PSweight: An R Package for Propensity Score Weighting Analysis

by Tianhui Zhou, Guangyu Tong, Fan Li, Laine E. Thomas and Fan Li

Abstract Propensity score weighting is an important tool for comparative effectiveness research. Besides the inverse probability of treatment weights (IPW), recent development has introduced a general class of balancing weights, corresponding to alternative target populations and estimands. In particular, the overlap weights (OW) lead to optimal covariate balance and estimation efficiency, and a target population of scientific and policy interest. We develop the R package **PSweight** to provide a comprehensive design and analysis platform for causal inference based on propensity score weighting. **PSweight** supports (i) a variety of balancing weights, (ii) binary and multiple treatments, (iii) simple and augmented weighting estimators, (iv) nuisance-adjusted sandwich variances, and (v) ratio estimands. **PSweight** also provides diagnostic tables and graphs for covariate balance assessment. We demonstrate the functionality of the package using a data example from the National Child Development Survey (NCDS), where we evaluate the causal effect of educational attainment on income.

1 Introduction

Propensity score is one of the most widely used causal inference methods for observational studies (Rosenbaum and Rubin, 1983). Propensity score methods include weighting, matching, stratification, regression, and mixed methods such as the augmented weighting estimators. The **PSweight** package provides a design and analysis pipeline for causal inference with propensity score weighting (Robins et al., 1994; Hirano et al., 2003; Lunceford and Davidian, 2004; Li et al., 2018). There are a number of existing R packages on propensity score weighting (see Table 1). Comparing to those, **PSweight** offers three major advantages: it incorporates (i) visualization and diagnostic tools of checking covariate overlap and balance, (ii) a general class of balancing weights, including overlap weights and inverse probability of treatment weights, and (iii) multiple treatments. More importantly, **PSweight** comprises a wide range of functionalities, whereas each of the competing packages only supports a subset of these functionalities. As such, **PSweight** is currently the most comprehensive platform for causal inference with propensity score weighting, offering analysts a one-stop shop for the design and analysis. Table 1 summarizes the key functionalities of **PSweight** in comparison to related existing R packages. We elaborate the main features of **PSweight** below.

PSweight facilitates better practices in the design stage of observational studies, an aspect that has not been sufficiently emphasized in related packages. Specifically, we provide a design module that facilitates visualizing overlap (also known as the positivity assumption) and evaluating covariate balance without access to the final outcome (Austin and Stuart, 2015). When there is limited overlap, **PSweight** allows for symmetric propensity score trimming (Crump et al., 2009; Yoshida et al., 2018) and optimal trimming (Crump et al., 2009; Yang et al., 2016) to improve the internal validity. We extend the class of balance metrics suggested in Austin and Stuart (2015) and Li et al. (2019) for binary treatments, and those in McCaffrey et al. (2013) and Li and Li (2019) for multiple treatments. In addition, the design module helps describe the weighted target population by providing the information required in the standard "Table 1" of a clinical article.

In addition to the standard inverse probability of treatment weights (IPW), **PSweight** implements the average treatment effect among the treated (Treated) weights, overlap weights (OW), matching weights (MW) and entropy weights (EW) for both binary (Li and Greene, 2013; Mao et al., 2018; Li et al., 2018; Zhou et al., 2020) and multiple treatments (Yoshida et al., 2017; Li and Li, 2019). All weights are members of the family of balancing weights (Li et al., 2018); the last three types of weights target at the subpopulation with improved overlap in the covariates between (or across) treatment groups, similar to the target population in randomized controlled trials (Thomas et al., 2020a,b). Among them, OW achieves optimal balance and estimation efficiency (Li et al., 2018, 2019). We also implement the augmented weighting estimators corresponding to each of the above weighting schemes (Mao et al., 2018). By default, **PSweight** employs parametric regression models to estimate propensity scores and potential outcomes. Nonetheless, it also allows for propensity scores to be estimated by external machine learning methods including generalized boosted regression models (McCaffrey et al., 2013) and super learner (Van der Laan et al., 2007), or importing any other propensity or outcome model estimates of interest.

To our knowledge, **PSweight** is the first R package to accommodate a variety of balancing weighting schemes with multiple treatments. Existing R packages such as **twang** (Ridgeway et al., 2020),

Table 1: Comparisons of existing R packages that implement propensity score weighting with discrete treatments. Binary treatments and additive estimands are implemented in all packages, and therefore those two columns are omitted.

	Multiple treatments	Balance diagnostics	IPW/ATT weights	OW/other weights	Ratio estimands	Augmented weighting	Nuisance-adj variance	Optimal trimming
PSweight	✓	✓	✓	✓	✓	✓	✓	✓
twang	✓	✓	✓	×	×	×	×	×
CBPS	✓	✓	✓	×	×	✓	✓	×
PSW	×	✓	✓	✓	✓	✓	✓	×
optweight	✓	×	✓	×	×	×	×	×
ATE	✓	✓	✓	×	×	×	✓	×
WeightIt	✓	×	✓	✓	×	×	×	×
causalweight	✓	×	✓	×	×	✓	×	×
sbw	×	✓	✓	×	×	×	×	×

CBPS (Fong et al., 2019), **optweight** (Greifer, 2019) have also implemented weighting-based estimation with multiple treatments, but focus on IPW. The **PSW** R package (Mao and Li, 2018) implements both OW and MW and allows for nuisance-adjusted variance estimation, but it is only for binary treatments. To better assist applied researchers to perform propensity score weighting analysis, this article provides a full introduction of the **PSweight** package. In what follows, we explain the methodological foundation of **PSweight** and outline the main functions and their arguments. We further illustrate the use of these functions with a data example that studies the causal effect of educational attainment on income, and finally conclude with a short discussion.

2   Overview of propensity score weighting

Before diving into the implementation details of **PSweight**, we briefly introduce the basics of the propensity score weighting framework.

Binary treatments

Assume we have an observational study with N units. Each unit i ($i = 1, 2, \ldots, N$) has a binary treatment indicator Z_i ($Z_i = 0$ for control and $Z_i = 1$ for treated), a vector of p covariates $X_i = (X_{1i}, \cdots, X_{pi})$. For each unit i, we assume a pair of potential outcomes $\{Y_i(1), Y_i(0)\}$ mapped to the treatment and control status, of which only the one corresponding to the observed treatment is observed, denoted by $Y_i = Z_i Y_i(1) + (1 - Z_i) Y_i(0)$; the other potential outcome is counterfactual.

Causal effects are contrasts of the potential outcomes of the same units in a *target population*, which usually is the population of a scientific interest (Thomas et al., 2020b). **PSweight** incorporates a general class of weighted average treatment effect (WATE) estimands. Specifically, assume the observed sample is drawn from a probability density $f(x)$, and let $g(x)$ denote the covariate density of the target population. The ratio $h(x) \propto g(x)/f(x)$ is called the *tilting function*, which adjusts the distribution of the observed sample to represent the target population. Denote the conditional expectation of the potential outcome by $m_z(x) = \mathbb{E}[Y(z)|X = x]$ for $z = 0, 1$. Then, we can represent the average treatment effect over the target population by a WATE estimand:

$$\tau^h = \mathbb{E}_g[Y(1) - Y(0)] = \frac{\mathbb{E}\{h(x)(m_1(x) - m_0(x))\}}{\mathbb{E}\{h(x)\}}. \tag{2.2.1}$$

To estimate (2.2.1), **PSweight** maintains two standard assumptions: (1) *unconfoundedness*: $\{Y(1), Y(0)\} \perp Z \mid X$; (2) *overlap*: $0 < P(Z = 1|X) < 1$. The propensity score is the probability of a unit being assigned to the treatment group given the covariates (Rosenbaum and Rubin, 1983): $e(x) = P(Z = 1|X = x)$. While assumption (1) is generally untestable and critically depends on substantive knowledge, assumption (2) can be checked visually from data by comparing the distribution of propensity scores between treatment and control groups.

For a given tilting function $h(x)$ (and correspondingly a WATE estimand τ^h), we can define the *balancing weights* (w_1, w_0) for the treated and control units: $w_1(x) \propto h(x)/e(x)$ and $w_0(x) \propto h(x)/\{1 - e(x)\}$. These weights balance the covariate distributions between the treated and control groups towards the target population (Li et al., 2018). **PSweight** implements the following Hájek

Table 2: Target populations, tilting functions, estimands and the corresponding balancing weights for binary treatments in **PSweight**.

