71, -0.19, 0.26)^T$
- $\gamma = -0.4$: True effect

# Function to reproduce Setoguchi et al. (2008)

```
sim_outcome(n = 1000, covmat = build_covariate()) %>%
  glimpse(width = 50)
#> Rows: 1,000
#> Columns: 13
#> $ w1           <fct> 0, 1, 1, 1, 0, 1, 1, 1,...
#> $ w2           <dbl> -0.2801, 0.3065, 0.6329...
#> $ w3           <fct> 0, 0, 0, 1, 1, 1, 1, 1,...
#> $ w4           <dbl> 1.6575, -1.4404, -1.939...
#> $ w5           <fct> 1, 1, 1, 0, 0, 1, 0, 0,...
#> $ w6           <fct> 0, 1, 1, 0, 0, 1, 1, 0,...
#> $ w7           <dbl> 0.4874, -0.0162, -0.155...
#> $ w8           <fct> 1, 1, 0, 0, 1, 0, 1, 1,...
#> $ w9           <fct> 1, 0, 0, 1, 1, 0, 1, 0,...
#> $ w10          <dbl> -0.3054, 0.5939, 0.4179...
#> $ exposure     <fct> 1, 1, 1, 1, 1, 0, 1, 1,...
#> $ y            <dbl> -120.253, 0.942, -51.95...
#> $ exposure_prob <dbl> 0.5000, 0.9072, 0.3465,...
```

# Monte Carlo simulation

- For simulation, 1000 replicates
- Sample size: 1000

```r
doMC::registerDoMC(cores = 4)
mc_list <- mc_setoguchi(
  N = 1000, n_dat = 1000,
  scenario = scen,
  parallel = TRUE
)
```

# Evaluation

# Average standardized absolute mean distance (ASAM)

- Covariate balancing: standardized mean differece, which is standardized by pooled sd
- Average the abs(covariate balancing) across all the covariates
- Lower: treatment and control groups are more similar w.r.t. the given covariates.

```
doMC::registerDoMC(cores = 8)
logit_asam <-
  mc_list %>%
  compute_asam(
    treatment = "exposure", outcome = "y", exclude = "exposure_prob",
    formula = exposure ~ . - y - exposure_prob, method = "logit",
    mc_col = "mcname", sc_col = "scenario", parallel = TRUE
  )
```

# ASAM for each model

**Table 1:** ASAM performance

| Scenarios | Logistic regression | Random forests | SVM |
|---|---|---|---|
| A | 0.012 | 0.012 | 0.010 |
| B | 0.031 | 0.028 | 0.041 |
| F | 0.036 | 0.034 | 0.043 |
| G | 0.077 | 0.074 | 0.081 |

- Under 0.2 is acceptable (Lee et al., 2010)
- All are OK.

# Effect estimator

## Estimation of ATE

- Inverse probability of treatment weighing (IPTW):

$$IPTW_i = \frac{Z_i}{\hat{e}_i} + \frac{1 - Z_i}{1 - \hat{e}_i}$$

- Inverse probability weighting (IPW): weighted regression of outcome on treatment $\hat{\Delta}_{IPW}$
- Stabilized inverse probability weighting (SIPW): $\hat{\Delta}_{SIPW}$

## Evaluation

- Empirical distribution
    - Histogram
    - Bias: difference between true effect ($\gamma = -0.4$)
    - Standard deviation
    - Confidence interval

# Inverse Probability of Treatment Weighing

```
doMC::registerDoMC(cores = 8)
iptw_logit <-
  mc_list %>%
  add_iptw(
    treatment = "exposure",
    formula = exposure ~ . - y - exposure_prob, method = "
    mc_col = "mcname", sc_col = "scenario", parallel = TRU
  )
```

# Empirical Distribution of IPTW

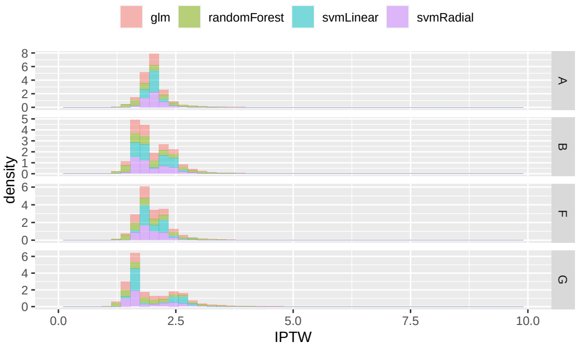


**Figure 2:** Empirical Distribution of IPTW

# IPW and SIPW

```
doMC::registerDoMC(cores = 8)
ipw_logit <-
  mc_list %>%
  compute_ipw(
    treatment = "exposure", outcome = "y",
    formula = exposure ~ . - y - exposure_prob,
    method = "logit",
    mc_col = "mcname", sc_col = "scenario",
    parallel = TRUE
  )
```

- weight of treatment: 1
- weight of control: $\frac{p_i}{1-p_i}$
- If $\hat{e}$ is proper
    - then two weights are similar
    - ATE estimate: difference of weighted means

