Propensity Score Weighting for Causal Subgroup Analysis in R: A Vignette

Siyun Yang Duke University
Tianhui Zhou Duke University
Fan Li Duke University
Laine E. Thomas Duke University

Abstract

The **PSweight.sga** package provides a pipeline for causal subgroup analysis with propensity score weighting for binary treatment comparison. It offers visualization tool for subgroup balance check though the Connect-S plot, and also offers point and variance estimation with a variety of weighting schemes for the (weighted) subgroup average treatment effects on both the additive and ratio scales. We provide an illustrative example to demonstrate the functionality of the **PSweight.sga** package. We show after implementing overlap weights paired with the post-LASSO algorithm from the **PSweight.sga** package, the subgroups achieve perfect covariate balance and the estimated subgroup average causal effects have smaller variance compared to the inverse probability weighting paired with a main effect logistic propensity score model.

Keywords: Causal inference, Propensity score, Weighting, Subgroup analysis.

1. Overview of Package

Propensity score is one of the most widely used causal inference methods for observational studies (Rosenbaum and Rubin 1983). The **PSweight.sga** package provides a pipeline for causal subgroup analysis with propensity score weighting (Yang, Lorenzi, Papadogeorgou, Wojdyla, Li, and Thomas 2020). For causal inference focusing on the average causal effect in the overall sample, please refer to R package **PSweight**(Zhou, Tong, Li, and Thomas 2020).

The **PSweight.sga** package includes two modules tailored for design and analysis of observational studies. The design module provides diagnostics to assess the adequacy of the propensity score model and the weighted target population, prior to the use of outcome data. The analysis module provides functions to estimate the causal estimands. Currently, only binary treatment indicator is supported. Table 1 summarizes the functions in the **PSweight.sga** package, and we briefly describe the two modules below.

1.1. Design Module

PSweight.sga offers the `SumStat_sga()` function to generate the estimated propensity scores and balance diagnostics within subgroups after propensity score weighting. It uses the following code snippet:

```
SumStat_sga(subgroup, xname, ps.formula, ps.estimate,
+     trtgrp, zname, yname, data, weight= "overlap", method="LASSO")
```

When subgrouping variables and covariates are provided through the argument `subgroup` and `xname`, by default, the propensity scores are estimated by the Post-LASSO. The treatment indicator must be provided through the argument `zname` with this option. Alternatively, the propensity scores are estimated by the logistic regression, through the argument `ps.formula`. `SumStat_sga()` produces a `SumStat_sga` object, with estimated propensity scores, subgroup level unweighted and weighted ASMD, variance inflation and effective sample sizes (defined in Yang *et al.* (2020)).

Table 1: Functions in the **PSweight.sga**

Function	Description
`SumStat_sga()`	Generate a `SumStat_sga` object with information of propensity scores, subgroup level unweighted and weighted ASMD, variance inflation and effective sample sizes
`plot.SumStat()`	Generate a Connect-S plot from the `SumStat_sga` object
`PSweight_sga()`	Generate a `PSweight_sga` object with information of propensity scores, potential average subgroup causal effects, bootstrap estimates, and propensity score weighting method
`summary.PSweight_sga()`	Generate a `summary.PSweight_sga` object with information of subgroup causal effects, variance and p values
`ForestPlot()`	Generate a forestplot plot from the `summary.PSweight_sga` object

Diagnostics of propensity score models can be visualized with the `plot.SumStat()` function. It takes the `SumStat_sga` object and produces a Connect-S plot based on the ASMD. Each row represents a subgroup variable, with subgroup sample size displaying at the end ; each column represents a confounder (or covariate) that we want to balance. The color of the dot is coded based on the ASMD of confounder X in subgroup S, with darker color meaning more severe imbalance. The end of each row also presents subgroup-specific approximate variance inflation. The plot function is implemented as follows:

```
plot(x, varlist=NULL, base=FALSE, plotsub=FALSE)
```

, where the ASMD before weighting can be presented through the argument `base=TRUE`, and the balance check for subgrouping variables can be displayed through the argument `plotsub=TRUE`.