Target population	Tilting function $h(x)$	Estimand	Balancing weights (w_1, w_0)
Combined	1	ATE	$\left(\frac{1}{e(x)}, \frac{1}{1-e(x)}\right)$
Treated	$e(x)$	ATT	$\left(1, \frac{e(x)}{1-e(x)}\right)$
Overlap	$e(x)(1-e(x))$	ATO	$(1-e(x), e(x))$
Matching	$\xi_1(x)$	ATM	$\left(\frac{\xi_1(x)}{e(x)}, \frac{\xi_1(x)}{1-e(x)}\right)$
Entropy	$\xi_2(x)$	ATEN	$\left(\frac{\xi_2(x)}{e(x)}, \frac{\xi_2(x)}{1-e(x)}\right)$

Notes: $\xi_1(x) = \min\{e(x), 1-e(x)\}$ and $\xi_2(x) = -\{e(x)\log(e(x)) + (1-e(x))\log(1-e(x))\}$.

estimator for WATE:

$$\hat{\tau}^h = \hat{\mu}_1^h - \hat{\mu}_0^h = \frac{\sum_{i=1}^N w_1(x_i)Z_iY_i}{\sum_{i=1}^N w_1(x_i)Z_i} - \frac{\sum_{i=1}^N w_0(x_i)(1-Z_i)Y_i}{\sum_{i=1}^N w_0(x_i)(1-Z_i)}, \tag{2.2.2}$$

where the weights are calculated based on the propensity scores estimated from the data. Clearly, specification of $h(x)$ defines the target population and estimands. **PSweight** primarily implements the following three types of balancing weights (see Table 2 for a summary):

- *Inverse probability of treatment weights* (IPW), whose target population is the combined treatment and control group represented by the observed sample, and the target estimand is the average treatment effect among the combined population (ATE).

- *Treated weights*, whose target population is the treated group, and target estimand is the average treatment effect for the treated population (ATT). Treated weights can be viewed as a special case of IPW because it inversely weights the control group.

- *Overlap weights* (OW) (Li et al., 2018; Li and Li, 2019), whose target population is the subpopulation with the most overlap in the observed covariates between treatment and control groups . In medicine this is known as the population in clinical equipoise and is the population eligible to be enrolled in randomized clinical trials. The target estimand of OW is the average treatment effect for the overlap population (ATO).

IPW has been the dominant weighting method in the literature, but has a well-known shortcoming of being sensitive to extreme propensity scores, which induces bias and large variance in estimating treatment effects. OW addresses the conceptual and operational problems of IPW. Among all balancing weights, OW leads to the smallest asymptotic (and often finite-sample) variance of the weighting estimator (2.2.2). (Li et al., 2018, 2019). Recent simulations also show that OW provides more stable causal estimates under limited overlap (Li et al., 2019; Mao et al., 2018; Yoshida et al., 2017, 2018), and is more robust to misspecification of the propensity score model (Zhou et al., 2020).

PSweight implements two additional types of balancing weights: matching weights (MW) (Li and Greene, 2013), and entropy weights (EW) (Zhou et al., 2020). Similar to OW, MW and EW focus on target populations with substantial overlap between treatment groups. Though having similar operating characteristics, MW and EW do not possess the same theoretical optimality as OW, and are less used in practice. Therefore, we will not separately describe MW and EW hereafter.

In observational studies, propensity scores are generally unknown and need to be estimated. Therefore, propensity score analysis usually involves two steps: (1) estimating the propensity scores, and (2) estimating the causal effects based on the estimated propensity scores. In **PSweight**, the default model for estimating propensity scores with binary treatments is a logistic regression model. Spline or polynomial models can be easily incorporated by adding `bs()`, `ns()` or `poly()` terms into the model formula. **PSweight** also allows for importing propensity scores estimated from external routines, such as boosted models or super learner.

Goodness-of-fit of the propensity score model is usually assessed based on the resulting covariate balance. In the context of propensity score weighting, this is measured based on either the absolute standardized difference (ASD):

$$\text{ASD} = \left| \frac{\sum_{i=1}^N w_1(x_i)Z_iX_{pi}}{\sum_{i=1}^N w_1(x_i)Z_i} - \frac{\sum_{i=1}^N w_0(x_i)(1-Z_i)X_{pi}}{\sum_{i=1}^N w_0(x_i)(1-Z_i)} \right| \Bigg/ \sqrt{\frac{s_1^2 + s_0^2}{2}}, \tag{2.2.3}$$

or the target population standardized difference (PSD), $\max\{\text{PSD}_0, \text{PSD}_1\}$, where

$$\text{PSD}_z = \left| \frac{\sum_{i=1}^N w_z(\boldsymbol{x}_i)\mathbb{1}\{Z_i = z\}X_{pi}}{\sum_{i=1}^N w_z(\boldsymbol{x}_i)\mathbb{1}\{Z_i = z\}} - \frac{\sum_{i=1}^N h(\boldsymbol{x}_i)X_{pi}}{\sum_{i=1}^N h(\boldsymbol{x}_i)} \right| \Bigg/ \sqrt{\frac{s_1^2 + s_0^2}{2}}. \tag{2.2.4}$$

In (2.2.3) and (2.2.4), s_z^2 is the variance (either unweighted or weighted, depending on user specification) of the pth covariate in group z, and (w_0, w_1) are the specified balancing weights. Setting $w_0 = w_1 = 1$ corresponds to the unweighted mean differences. ASD and PSD are often displayed as column in the baseline characteristics table (known as the "Table 1") and visualized via a Love plot (also known as a forest plot) (Greifer, 2018). A rule of thumb for determining adequate balance is when ASD of all covariates is controlled within 0.1 (Austin and Stuart, 2015).

Multiple treatments

Li and Li (2019) extend the framework of balancing weights to multiple treatments. Assume that we have J ($J \geq 3$) treatment groups, and let Z_i stand for the treatment received by unit i, $Z_i \in \{1, \ldots, J\}$. We further define $D_{ij} = \mathbb{1}\{Z_i = j\}$ as a set of multinomial indicator, satisfying $\sum_{i=1}^J D_{ij} = 1$ for all j. Denote the potential outcome for unit i under treatment j as $Y_i(j)$, of which only the one corresponding to the received treatment, $Y_i = Y_i(Z_i)$, is observed. The generalized propensity score is the probability of receiving a potential treatment j given $\boldsymbol{X}$ (Imbens, 2000): $e_j(\boldsymbol{x}) = P(Z = j|\boldsymbol{X} = \boldsymbol{x})$, with the constraint that $\sum_{j=1}^J e_j(\boldsymbol{x}) = 1$.

To define the target estimand, let $m_j(\boldsymbol{x}) = \mathbb{E}[Y(j)|\boldsymbol{X} = \boldsymbol{x}]$ be the conditional expectation of the potential outcome in group j. For specified tilting function $h(\boldsymbol{x})$ and target density $g(\boldsymbol{x}) \propto f(\boldsymbol{x})h(\boldsymbol{x})$, the jth average potential outcome among the target population is

$$\mu_j^h = \mathbb{E}_g[Y(j)] = \frac{\mathbb{E}\{h(\boldsymbol{x})m_j(\boldsymbol{x})\}}{\mathbb{E}\{h(\boldsymbol{x})\}}. \tag{2.2.5}$$

Causal estimands can then be constructed in a general manner as contrasts based on μ_j^h. For example, the most commonly seen estimands in multiple treatments are the pairwise average treatment effects between groups j and j': $\tau_{j,j'}^h = \mu_j^h - \mu_{j'}^h$. This definition can be generalized to arbitrary linear contrasts. Denote $\boldsymbol{a} = (a_i, \cdots, a_J)$ as a contrast vector of length J. A general class of additive estimands is

$$\tau^h(\boldsymbol{a}) = \sum_{j=1}^J a_j \mu_j^h. \tag{2.2.6}$$

Specific choices for $\boldsymbol{a}$ with nominal and ordinal treatments can be found in Li and Li (2019). Similar to before, propensity score weighting analysis with multiple treatments rests on two assumptions: (1) *weak unconfoundedness*: $Y(j) \perp \mathbb{1}\{Z = j\}|\boldsymbol{X}$, for all j, and (2) *Overlap*: the generalized propensity score is bounded away from 0 and 1: $0 < e_j(\boldsymbol{x}) < 1$, for all j.

With multiple treatments, the tilting function $h(\boldsymbol{x})$ specifies the target population, estimand, and balancing weights. For a given $h(\boldsymbol{x})$, the balancing weights for the jth treatment group $w_j(\boldsymbol{x}) \propto h(\boldsymbol{x})/e_j(\boldsymbol{x})$. Then the Hájek estimator for μ_j^h is

$$\hat{\mu}_j^h = \frac{\sum_{i=1}^N w_j(\boldsymbol{x}_i)D_{ij}Y_i}{\sum_{i=1}^N w_j(\boldsymbol{x}_i)D_{ij}}. \tag{2.2.7}$$

Contrasts based on $\hat{\mu}_j^h$ can be obtained for any $\boldsymbol{a}$ to estimate the additive causal estimand $\tau^h(\boldsymbol{a})$. Of note, we only consider types of estimands that are transitive, and therefore the ATT estimands introduced in Lechner (2001) is not implemented. In parallel to binary treatments **PSweight** implements five types of balancing weights with multiple treatments: IPW, treated weights, OW, MW, and EW, and the corresponding target estimand of each weighting scheme is its pairwise (between each pair of treatments) counterpart in binary treatments.

Among all the weights, OW minimizes the total asymptotic variances of all pairwise comparisons, and has been shown to have the best finite-sample efficiency in estimating pairwise WATEs (Li and Li, 2019). Table 3 summarizes the target population, tilting function and balancing weight for multiple treatments that are available in **PSweight**.