# Empirical Distribution of IPW

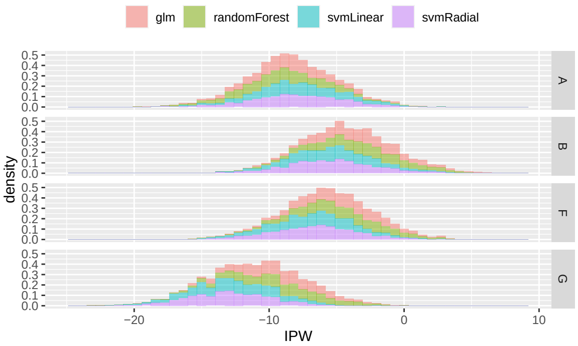


**Figure 3:** Empirical Distribution of IPW

# Empirical Distribution of SIPW

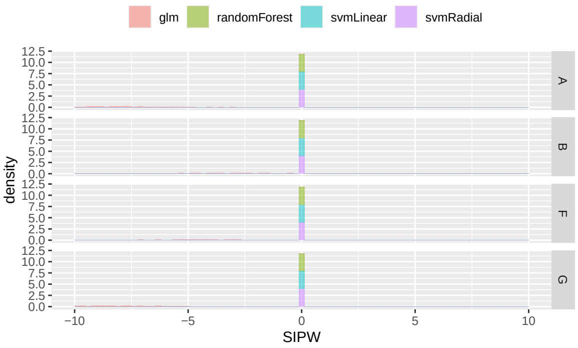


**Figure 4:** Empirical Distribution of SIPW

# Performance Metric of IPW

| Metric | Scenarios | Model | | | |
|---|---|---|---|---|---|
| | | Logistic regression | Random forests | SVM (Linear) | SVM (Radial) |
| bias | A | 8.88 | 9.57 | 8.24 | 8.42 |
| | B | 4.21 | 4.27 | 5.95 | 5.99 |
| | F | 5.25 | 5.37 | 7.31 | 7.16 |
| | G | 9.78 | 9.62 | 13.11 | 13.34 |
| estimate | A | -8.47 | -9.16 | -7.83 | -8.00 |
| | B | -3.33 | -3.15 | -5.37 | -5.38 |
| | F | -4.69 | -4.73 | -6.88 | -6.71 |
| | G | -9.38 | -9.21 | -12.71 | -12.94 |
| mse | A | 74.18 | 88.14 | 66.38 | 69.79 |
| | B | 18.29 | 19.72 | 35.49 | 35.89 |
| | F | 27.58 | 30.11 | 52.09 | 50.13 |
| | G | 89.72 | 89.86 | 161.94 | 166.77 |
| sd | A | 3.00 | 3.38 | 3.35 | 3.47 |
| | B | 3.11 | 3.48 | 3.28 | 3.33 |
| | F | 3.02 | 3.38 | 3.18 | 3.22 |
| | G | 3.02 | 3.50 | 3.24 | 3.08 |

# Performance Metric of SIPW

| Metric | Scenarios | Model | | | |
|---|---|---|---|---|---|
| | | Logistic regression | Random forests | SVM (Linear) | SVM (Radial) |
| bias | A | 0.402 | 0.402 | 0.402 | 0.402 |
| | B | 0.401 | 0.401 | 0.401 | 0.401 |
| | F | 0.401 | 0.401 | 0.402 | 0.402 |
| | G | 0.402 | 0.402 | 0.403 | 0.403 |
| estimate | A | -0.002 | -0.002 | -0.002 | -0.002 |
| | B | -0.001 | -0.001 | -0.001 | -0.001 |
| | F | -0.001 | -0.001 | -0.002 | -0.002 |
| | G | -0.002 | -0.002 | -0.003 | -0.003 |
| mse | A | 0.158 | 0.158 | 0.158 | 0.158 |
| | B | 0.159 | 0.159 | 0.159 | 0.159 |
| | F | 0.159 | 0.159 | 0.159 | 0.159 |
| | G | 0.158 | 0.158 | 0.157 | 0.157 |
| sd | A | 0.001 | 0.001 | 0.001 | 0.001 |
| | B | 0.001 | 0.001 | 0.001 | 0.001 |
| | F | 0.001 | 0.001 | 0.001 | 0.001 |
| | G | 0.001 | 0.001 | 0.001 | 0.001 |

# Related Contents

# About this project

### Project repository

https://github.com/ygeunkim/psweighting-ml

### Project package

https://github.com/ygeunkim/propensityml

# References I

Lee, B. K., Lessler, J., and Stuart, E. A. (2010). Improving propensity score weighting using machine learning. *Statistics in Medicine*, 29(3):337–346.

Pirracchio, R., Petersen, M. L., and van der Laan, M. (2014). Improving propensity score estimators' robustness to model misspecification using super learner. *American Journal of Epidemiology*, 181(2):108–119.

Setoguchi, S., Schneeweiss, S., Brookhart, M. A., Glynn, R. J., and Cook, E. F. (2008). Evaluating uses of data mining techniques in propensity score estimation: a simulation study. *Pharmacoepidemiology and Drug Safety*, 17(6):546–555.