1.2. Analysis Module

The analysis module of **PSweight.sga** includes two functions: `PSweight_sga()` and `summary.PSweight_sga()`

The `PSweight_sga()` function estimates the average potential outcomes in the target subpopulation. By default, the bootstrap sample variance is implemented. The `weight` argument can take `"IPW"`, `"treated"`, and `"overlap"`. More detailed descriptions of each input argument in the `PSweight_sga()` function can be found in Table 2. A typical `PSweight_sga()` code snippet looks like

```
PSweight_sga (ps.formula, ps.estimate, subgroup, xname,
+     trtgrp, zname, yname, data, R=50, weight="overlap", method="LASSO")
```

Table 2: Arguments for function `PSweight_sga()` in the analysis module of **PSweight.sga**.

Argument	Description	Default
`ps.formula`	An optional symbolic description of the propensity score model when `xname` is not provided.	`NULL`
`ps.estimate`	An optional matrix or a vector with externally estimated propensity scores for each observation.	`NULL`
`subgroup`	A character vector specifying name of the subgrouping variables.	`NULL`
`xname`	A character vector specifying name of the confounders (or covariates).	`NULL`
`trtgrp`	An optional character defining the *treated* population for estimating (pairwise) ATT. It can also be used to specify the treatment level when only a vector of values are supplied for `ps.estimate` in the binary treatment setting.	Last value in alphabetic order
`zname`	An optional character specifying the name of the treatment variable when `ps.formula` is not provided.	`NULL`
`yname`	A character specifying name of the outcome variable in `data`.	
`weight`	A character specifying the type of weights to be used.	`"overlap"`
`R`	Number of bootstrap replicates	50
`method`	a character to specify the method for propensity model.	`"LASSO"` if `"xname"` is provided, otherwise `"glm"` if `ps.formula` is provided

The `summary.PSweight_sga()` function synthesizes information from the `PSweigh_sga` object for statistical inference, including test of subgroup average treatment effect and test of heterogenerous treatment effect across subgroup levels. A typical code snippet looks like

```
summary(object, type = "DIF", het = FALSE)
```

In the `summary.PSweight_sga()` function, the argument `type` corresponds to the three types estimands: `type = "DIF"` is the default argument that specifies the additive causal contrasts;

`type = "RR"` specifies the contrast on the log scale; `type = "OR"` specifies the contrast on the log odds scale (Zhou *et al.* 2020). The `summary.PSweight_sga()` provides a comparison of the average subgroup potential outcomes. Confidence intervals (CIs) and p-values are obtained using bootstrap.

2. Illustration with a simulated data

We demonstrate **PSweight.sga** in a simulated data that estimates the subgroup causal effect defined by pre-specified binary variables. The simulated dataset includes 3000 rows, with each row represents information recorded from each individual. There are 22 variables (columns), including 20 covariates, 1 binary treatment indicator and 1 outcome variable. The binary treatment is the variable "Treatment", and the outcome of interest is variable "Y". X1- X18 are confounders among which X1- X8 are binary, and X9- X18 are continuous. The pre-specified subgroups are defined by binary variables X1- X3, and X19- X20. Information on the study variables can be summarized using the `str()` function as below:

```
R> str(psdata_sga)
```

```
'data.frame':        3000 obs. of  22 variables:
 $ Treatment: int  1 0 0 1 1 0 0 1 1 0 ...
 $ X1       : int  0 0 0 0 0 1 0 1 1 0 ...
 $ X2       : int  1 1 0 0 1 0 0 0 1 0 ...
 $ X3       : int  0 0 1 1 0 0 0 1 1 0 ...
 $ X4       : int  0 0 1 1 0 0 0 0 0 1 ...
 $ X5       : int  1 0 0 0 0 0 0 1 1 0 ...
 $ X6       : int  1 0 0 0 1 1 0 1 0 1 ...
 $ X7       : int  0 0 1 0 0 0 0 0 0 1 ...
 $ X8       : int  0 0 0 0 0 1 1 0 0 0 ...
 $ X9       : num  -1.354 0.632 1.639 -2.319 0.851 ...
 $ X10      : num  -0.579 -1.511 -2.124 -0.925 -0.559 ...
 $ X11      : num  -0.861 0.5734 0.0878 0.3704 -1.1034 ...
 $ X12      : num  0.973 -0.187 1.771 0.743 0.166 ...
 $ X13      : num  0.61915 0.16033 -0.39919 -0.00301 0.447 ...
 $ X14      : num  1.3854 -0.3761 -0.8161 -0.0866 0.2918 ...
 $ X15      : num  -1.487 0.723 0.208 -0.091 0.623 ...
 $ X16      : num  0.639 0.96 -0.844 0.534 -1.035 ...
 $ X17      : num  0.375 -0.206 -1.645 -1.535 0.871 ...
 $ X18      : num  0.37 0.256 -0.501 -0.331 0.689 ...
 $ X19      : int  0 1 0 1 1 0 0 1 0 0 ...
 $ X20      : int  0 1 0 0 0 0 0 0 1 1 ...
 $ Y        : num  -0.696 2.473 -0.493 -0.541 1.516 ...
```