To estimate the generalized propensity scores for multiple treatments, the default model in **PSweight** is a multinomial logistic model. **PSweight** also allows for externally estimated generalized propensity scores. Goodness-of-fit of the generalized propensity score model is assessed by

Table 3: Target populations, tilting functions, and the corresponding balancing weights for multiple treatments in **PSweight**.

Target population	Tilting function $h(x)$	Balancing weights $\{w_j(x), j = 1, \ldots, J\}$
Combined	1	$\{1/e_j(x)\}$
Treated (j'th group)	$e_{j'}(x)$	$\left\{e_{j'}(x)/e_j(x)\right\}$
Overlap	$\{\sum_{k=1}^{J} 1/e_k(x)\}^{-1}$	$\left\{\{\sum_{k=1}^{J} 1/e_k(x)\}^{-1}/e_j(x)\right\}$
Matching	$\min_k\{e_k(x)\}$	$\{\min_k\{e_k(x)\}/e_j(x)\}$
Entropy	$-\sum_{k=1}^{J} e_k(x)\log\{e_k(x)\}$	$\left\{-\sum_{k=1}^{J} e_k(x)\log\{e_k(x)\}/e_j(x)\right\}$

the resulting covariate balance, which is measured by the pairwise versions of the ASD and PSD. The detailed formula of these metrics can be found in Li and Li (2019). A common threshold for balance is that the maximum pairwise ASD or maximum PSD is below 0.1.

Propensity score trimming

Propensity score trimming excludes units with estimated (generalized) propensity scores close to zero (or one). It is a popular approach to address the extreme weights problem of IPW. **PSweight** implements the symmetric trimming rules in Crump et al. (2009) and Yoshida et al. (2018). Operationally, we allow users to specify a single cutoff δ on the estimated generalized propensity scores, and only includes units for analysis if $\min_j\{e_j(x)\} \in [\delta, 1]$. With binary treatments, the symmetric trimming rule reduces to $e(x) \in [\delta, 1 - \delta]$. The natural restriction $\delta < 1/J$ must be satisfied due to the constraint $\sum_{j=1}^{J} e_j(x) = 1$. To avoid specifying an arbitrary trimming threshold δ, **PSweight** also implements the optimal trimming rules of Crump et al. (2009) and Yang et al. (2016), which minimizes the (total) asymptotic variance(s) for estimating the (pairwise) ATE among the class of all trimming rules. OW can be viewed as a continuous version of trimming because it smoothly down-weigh the units with propensity scores close to 0 or 1, and thus avoids specifying a threshold.

Augmented weighting estimators

PSweight also implements augmented weighting estimators, which augment a weighting estimator by an outcome regression and improves the efficiency. With IPW, the augmented weighting estimator is known as the doubly-robust estimator (Lunceford and Davidian, 2004; Bang and Robins, 2005; Funk et al., 2011). With binary treatments, the augmented estimator with general balancing weights are discussed Hirano et al. (2003) and Mao et al. (2018). Below, we briefly outline the form of this estimator with multiple treatments. Recall the conditional mean of $Y_i(j)$ given X_i and treatment $Z_i = j$ as $m_j(x_i) = \mathbb{E}[Y_i(j)|X_i = x_i] = \mathbb{E}[Y_i|X_i = x_i, Z_i = j]$. This conditional mean can be estimated by generalized linear models, kernel estimators, or machine learning models. **PSweight** by default employs the generalized linear models, but also allows estimated values from other routines. When $m_j(x_i)$ is estimated by generalized linear models, **PSweight** currently accommodates three types of outcomes: continuous, binary and count outcomes (with or without an offset), using the canoncal link function.

With a pre-specified tilting function, the augmented weighting estimator for group j is

$$\hat{\mu}_j^{h,\text{aug}} = \frac{\sum_{i=1}^{N} w_j(x_i)D_{ij}\{Y_i - m_j(x_i)\}}{\sum_{i=1}^{N} w_j(x_i)D_{ij}} + \frac{\sum_{i=1}^{N} h(x_i)m_j(x_i)}{\sum_{i=1}^{N} h(x_i)}. \tag{2.2.8}$$

The first term of (2.2.8) is the Hájek estimator of the regression residuals, and the second term is the standardized average potential outcome (a g-formula estimator). With IPW, (2.2.8) is consistent to $\mathbb{E}[Y(j)]$ when either the propensity score model or the outcome model is correctly specified, but not necessarily both. For other balancing weights, (2.2.8) is consistent to the WATE when the propensity model is correctly specified, regardless of outcome model specification. When both models are correctly specified, (2.2.8) achieves the lower bound of the variance for regular and asymptotic linear estimators (Robins et al., 1994; Hirano et al., 2003; Mao et al., 2018).

Ratio causal estimands

With binary and count outcomes, ratio causal estimands are often of interest. Using notation from the multiple treatments as an example, once we use weighting to obtain estimates for the set of average

potential outcomes $\{\mu_j^h, j = 1, \ldots, J\}$, we can directly estimate the causal relative risk (RR) and causal odds ratio (OR), defined as

$$\tau_{j,j'}^{h,\mathrm{RR}} = \frac{\mu_j^h}{\mu_{j'}^h}, \qquad \tau_{j,j'}^{h,\mathrm{OR}} = \frac{\mu_j^h/(1-\mu_j^h)}{\mu_{j'}^h/(1-\mu_{j'}^h)}. \tag{2.2.9}$$

Here the additive estimand $\tau_{j,j'}^{h,\mathrm{RD}} = \mu_j^h - \mu_{j'}^h$ is the causal risk difference (RD). **PSweight** supports a class of ratio estimands for any given contrasts $\boldsymbol{a}$. Specifically, we define the log-RR type parameters by

$$\lambda^{h,\mathrm{RR}}(\boldsymbol{a}) = \sum_{j=1}^{J} a_j \log\left(\mu_j^h\right), \tag{2.2.10}$$

and the log-OR type parameters by

$$\lambda^{h,\mathrm{OR}}(\boldsymbol{a}) = \sum_{j=1}^{J} a_j \left\{\log\left(\mu_j^h\right) - \log\left(1 - \mu_j^h\right)\right\}. \tag{2.2.11}$$

With nominal treatments, the contrast vector $\boldsymbol{a}$ can be specified to encode pairwise comparisons in the log scale (as in (2.2.10)) or in the log odds scale (as in (2.2.11)), in which case $\exp\{\lambda^{h,\mathrm{RR}}(\boldsymbol{a})\}$ and $\exp\{\lambda^{h,\mathrm{OR}}(\boldsymbol{a})\}$ become the causal RR and causal OR in (2.2.9). User-specified contrasts $\boldsymbol{a}$ can provide a variety of nonlinear estimands. For example, when $J = 3$, with $\boldsymbol{a} = (1, -2, 1)^T$ one can use **PSweight** to assess the equality of two consecutive causal RR: $H_0 : \mu_3^h/\mu_2^h = \mu_2^h/\mu_1^h$.

Variance and interval estimation

PSweight by default implements the empirical sandwich variance for propensity score weighting estimators (Lunceford and Davidian, 2004; Li et al., 2019; Mao et al., 2018) based on the M-estimation theory (Stefanski and Boos, 2002). The variance adjusted for the uncertainty in estimating the propensity score and outcome models, and are sometime referred to as the nuisance-adjusted sandwich variance. Below we illustrate the main steps with multiple treatments and general balancing weights. Write $\boldsymbol{\theta} = \left(\nu_1, \ldots, \nu_J, \eta_1, \ldots, \eta_J, \boldsymbol{\beta}^T, \boldsymbol{\alpha}^T\right)^T$ as the collection of parameters to be estimated. Then $\left\{\hat{\mu}_j^{h,\mathrm{aug}} = \hat{\nu}_j + \hat{\eta}_j : j = 1, \ldots, J\right\}$ jointly solve

$$\sum_{i=1}^{N} \Psi_i(\boldsymbol{\theta}) = \sum_{i=1}^{N} \begin{pmatrix} w_1(\boldsymbol{x}_i) D_{i1}\{Y_i - m_1(\boldsymbol{x}_i; \boldsymbol{\alpha}) - \nu_1\} \\ \vdots \\ w_J(\boldsymbol{x}_i) D_{iJ}\{Y_i - m_J(\boldsymbol{x}_i; \boldsymbol{\alpha}) - \nu_J\} \\ h(\boldsymbol{x}_i)\{m_1(\boldsymbol{x}_i; \boldsymbol{\alpha}) - \eta_1\} \\ \vdots \\ h(\boldsymbol{x}_i)\{m_J(\boldsymbol{x}_i; \boldsymbol{\alpha}) - \eta_J\} \\ S_{\boldsymbol{\beta}}(Z_i, \boldsymbol{x}_i; \boldsymbol{\beta}) \\ S_{\boldsymbol{\alpha}}(Y_i, Z_i, \boldsymbol{x}_i; \boldsymbol{\alpha}) \end{pmatrix} = \boldsymbol{0},$$

where $S_{\boldsymbol{\beta}}(Z_i, \boldsymbol{x}_i; \boldsymbol{\beta})$ and $S_{\boldsymbol{\alpha}}(Y_i, Z_i, \boldsymbol{x}_i; \boldsymbol{\alpha})$ are the score functions of the propensity score model and the outcome model. The empirical sandwich variance estimator is

$$\widehat{\mathbb{V}}(\hat{\boldsymbol{\theta}}) = \left\{\sum_{i=1}^{N} \frac{\partial}{\partial \boldsymbol{\theta}^T} \Psi_i(\hat{\boldsymbol{\theta}})\right\}^{-1} \left\{\sum_{i=1}^{N} \Psi_i(\hat{\boldsymbol{\theta}}) \Psi_i^T(\hat{\boldsymbol{\theta}})\right\} \left\{\sum_{i=1}^{N} \frac{\partial}{\partial \boldsymbol{\theta}} \Psi_i^T(\hat{\boldsymbol{\theta}})\right\}^{-1}.$$

Because $\hat{\mu}_j^{h,\mathrm{aug}} = \hat{\nu}_j + \hat{\eta}_j$, the variance of arbitrary linear contrasts based on the average potential outcomes can be easily computed by applying the Delta method to the joint variance $\widehat{\mathbb{V}}(\hat{\boldsymbol{\theta}})$. For the Hájek weighting estimators, variance is estimated by removing $S_{\boldsymbol{\alpha}}(Y_i, Z_i, \boldsymbol{x}_i; \boldsymbol{\alpha})$ as well as the components involving $m_j(\boldsymbol{x}_i; \boldsymbol{\alpha})$ in $\Psi_i(\boldsymbol{\theta})$. Finally, when propensity scores and potential outcomes are not estimated through the generalized linear model or are supplied externally, or MW are used (since the tilting function is not everywhere differentiable), **PSweight** ignores the uncertainty in estimating $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ and removes $S_{\boldsymbol{\beta}}(Z_i, \boldsymbol{x}_i; \boldsymbol{\beta})$ and $S_{\boldsymbol{\alpha}}(Y_i, Z_i, \boldsymbol{x}_i; \boldsymbol{\alpha})$ in $\Psi_i(\boldsymbol{\theta})$ in the calculation of the empirical sandwich variance. Based on the estimated variance, **PSweight** computes the associated symmetric confidence intervals and p-values via the normal approximation.

For ratio causal estimands, **PSweight** applies the logarithm transformation to improve the accuracy of the normal approximation (Agresti, 2003). For estimating the variance of causal RR, we first obtain the joint variance of $\left(\log\left(\hat{\mu}_1^{h,\mathrm{aug}}\right), \ldots \log\left(\hat{\mu}_J^{h,\mathrm{aug}}\right)\right)^T$ using the Delta method, and then estimate the

variance of $\lambda^{h,\text{RR}}(\boldsymbol{a})$. Once the symmetric confidence intervals are obtained for $\lambda^{h,\text{RR}}(\boldsymbol{a})$ using the normal approximation, we can exponentiate the upper and lower confidence limits to derive the asymmetric confidence intervals for the causal RR. Confidence intervals for the causal OR are computed similarly.

PSweight also allows using bootstrap to estimate variances, which can be much more computationally intensive than the closed-form sandwich estimator but sometimes give better finite-sample performance in small samples. By default, **PSweight** resamples $R = 50$ bootstrap replicates with replacement. For each replicate, the weighting estimator (2.2.7) or the augmented weighting estimtor (2.2.8) is implemented, providing R estimates of the J average potential outcomes (an $R \times J$ matrix). Then for any contrast vector $\boldsymbol{a} = (a_1, \cdots, a_J)^T$, **PSweight** obtains R bootstrap estimates:

$$\hat{\mathbb{T}}^h(\boldsymbol{a})_{bootstrap} = \left\{ \hat{\tau}^h(\boldsymbol{a})_1 = \sum_{j=1}^{J} a_j \hat{\mu}_{j,1}^h, \; \ldots, \hat{\tau}^h(\boldsymbol{a})_R = \sum_{j=1}^{J} a_j \hat{\mu}_{j,R}^h \right\}.$$

The sample variance of $\hat{\mathbb{T}}^h(\boldsymbol{a})_{bootstrap}$ is reported by **PSweight** as the bootstrap variance; the lower and upper 2.5% quantiles of $\hat{\mathbb{T}}^h(\boldsymbol{a})_{bootstrap}$ form the 95% bootstrap interval estimate

3 Overview of package

The **PSweight** 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. We briefly describe the two modules below.

Design module

PSweight offers the `SumStat()` function to visualize the distribution of the estimated propensity scores, to assess the balance of covariates under different weighting schemes, and to characterize the weighted target population. It uses the following code snippet:

```
SumStat(ps.formula, ps.estimate = NULL, trtgrp = NULL, Z = NULL, covM = NULL,
    zname = NULL,  xname = NULL, data = NULL,weight = "overlap", delta = 0,
    method = "glm", ps.control = list())
```

By default, the (generalized) propensity scores are estimated by the (multinomial) logistic regression, through the argument `ps.formula`. Alternatively, `gbm()` functions in the **gbm** package (Greenwell et al., 2019) or the `SuperLearner()` function in the **SuperLearner** package (Polley et al., 2019) can also be called by using `method = "gbm"` or `method = "SuperLearner"`. Additional parameters of those functions can be supplied through the `ps.control` argument. The argument `ps.estimate` supports estimated propensity scores from external routines. `SumStat()` produces a SumStat object, with estimated propensity scores, unweighted and weighted covariate means for each treatment group, balance diagnostics, and effective sample sizes (defined in (Li and Li, 2019)). We then provide a `summary.SumStat()` function, which takes the SumStat object and summarizes weighted covariate means by treatment groups and the between-group differences in either ASD or PSD. The default options in `weighted.var = TRUE` and `metric = "ASD"` yield ASD based on weighted standard deviations in Austin and Stuart (2015). The weighted covariate means can be used to build a baseline characteristics "Table 1" to illustrate the target population where trimming or balancing weights are applied.

```
summary(object, weighted.var = TRUE, metric = "ASD")
```

Diagnostics of propensity score models can be visualized with the `plot.SumStat()` function. It takes the SumStat object and produces a balance plot (`type = "balance"`) based on the ASD and PSD. A vertical dashed line can be set by the `threshold` argument, with a default value equal to 0.1. The `plot.SumStat()` function can also supply density plot (`type = "density"`), or histogram (`type = "hist"`) of the estimated propensity scores. The histogram, however, is only available for the binary treatment case. The plot function is implemented as follows:

```
plot(x, type = "balance", weighted.var = TRUE, threshold = 0.1, metric = "ASD")
```

In the design stage, propensity score trimming can be carried out with the `PStrim()` function. The trimming threshold `delta` is set to 0 by default. `PStrim()` also enables optimal trimming rules (`optimal = TRUE`) that give the most statistically efficient (pairwise) subpopulation ATE, among all

Table 4: Functions in the design module of **PSweight**.

Function	Description
SumStat()	Generate a SumStat object with information of propensity scores and weighted covariate balance
summary.SumStat()	Summarize the SumStat object and return weighted covariate means by treatment groups and weighted or unweighted between-group differences in ASD or PSD
plot.SumStat()	Plot the distribution of propensity scores or weighted covariate balance metrics from the SumStat object
PStrim()	Trim the data set based on estimated propensity scores

possible trimming rules. A trimmed data set along with a summary of trimmed cases will be returned by PStrim(). This function is given below:

```
PStrim(data, ps.formula = NULL, zname = NULL, ps.estimate = NULL, delta = 0,
    optimal = FALSE, method = "glm", ps.control = list())
```

Alternatively, trimming is also anchored in the SumStat() function with the delta argument. All functions in the design module are summarized in Table 4.

Analysis module

The analysis module of **PSweight** includes two functions: PSweight() and summary.PSweight(). The PSweight() function estimates the average potential outcomes in the target population, $\{\mu_j^h, j = 1, \dots, J\}$, and the associated variance-covariance matrix. By default, the empirical sandwich variance is implemented, but bootstrap variance can be obtained with the argument bootstrap = TRUE). The weight argument can take "IPW", "treated", "overlap", "matching" or "entropy", corresponding to the weights introduced in Tables 2 and 3. More detailed descriptions of each input argument in the PSweight() function can be found in Table 5. A typical PSweight() code snippet looks like

```
PSweight(ps.formula, ps.estimate, trtgrp, zname, yname, data, weight = "overlap",
    delta = 0, augmentation = FALSE, bootstrap = FALSE, R = 50, out.formula = NULL,
    out.estimate = NULL, family = "gaussian", ps.method = "glm", ps.control = list(),
    out.method = "glm", out.control = list())
```

Similar to the design module, the summary.PSweight() function synthesizes information from the PSweight object for statistical inference. A typical code snippet looks like

```
summary(object, contrast, type = "DIF", CI = TRUE)
```

In the summary.PSweight() 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 as in equation (2.2.10); type = "OR" specifies the contrast on the log odds scale as in equation (2.2.11). Confidence intervals and p-values are obtained using normal approximation and reported by the summary.PSweight() function. The argument contrast represents a contrast vector $\boldsymbol{a}$ or matrix with multiple contrast row vectors. If contrast is not specified, summary.PSweight() provides all pairwise comparisons of the average potential outcomes. By default, confidence interval is printed (CI = TRUE); alternatively, one can print the test statistics and p-values by CI = FALSE.