We first estimate propensity scores via the traditional main effect logistic regression, and pair it with IPW. The output summarizes the treatment group level (for defining ATT) (`"trtgrp"`), estimated propensity scores (`"propensity"`), weighting scheme (`"ps.weights"`), the absolute standardized mean difference before (`"ASD_bs"`) and after weighting (`"ASD"`), variance inflation factor (`"vif"`), effective sample size (`"ess"`), subgroup sample sizes (`"nsubg"`), subgrouping variables (`"subgroup"`) and propensity score method (`"method"`).

```
R> # pre-specify the confounder names:  X1-X20
R> xname <- paste0('X', 1:20)
# pre-specify subgroups of interest by column names
R> subgroup <- paste0('X',c(1,2,19,20))
R> ps.form_m <- paste("Treatment~",paste0("X",1:20,collapse = "+"))
R> p <- SumStat_sga(ps.formula = ps.form_m, subgroup = subgroup,
+     data=psdata_sga, method="glm", weight="IPW")
R> summary(p)

          Length Class  Mode
trtgrp        0   -none- NULL
propensity 6000   -none- numeric
ps.weights    1   -none- character
ASD         220   -none- numeric
ASD_bs      220   -none- numeric
vif          11   -none- numeric
vif_bs       11   -none- numeric
nsubg        11   -none- numeric
ess          22   -none- numeric
ess_bs       22   -none- numeric
subgroup      5   -none- character
method        1   -none- character

R> plot(p, base = T)
R> plot(p)
```

The "base = T" option in plot.SumStat_sga() function plots the Connect-S plot of the unweighted subgroup ASMD, displayed in Figure 1. Subgroups are displayed in rows and all confounders are displayed in columns. A common threshold for balance is that the ASMD should be below 0.1 or 0.2. Severe imbalance is prevalent in all subgroups and the overall sample. Figure 2 displays the Connect-S plot of the subgroup ASMD and approximate variance inflation after applying IPW-Main. Although IPW-Main balances covariates in the overall sample and greatly improves subgroup imbalance, substantial imbalance still exists in subgroups such as in subgroup X19=1 and X20=1.

Next, we fit a Post-LASSO propensity score model, paired with OW, and conduct balance check again. The ("nonzero_coef") option returns the main effects and subgroup-confounder interactions selected by the LASSO algorithm. Figure 3 shows that OW-pLASSO successfully balances all covariates in the subgroups and overall sample.

```
R> set.seed(123)
R> plasso <- SumStat_sga( subgroup=subgroup, xname=xname, zname="Treatment",
data=psdata_sga, method='LASSO', weight="overlap")
R> plasso$nonzero_coef

[1] "(Intercept)" "X3"          "X4"          "X5"          "X6"          "X7"
[7] "X8"          "X9"          "X10"         "X11"         "X12"         "X13"
[13] "X14"         "X15"         "X16"         "X17"         "X18"         "X1"
[19] "X2"          "X19"         "X20"         "X4:X1"       "X9:X1"       "X13:X1"
```