4 Case study with the NCDS data

We demonstrate **PSweight** in a case study that estimates the causal effect of educational attainment on hourly wage, based on the National Child Development Survey (NCDS) data. The National Child Development Survey (NCDS) is a longitudinal study on children born in the United Kingdom (UK) in 1958 [1]. NCDS collected information such as educational attainment, familial backgrounds, and socioeconomic and health well being on $17,415$ individuals. We followed Battistin and Sianesi (2011) to pre-process the data and obtain a subset of 3,642 males employed in 1991 with complete educational attainment and wage information for analysis. For illustration, we use the Multiple Imputation by Chained Equations in (Buuren and Groothuis-Oudshoorn, 2010) to impute missing covariates and

---

[1] https://cls.ucl.ac.uk/cls-studies/1958-national-child-development-study/

Table 5: Arguments for function `PSweight()` in the analysis module of **PSweight**.

Argument	Description	Default
`ps.formula`	A symbolic description of the propensity score model.	–
`ps.estimate`	An optional matrix or data frame with externally estimated (generalized) propensity scores for each observation; can also be a vector with binary treatments.	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"
`delta`	Trimming threshold for (generalized) propensity scores.	0
`augmentation`	Logical value of whether augmented weighting estimators should be used.	FALSE
`bootstrap`	Logical value of whether bootstrap is used to estimate the standard error	FALSE
`R`	Number of bootstrap replicates if `bootstrap = TRUE`	50
`out.formula`	A symbolic description of the outcome model to be estimated when `augmentation = TRUE`	
`out.estimate`	An optional matrix or data frame containing externally estimated potential outcomes for each observation under each treatment level.	NULL
`family`	A description of the error distribution and canonical link function to be used in the outcome model if `out.formula` is provided	"gaussian"
`ps.method`	a character to specify the method for propensity model.	"glm"
`ps.control`	A list to specify additional options when method is set to "gbm" or "SuperLearner".	list()
`out.method`	A character to specify the method for outcome model.	"glm"
`out.control`	A list to specify additional options when methodout is set to "gbm" or "SuperLearner".	list()

obtain a single imputed data set for all subsequent analysis.[2] The outcome variable wage is log of the gross hourly wage in Pound. The treatment variable is educational attainment. For the multiple treatment case, To start with, we created `Dmult` as a treatment variable with three levels: ">=A/eq", "O/eq" and "None", representing advanced qualification (1,806 individuals), intermediate qualification (941 individuals) and no qualification (895 individuals). We consider twelve pre-treatment covariates or potential confounders. The variable `white` indicates whether an individual identified himself as white race; `scht` indicates the school type they attended at age 16; `qmab` and `qmab2` are math test scores at age 7 and 11; `qvab` and `qvab2` are two reading test scores at age 7 and 11; `sib_u` stands for the number of siblings; `agepa` and `agema` are the ages of parents in year 1,974; in the same year, the employment status of mother `maemp` was also collected; `paed_u` and `maed_u` are the years of education for parents. For simplicity, we will focus on IPW and the three types of weights that improve covariate overlap: OW, MW and EW.

Estimating generalized propensity scores and balance assessment

We use `Dmult`, the three-level variable, as the treatment of interest. About one half of the population attained advanced academic qualification, there are approximately equal numbers of individuals with intermediate academic qualification or no academic qualification. To illustrate the estimation and inference for ratio estimands, we also introduce a binary outcome of wage, `wagebin`. The dichotomized wage was obtained with the cutoff of the average hourly wage of actively employed British male aged 30-39 in 1991[3]. The averaged hourly wage is 8.23, and we take $\log(8.23) \approx 2.10$ as the cutoff. Among the study participants, we observe 1610 and 2032 individuals above and below the average, and we are interested in estimating the pairwise (weighted) average treatment effect of the academic qualification for obtaining above-average hourly wage.

We specify a multinominal regression model, `ps.mult`, to estimate the generalized propensity scores.

```
ps.mult <- Dmult ~ white + maemp + as.factor(scht) + as.factor(qmab)
```

---

[2]Ten out of twelve pre-treatment covariates we considered have missingness. The smallest missingness proportion is 4.9% and the largest missingness proportion is 17.2%. We considered one imputed complete data set for illustrative purposes, but note that a more rigorous analysis could proceed by combining analyses from multiple imputed data sets via the Rubin's rule.

[3]https://www.ons.gov.uk/employmentandlabourmarket/peopleinwork/earningsandworkinghours/

```
        as.factor(qmab2) + as.factor(qvab) + as.factor(qvab2) + paed_u + maed_u +
        agepa + agema + sib_u + paed_u * agepa + maed_u * agema
```

Then we obtain the propensity score estimates and assess weighted covariate balance with the `SumStat()` function.

```
bal.mult <-  SumStat(ps.formula = ps.mult,
    weight = c("IPW", "overlap", "matching", "entropy"), data = NCDS)
plot(bal.mult, type = "density")
```
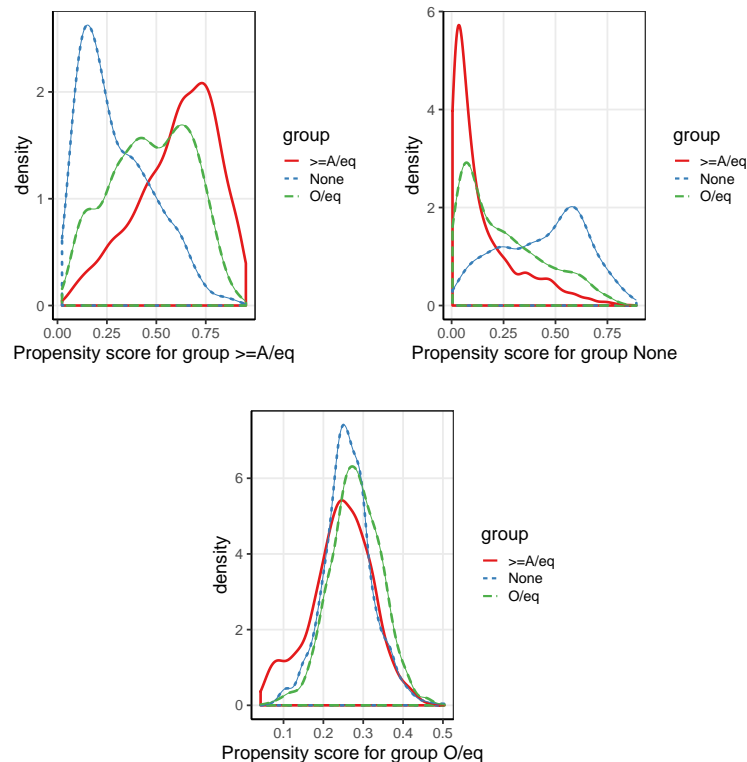

Figure 1: Density plots of estimated generalized propensity scores with respect to the three-level treatment variable `Dmult` generated by `plot.SumStat()` function in the **PSweight** package.

The distributions of generalized propensity scores are given in Figure 1 (in alphabetic order of the names of treatment groups). For the generalized propensity score to receive the advanced qualification (">=A/eq") or no qualification ("None"), there is a mild lack of overlap due to separation of the group-specific distribution. Since `bal.mult` includes four weighting schemes, we plot the maximum pairwise ASD and assess the (weighted) covariate balance in a single Love plot.

```
plot(bal.mult, metric = "ASD")
```

The covariates are imbalanced across the three groups prior to any weighting. Although IPW can generally improve covariate balance, the maximum pairwise ASD still ocassionally exceeds the threshold 0.1 due to lack of overlap. In contrast, OW, MW and EW all emphasize the subpopulation with improved overlap and provide better balance across all covariates.

Generalized propensity score trimming

The **PSweight** package can perform trimming based on (generalized) propensity scores. As IPW does not adequately balance the covariates across the three groups in Figure 2, we explore trimming as a way to improve balance for IPW. There are two types of trimming performed by the **PSweight** package: (1) symmetric trimming that removes units with extreme (generalized propensity scores) (Crump et al., 2009; Yoshida et al., 2018) and (2) optimal trimming that provides the most efficient IPW estimator for estimating (pairwise) ATE (Crump et al., 2009; Yang et al., 2016). Specifically, the symmetric trimming is supported by both the `SumStat()` and `PSweight()` functions through the `delta` argument. Both functions refit the (generalized) propensity score model after trimming
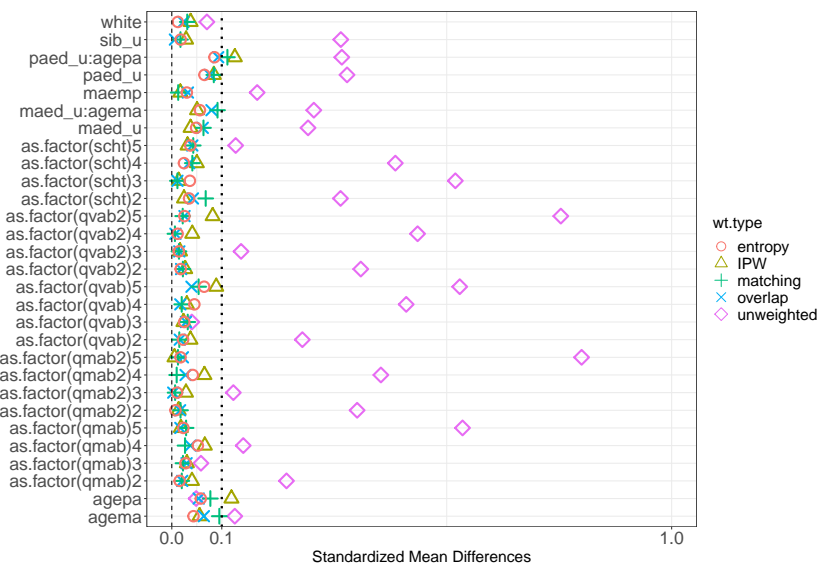
Figure 2: Love plot with the three-level treatment variable `Dmult` using the maximum pairwise ASD metric, generated by `plot.SumStat()` function in the **PSweight** package.

following the recommendations in Li et al. (2019). We also provide a stand-alone `PStrim` function that performs both symmetric trimming and optimal trimming. Following Yoshida et al. (2018), with three treatment groups, we exclude all individuals with the estimated generalized propensity scores less than $\delta = 0.067$. This threshold removes a substantial amount of individuals in the advanced qualification group (information can be pulled from the `trim` element in the `SumStat` object). As discussed in Yoshida et al. (2018), propensity trimming could improve the estimation of ATE and ATT, but barely have any effect for estimation of ATO and ATM. Evidently, Figure 3 indicates that IPW controls all pairwise ASD within 10% in the trimmed sample. Trimming had nearly no effect on the weighted balance for OW, MW and EW.

```
bal.mult.trim <- SumStat(ps.formula = ps.mult, weight = c("IPW", "overlap", "matching",
    "entropy"), data = NCDS, delta = 0.067)

bal.mult.trim

1050 cases trimmed,  2592 cases remained

trimmed result by trt group:
        >=A/eq None O/eq
trimmed     778   71  201
remained   1028  824  740

weights estimated for:  IPW overlap matching entropy

plot(bal.mult.trim,metric = "ASD")
```

Alternatively, if one does not specify the trimming threshold, the `PStrim` function supports the optimal trimming procedure that identifies the optimal threshold based on data. Example syntax is given as follows. By pulling out the summary statistics for trimming, we can see that optimal trimming excludes 27%, 9% and 2% of the individuals among those with advanced qualification, intermediate qualification and no qualification, respectively. The exclusion is more conservative compared to symmetric trimming with $\delta = 0.067$. However, the resulting covariate balance after optimal trimming is similar to Figure 3 and omitted.

```
PStrim(ps.formula = ps.mult, data = NCDS, optimal = TRUE)

        >=A/eq None O/eq
trimmed     479   21   82
remained   1327  874  859
```
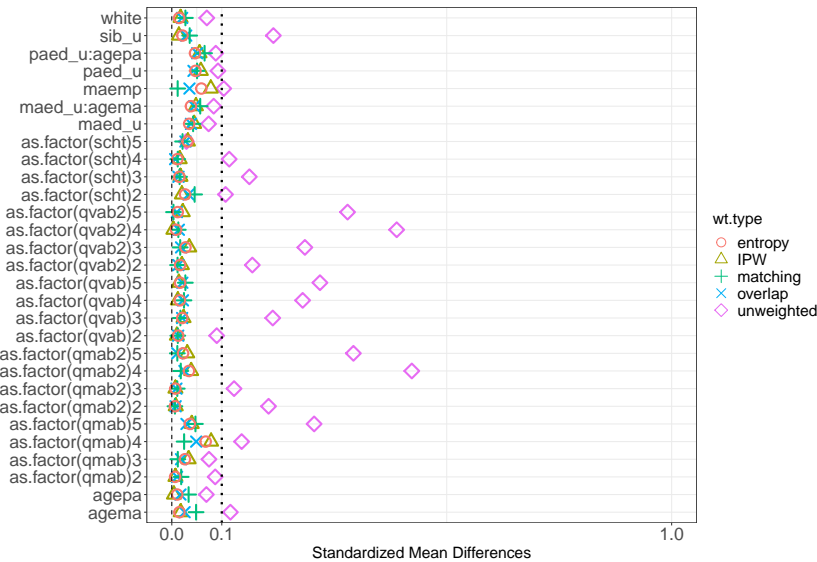
Figure 3: Love plot with the three-level treatment variable `Dmult` using the maximum pairwise ASD metric, after symmetric trimming with $\delta = 0.067$. This plot is generated by `plot.SumStat()` function in the **PSweight** package.

Estimation and inference of pairwise (weighted) average treatment effects

For illustration, we estimate the ratio estimands using the binary outcome `wagebin`. For illustration, we will only estimate the causal effects based on the data without trimming, and the analysis with the trimmed data follows the exact same steps. Based on the multinomial logistic propensity score model, we obtain the pairwise causal RR among the combined population via IPW.

```
ate.mult <- PSweight(ps.formula = ps.mult, yname = "wagebin", data = NCDS,
    weight = "IPW")}
contrasts.mult <- rbind(c(1,-1, 0), c(1, 0,-1), c(0, -1, 1))
sum.ate.mult.rr <- summary(ate.mult, type = "RR", contrast = contrasts.mult)
sum.ate.mult.rr

Closed-form inference:

Inference in log scale:
Original group value:  >=A/eq, None, O/eq

Contrast:
          >=A/eq None O/eq
Contrast 1     1   -1    0
Contrast 2     1    0   -1
Contrast 3     0   -1    1


          Estimate Std.Error      lwr      upr  Pr(>|z|)
Contrast 1 0.607027  0.115771  0.380120 0.83393 1.577e-07 ***
Contrast 2 0.459261  0.052294  0.356767 0.56176 < 2.2e-16 ***
Contrast 3 0.147766  0.121692 -0.090746 0.38628    0.2246
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

By providing the appropriate contrast matrix, we obtain all pairwise comparisons of the average potential outcomes on the log scale with the `summary.PSweight()` function, and estimate $\lambda^{h,\mathrm{RR}}(\boldsymbol{a})$ for contrast vector $\boldsymbol{a}$. The p-values provides statistical evidence against the weak causal null H_0 : $\lambda^{h,\mathrm{RR}}(\boldsymbol{a}) = 0$. It is found that, among the combined population, the proportion that receives an above-average hourly wage when everyone attains advanced qualification is $\exp(0.607) = 1.83$ times that when everyone attains no academic qualification. Further, the proportion that receives an above-average hourly wage when everyone attains advanced qualification is $\exp(0.459) = 1.58$ times that when everyone attains intermediate qualification. Both effects are significant at the 0.05 levels and provides strong evidence against the corresponding causal null (p-value < 0.001). However, if

everyone attains intermediate qualification, the proportion that receives an above-average hourly wage is only slightly higher compared to without qualification, with a p-value exceeding 0.05. To directly report the causal RR and its confidence intervals, we can simply exponentiate the point estimate and confidence limits provided by the summary.PSweight() function.

```
exp(sum.ate.mult.rr$estimates[,c(1,4,5)])

            Estimate        lwr       upr
Contrast 1 1.834968 1.4624601 2.302358
Contrast 2 1.582904 1.4287028 1.753749
Contrast 3 1.159241 0.9132496 1.471493
```

Focusing on the target population that has the most overlap in the observed covariates, we further use the OW to estimate the pairwise causal RR. OW theoretically provides the best internal validity for pairwise comparisons; Figure 3 also indicates that OW achieves better covariate balance among the overlap population. Exponentiating the results provided by the summary.PSweight() function, we observe each pairwise causal RR has a larger effect size among the overlap weighted population. Interestingly, among the overlap population, the proportion that receives an above-average hourly wage when everyone attains intermediate qualification becomes approximately 1.55 times that when everyone attains no academic qualification, and the associated 95% CI excludes the null. Moreover, the standard errors for the pairwise comparisons are smaller when using OW versus IPW, implying that OW analysis generally corresponds to increased power by focusing on a population with equipoise. We repeat the analysis using both MW and EW; the results are similar to OW for this analysis and therefore omitted for brevity.

```
ato.mult <- PSweight(ps.formula = ps.mult, yname = "wagebin", data = NCDS,
    weight = "overlap")
sum.ato.mult.rr <- summary(ato.mult, type = "RR", contrast = contrasts.mult)
exp(sum.ato.mult.rr$estimates[,c(1,4,5)])

            Estimate       lwr       upr
Contrast 1 2.299609 1.947140 2.715882
Contrast 2 1.527931 1.363092 1.712705
Contrast 3 1.505048 1.257180 1.801785
```

The above output suggests that among the overlap population, the causal RR for comparing advanced qualification to intermediate qualification is similar in magnitude to that for comparing intermediate qualification to no qualification. We can formally test for the equality of two consecutive causal RR based on the null hypothesis $H_0 : \mu_3^h/\mu_2^h = \mu_2^h/\mu_1^h$. Operationally, we need to specify the corresponding contrast vector contrast = c(1,1,-2). The p-value for testing this null is 0.91 (output omitted for brevity), and suggests a lack of evidence against the equality of consecutive causal RR at the 0.05 level.