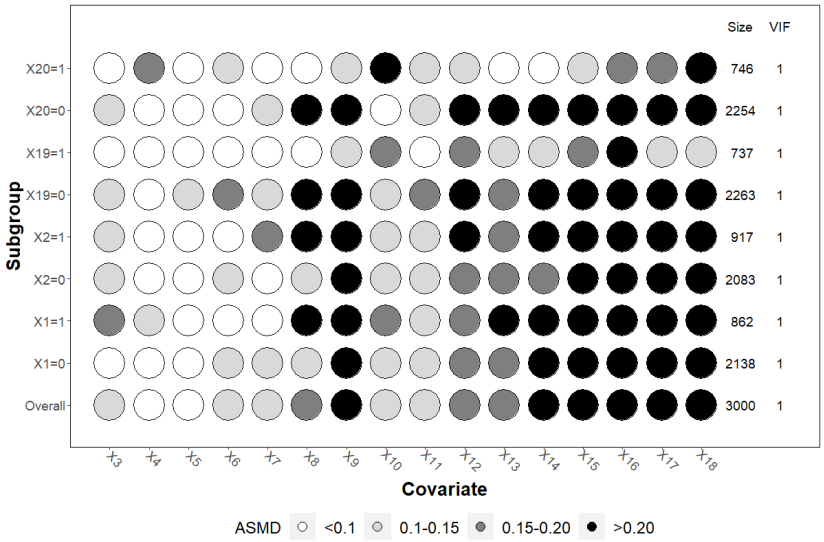

Figure 1: Connect-S plot of the unweighted subgroup ASMD, generated by `plot.SumStat_sga()` function. Subgroups are displayed in rows and all confounders are displayed in columns.

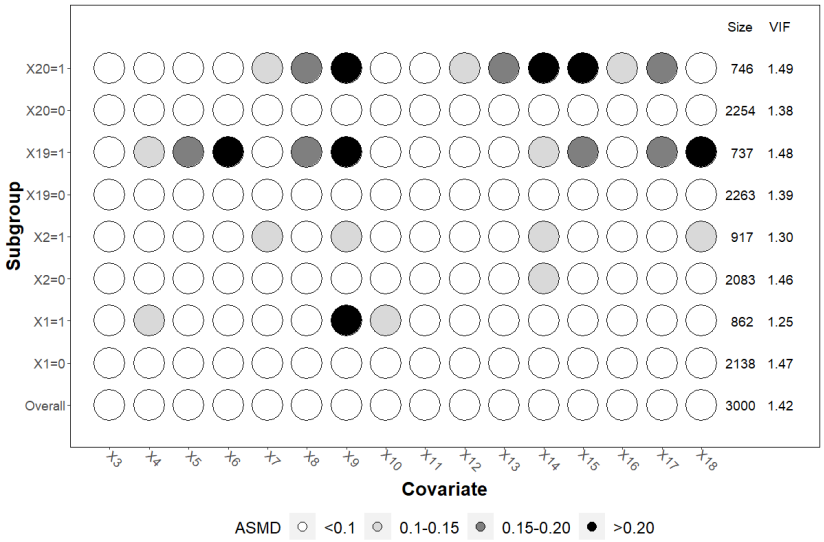

Figure 2: Connect-S plot of the subgroup ASMD and approximate variance inflation after applying IPw-Main, generated by `plot.SumStat_sga()` function. Subgroups are displayed in rows and all confounders are displayed in columns.

```
[25] "X14:X1"      "X7:X2"       "X8:X2"       "X9:X2"       "X14:X2"      "X5:X19"
[31] "X6:X19"      "X8:X19"      "X9:X19"      "X11:X19"     "X14:X19"     "X15:X19"
[37] "X17:X19"     "X18:X19"     "X8:X20"      "X9:X20"      "X10:X20"     "X12:X20"
[43] "X13:X20"     "X14:X20"     "X15:X20"     "X16:X20"     "X17:X20"
```

```
R> plot(plasso)
```