```
summary(ato.mult, type = "RR", contrast = c(1, 1, -2), CI = FALSE)
```

With the binary outcome wagebin, we can also estimate the pairwise causal OR among a specific target population. For example, using OW, the causal conclusions regarding the effectiveness due to attaining academic qualification do not change, because all three 95% confidence intervals exclude null. However, the pairwise causal OR appear larger than the pairwise causal RR. This is expected because our outcome of interest is not uncommon (Nurminen, 1995). For rare outcomes, causal OR approximates causal RR.

```
sum.ato.mult.or <- summary(ato.mult, type = "OR", contrast = contrasts.mult)
exp(sum.ato.mult.or$estimates[,c(1,4,5)])

            Estimate       lwr       upr
Contrast 1 3.586050 2.841383 4.525879
Contrast 2 2.050513 1.696916 2.477791
Contrast 3 1.748855 1.375483 2.223578
```

As a final step, we illustrate how to combine OW with outcome regression and estimate the pairwise causal RR among the overlap population. We use the same set of covariates in the binary outcome regression model.

```
 out.wagebin <- wagebin ~ white + maemp + as.factor(scht) + as.factor(qmab) +
    as.factor(qmab2) + as.factor(qvab) + as.factor(qvab2) + paed_u + maed_u +
    agepa + agema + sib_u + paed_u * agepa + maed_u * agema
```

Loading this outcome regression formula into the `PSweight()` function, and specifying `family = "binomial"` to indicate the type of outcome, we obtain the augmented overlap weighting estimates on the log RR scale. Exponentiating the point estimates and confidence limits, one reports the pairwise causal RR. The pairwise causal RR reported by the augmented OW estimator is similar to that reported by the simple OW estimator; further, the width of the confidence interval is also comparable before and after outcome augmentation, and the causal conclusions based on pairwise RR remain the same. The similarity between simple and augmented OW estimators implies that OW itself may already be efficient.

```
ato.mult.aug <- PSweight(ps.formula = ps.mult, yname = "wagebin", data = NCDS,
    augmentation = TRUE, out.formula = out.wagebin, family = "binomial")
sum.ato.mult.aug.rr <- summary(ato.mult.aug, type = "RR", contrast = contrasts.mult)
exp(sum.ato.mult.aug.rr$estimates[,c(1,4,5)])

          Estimate      lwr      upr
Contrast 1 2.310628 1.957754 2.727105
Contrast 2 1.540176 1.375066 1.725111
Contrast 3 1.500237 1.253646 1.795331
```

Using machine learning to estimate propensity scores and potential outcomes

As an alternative to the default generalized linear models, we can use more advanced machine learning models to estimate propensity scores and potential outcomes. Flexible propensity score and outcome estimation has been demonstrated to reduce bias due to model misspecification, and potentially improve covariate balance (Lee et al., 2010; Hill, 2011; McCaffrey et al., 2013). This can be achieved in **PSweight** for both balance check and constructing weighted estimator by specifying the method as the generalized boosted model (GBM) or the super learner methods. Additional model specifications for these methods can be supplied through `ps.control` and `out.control`. Machine learning models that are included in neither gbm nor SuperLearner could be estimated externally and then imported through the `ps.estimate` and `out.estimate` arguments. These two arguments broaden the utility of **PSweight** where any externally generated estimates of propensity scores and potential outcomes models can be easily incorporated.

We now illustrate the use of GBM as an alternative of the default generalized linear models. For simplicity, this illustration is based on binary education. Specifically, we created `Dany` to indicate whether one had attained any academic qualification. There are 2,399 individuals that attained academic qualification, and 1,243 individuals without any. GBM is a family of non-parametric tree-based regressions that allow for flexible non-linear relationships between predictors and outcomes (Friedman et al., 2000). The following propensity model formula is specified; the formula does not include interactions terms because boosted regression is already capable of capturing non-linear effects and interactions (McCaffrey et al., 2004). In this illustration, we use the AdaBoost (Freund and Schapire, 1997) algorithm to fit the propensity model through the control setting, `ps.control=list(distribution = "adaboost")`. We use the default values for other model parameters such as the number of trees (`n.trees = 100`), interaction depth (`interaction.depth = 1`), the minimum number of observations in the terminal nodes (`n.minobsinnode = 1`), shrinkage reduction (`shrinkage = 0.1`), and bagging fraction (`shrinkage = 0.5`). Alternative values for these parameters could also be passed through `ps.control`.

```
ps.any.gbm <- Dany ~ white + maemp + as.factor(scht) + as.factor(qmab) +
    as.factor(qmab2) + as.factor(qvab) + as.factor(qvab2) + paed_u +  maed_u+
    agepa + agema + sib_u
bal.any.gbm <-SumStat(ps.formula = ps.any.gbm, data= NCDS,  weight = "overlap",
    method = "gbm", ps.control = list(distribution = "adaboost"))
```

The balance check through `plot.SumStat()` suggests substantial improvement in covariate balance with SMD of all covariates below 0.1 after weighting. After assessing balance and confirming the adequacy of the propensity score model, we further fit the outcome model using GBM with the default logistic regression and parameters. In the `PSweight()` function, we can specify both `ps.method = "gbm"` and `out.method = "gbm"` and leave the `out.control` argument as default. The detailed code and summary of the output is in below. Here we redefine the propensity score model without interaction terms because GBM considers interactions between covariates by default. The results using GBM, in this example, are very similar to those using generalized linear models (results omitted).

```
out.wage.gbm <- wage ~ white + maemp + as.factor(scht) + as.factor(qmab) +
    as.factor(qmab2) + as.factor(qvab) + as.factor(qvab2) + paed_u +
```

```
    maed_u + agepa + agema + sib_u
ato.any.aug.gbm <- PSweight(ps.formula = ps.any.gbm, yname = "wagebin",
    data = NCDS,  augmentation = TRUE, out.formula = out.wage.gbm,
    ps.method = "gbm", ps.control = list(distribution = "adaboost"),
    out.method = "gbm")
summary(ato.any.aug.gbm, CI = FALSE)

Closed-form inference:

Original group value:  0, 1

Contrast:
           0 1
Contrast 1 -1 1

          Estimate Std.Error z value  Pr(>|z|)
Contrast 1 0.186908  0.018609  10.044 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

5   Summary

Propensity score weighting is an important tool for causal inference and comparative effectiveness research. This paper introduces the **PSweight** package and demonstrates its functionality with the NCDS data example in the context of binary and multiple treatment groups. In addition to providing easy-to-read balance statistics and plots to aid the design of observational studies, the **PSweight** offers point and variance estimation with a variety of weighting schemes for the (weighted) average treatment effects on both the additive and ratio scales. These weighting schemes include the optimal overlap weights recently introduced in Li et al. (2018) and Li and Li (2019), and could help generate valid causal comparative effectiveness evidence among the population at equipoise.

Although propensity score weighting has been largely developed in observational studies, it is also an important tool for covariate adjustment in randomized controlled trials (RCTs). Williamson et al. (2014) showed that IPW can reduce the variance of the unadjusted difference-in-means treatment effect estimator in RCTs, and Shen et al. (2014) proved that the IPW estimator is semiparametric efficient and asymptotically equivalent to the analysis of covariance (ANCOVA) estimator (Tsiatis et al., 2008). Zeng et al. (2020) generalized these results of IPW to the family of balancing weights. Operationally, there is no difference in implementing propensity score weighting between RCTs and observational studies. Therefore, **PSweight** is directly applicable to perform covariate-adjusted analysis in RCTs.

The **PSweight** package is under continuing development to include other useful components for propensity score weighting analysis. Specifically, future versions of **PSweight** will include components to enable pre-specified subgroup analysis with balancing weights and flexible variable selection tools (Yang et al., 2021). We are also studying overlap weighting estimators with time-to-event outcomes and complex survey designs. Those new features are being actively developed concurrently with our extensions to the methodology.