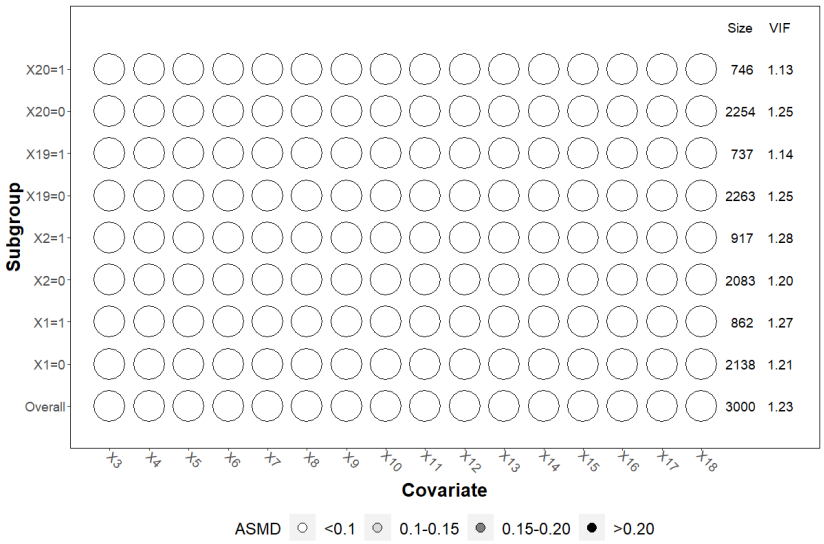

Figure 3: Connect-S plot of the subgroup ASMD and approximate variance inflation after applying OW-pLASSO, generated by `plot.SumStat_sga()` function. Subgroups are displayed in rows and all confounders are displayed in columns.

We estimate the subgroup weighted average causal effects using the outcome variable "Y" in the data. The following code estimate the effects using IPW-Main and OW-pLASSO, separately. The returns of the `summary.PSweight_sga()` function include the subgroup level causal effect estimate, the bootstrap SE, the associated confidence interval and p value. The results suggest that there are statistically significant causal effects in the overall sample, and subgroups.

```
R> p1 <- PSweight_sga(ps.formula=ps.form_m,subgroup=subgroup,yname="Y",data=psdata_sga,
+    R=50,weight="IPW")
R> s1 <- summary(p1)
R> p2 <- PSweight_sga(xname=xname, subgroup=subgroup, yname="Y", zname="Treatment",
+    data=psdata_sga, R=50, method='LASSO', weight="overlap")
R> s2 <- summary(p2)
R> s2$estimates
```

```
          Estimate  Std.Error  Lower.CL   Upper.CL  p.value
Overall -0.9953276 0.06679451 -1.132407 -0.8851090  0
X1=0    -1.0131678 0.08111569 -1.185255 -0.8887685  0
X1=1    -0.9511043 0.12942170 -1.132639 -0.6779124  0
X2=0    -1.0316631 0.07750539 -1.173646 -0.8989386  0
X2=1    -0.9130546 0.12063904 -1.158183 -0.7320105  0
```

```
X19=0   -1.0056671 0.08174619 -1.163358 -0.8855424  0
X19=1   -0.9705369 0.13814453 -1.262539 -0.7603661  0
X20=0   -0.8936755 0.07628685 -1.033460 -0.7442411  0
X20=1   -1.2319122 0.12020582 -1.485269 -1.0959429  0
```

Further, test of heterogenerous causal effect is conducted for the pairwise average causal effects between subgroups levels when there are more than three levels through the `"het=T"` option. The results suggest that there is statistically significant HTE in the subgroups defined by variable X20.

```
R> summary(p2, het=T)$het_eval
```

```
                     Estimate Std.Error   Lower.CL  Upper.CL      p.value
X1=1 - X1=0 :    -0.06206344 0.1593898 -0.4339522   0.2213125    0.72
X2=1 - X2=0 :    -0.11860847 0.1378485 -0.3075499  0.1413782     0.48
X19=1 - X19=0 :  -0.03513017 0.1685837 -0.3089327  0.2217957     0.84
X20=1 - X20=0 :   0.33823670 0.1432764  0.1040208  0.5998915     0.00
```

Last, we compare the estimated subgroup causal effects and the corresponding 95% CIs from IPW-Main and OW-pLASSO in a forest plot. Firgure 4 shows in the overall sample, the CI of IPW-Main is narrower than the OW-pLASSO. However, in almost all subgroups, OW-pLASSO is more efficient. In the subgroups defined by X19=1 and X20=1, the point estimates from IPW-Main and OW-pLASSO are dramatically different as the IPW-Main fails to correct the severe imbalance present in these subgroups, resulting in biased estimation of the subgroup causal effects.

```
R> ForestPlot(list(s1, s2), legend_label = c("IPW-Main", "OW-pLASSO"),
+     xlab="Causal Effects in Y")
```

3. Summary

Propensity score weighting is an important tool for causal inference and comparative effectiveness research. This vignettee introduces the **PSweight.sga** package and demonstrates its functionality with a simulated data example. The **PSweight.sga** provides visualization tool for subgroup balance check though the Connect-S plot. It also offers point and variance estimation with a variety of weighting schemes for the (weighted) subgroup average treatment effects on both the additive and ratio scales. The simple example demonstrates the ability of OW-pLASSO algorithm to achieve perfect subgroup covariate balance and small variance in causal subgroup analysis.

References

Rosenbaum PR, Rubin DB (1983). "The Central Role of the Propensity Score in Observational Studies for Causal Effects." *Biometrika*, **70**(1), 41–55. doi:10.1093/biomet/70.1.41.

Yang S, Lorenzi E, Papadogeorgou G, Wojdyla DM, Li F, Thomas LE (2020). "Propensity Score Weighting for Causal Subgroup Analysis." *arXiv preprint arXiv:2010.02121*.

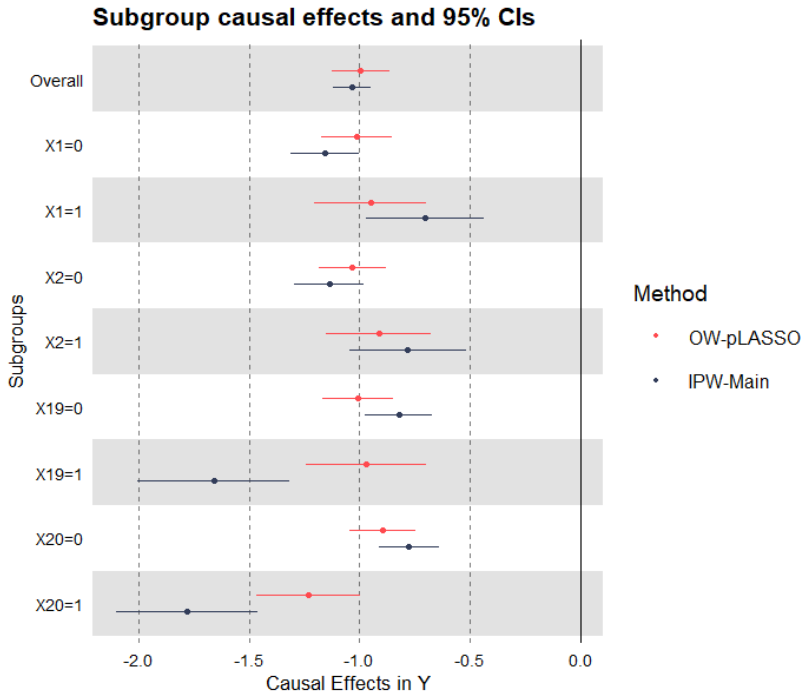

Figure 4: Point estimates and 95% confidence intervals of the treatment effects.

Zhou T, Tong G, Li F, Thomas LE (2020). "Psweight: An r package for propensity score weighting analysis." *arXiv preprint arXiv:2010.08893*.