6   Acknowledgement

Bibliography

A. Agresti. *Categorical Data Analysis*, volume 482. John Wiley & Sons, 2003. doi: 10.1080/02664763. 2013.854979. [p287]

P. C. Austin and E. A. Stuart. Moving towards best practice when using inverse probability of treatment weighting (iptw) using the propensity score to estimate causal treatment effects in observational studies. *Statistics in Medicine*, 34(28):3661–3679, 2015. doi: 10.1002/sim.6607. [p282, 285, 288]

H. Bang and J. M. Robins. Doubly robust estimation in missing data and causal inference models. *Biometrics*, 61(4):962–973, 2005. doi: 10.1111/j.1541-0420.2005.00377.x. [p286]

E. Battistin and B. Sianesi. Misclassified treatment status and treatment effects: An application to returns to education in the United Kingdom. *Review of Economics and Statistics*, 93(2):495–509, 2011. doi: 10.1162/REST_a_00175. [p289]

E. Battistin and B. Sianesi. Replication data for: Misclassified treatment status and treatment effects: An application to returns to education in the United Kingdom, 2012. URL https://doi.org/10.7910/DVN/EPCYUL. [p296]

H. Bodory and M. Huber. **causalweight**: *Causal Inference Based on Inverse Probability Weighting, Doubly Robust Estimation, and Double Machine Learning*, 2020. URL https://CRAN.R-project.org/package=causalweight. R package version 0.2.1. [p283]

S. v. Buuren and K. Groothuis-Oudshoorn. **mice**: Multivariate imputation by chained equations in R. *Journal of Statistical Software*, pages 1–68, 2010. doi: 10.18637/jss.v045.i03. [p289]

R. K. Crump, V. J. Hotz, G. W. Imbens, and O. A. Mitnik. Dealing with limited overlap in estimation of average treatment effects. *Biometrika*, 96(1):187–199, 2009. doi: 10.1093/biomet/asn055. [p282, 286, 291]

C. Fong, M. Ratkovic, and K. Imai. **CBPS**: *Covariate Balancing Propensity Score*, 2019. URL https://CRAN.R-project.org/package=CBPS. R package version 0.21. [p283]

Y. Freund and R. E. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1):119–139, 1997. doi: 10.1006/jcss.1997.1504. [p295]

J. Friedman, T. Hastie, and R. Tibshirani. Additive logistic regression: A statistical view of boosting. *The Annals of Statistics*, 28(2):337–407, 2000. doi: 10.1214/aos/1016218223. [p295]

M. J. Funk, D. Westreich, C. Wiesen, T. Stürmer, M. A. Brookhart, and M. Davidian. Doubly robust estimation of causal effects. *American Journal of Epidemiology*, 173(7):761–767, 2011. doi: 10.1093/aje/kwq439. [p286]

B. Greenwell, B. Boehmke, J. Cunningham, and G. Developers. **gbm**: *Generalized Boosted Regression Models*, 2019. URL https://CRAN.R-project.org/package=gbm. R package version 2.1.5. [p288]

N. Greifer. Covariate balance tables and plots: A guide to the cobalt package. Technical report, Johns Hopkins University, 2018. [p285]

N. Greifer. **optweight**: *Targeted Stable Balancing Weights Using Optimization*, 2019. URL https://CRAN.R-project.org/package=optweight. R package version 0.2.5. [p283]

N. Greifer. **WeightIt**: *Weighting for Covariate Balance in Observational Studies*, 2020. URL https://CRAN.R-project.org/package=WeightIt. R package version 0.10.2. [p283]

A. Haris and G. Chan. **ATE**: *Inference for Average Treatment Effects using Covariate Balancing*, 2015. URL https://CRAN.R-project.org/package=ATE. R package version 0.2.0. [p283]

J. L. Hill. Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics*, 20(1):217–240, 2011. doi: 10.1198/jcgs.2010.08162. [p295]

K. Hirano, G. Imbens, and G. Ridder. Efficient estimation of average treatment effects using the estimated propensity score. *Econometrica*, 71:1161–1189, 2003. doi: 10.3386/t0251. [p282, 286]

G. W. Imbens. The role of the propensity score in estimating dose-response functions. *Biometrika*, 87(3):706–710, 2000. doi: 10.1093/biomet/87.3.706. [p285]

M. Lechner. Identification and estimation of causal effects of multiple treatments under the conditional independence assumption. In *Econometric Evaluation of Labour Market Policies*, pages 43–58. Springer, 2001. doi: 10.2139/ssrn.177089. [p285]

B. K. Lee, J. Lessler, and E. A. Stuart. Improving propensity score weighting using machine learning. *Statistics in Medicine*, 29(3):337–346, 2010. doi: 10.1002/sim.3782. [p295]

F. Li and F. Li. Propensity score weighting for causal inference with multiple treatments. *The Annals of Applied Statistics*, 13(4):2389–2415, 2019. doi: 10.1214/19-AOAS1282. [p282, 284, 285, 286, 288, 296]

F. Li, K. L. Morgan, and A. M. Zaslavsky. Balancing covariates via propensity score weighting. *Journal of the American Statistical Association*, 113(521):390–400, 2018. doi: 10.1080/01621459.2016.1260466. [p282, 283, 284, 296]

F. Li, L. E. Thomas, and F. Li. Addressing extreme propensity scores via the overlap weights. *American Journal of Epidemiology*, 1(188):250–257, 2019. doi: 10.1093/aje/kwy201. [p282, 284, 287, 292]

L. Li and T. Greene. A weighting analogue to pair matching in propensity score analysis. *International Journal of Biostatistics*, 9(2):1–20, 2013. doi: 10.1515/ijb-2012-0030. [p282, 284]

J. K. Lunceford and M. Davidian. Stratification and weighting via the propensity score in estimation of causal treatment effects: A comparative study. *Statistics in Medicine*, 23(19):2937–2960, 2004. doi: 10.1002/sim.1903. [p282, 286, 287]

H. Mao and L. Li. **PSW**: *Propensity Score Weighting Methods for Dichotomous Treatments*, 2018. URL https://CRAN.R-project.org/package=PSW. R package version 1.1-3. [p283]

H. Mao, L. Li, and T. Greene. Propensity score weighting analysis and treatment effect discovery. *Statistical Methods in Medical Research*, page In press, 2018. doi: 10.1177/0962280218781171. [p282, 284, 286, 287]

D. F. McCaffrey, G. Ridgeway, and A. Morral. Propensity score estimation with boosted regression for evaluating causal effects in observational studies. *Psychological Methods*, 9(4):403–425, 2004. doi: 10.1037/1082-989X.9.4.403. [p295]

D. F. McCaffrey, B. A. Griffin, D. Almirall, M. E. Slaughter, R. Ramchand, and L. F. Burgette. A tutorial on propensity score estimation for multiple treatments using generalized boosted models. *Statistics in Medicine*, 32(19):3388–3414, 2013. doi: 10.1002/sim.5753. [p282, 295]

M. Nurminen. To use or not to use the odds ratio in epidemiologic analyses? *European Journal of Epidemiology*, 11(4):365–371, 1995. doi: 10.1007/BF01721219. [p294]

E. Polley, E. LeDell, C. Kennedy, and M. van der Laan. **SuperLearner**: *Super Learner Prediction*, 2019. URL https://CRAN.R-project.org/package=SuperLearner. R package version 2.0-26. [p288]

R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2016. URL https://www.R-project.org/. ISBN 3-900051-07-0. [p]

G. Ridgeway, D. McCaffrey, A. Morral, B. A. Griffin, L. Burgette, and M. Cefalu. **twang**: *Toolkit for Weighting and Analysis of Nonequivalent Groups*, 2020. URL https://CRAN.R-project.org/package=twang. R package version 1.6. [p282, 283]

J. M. Robins, A. Rotnitzky, and L. P. Zhao. Estimation of regression-coefficients when some regressors are not always observed. *Journal of the American Statistical Association*, 89(427):846–866, 1994. doi: 10.2307/2290910. [p282, 286]

P. R. Rosenbaum and D. B. Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 1983. doi: 10.1093/biomet/70.1.41. [p282, 283]

C. Shen, X. Li, and L. Li. Inverse probability weighting for covariate adjustment in randomized studies. *Statistics in Medicine*, 33(4):555–568, 2014. doi: 10.1186/s12874-020-00947-7. [p296]

L. A. Stefanski and D. D. Boos. The calculus of m-estimation. *American Statistician*, 56(1):29–38, 2002. doi: 10.1198/000313002753631330. [p287]

L. E. Thomas, F. Li, and M. J. Pencina. Overlap weighting: A propensity score method that mimics attributes of a randomized clinical trial. *Journal of the American Medical Association*, 323(23):2417–2418, 2020a. doi: 10.1001/jama.2020.7819. [p282]

L. E. Thomas, F. Li, and M. J. Pencina. Using propensity score methods to create target populations in observational clinical research. *Journal of the American Medical Association*, 323(5):466–467, 2020b. doi: 10.1001/jama.2019.21558. [p282, 283]

A. A. Tsiatis, M. Davidian, M. Zhang, and X. Lu. Covariate adjustment for two-sample treatment comparisons in randomized clinical trials: A principled yet flexible approach. *Statistics in Medicine*, 27(23):4658–4677, 2008. doi: 10.1002/sim.3113. [p296]

M. J. Van der Laan, E. C. Polley, and A. E. Hubbard. Super Learner. *Statistical Applications in Genetics and Molecular Biology*, 6(1), 2007. doi: 10.2202/1544-6115.1309. [p282]

E. J. Williamson, A. Forbes, and I. R. White. Variance reduction in randomised trials by inverse probability weighting using the propensity score. *Statistics in Medicine*, 33(5):721–737, 2014. doi: 10.1002/sim.5991. [p296]

S. Yang, G. W. Imbens, Z. Cui, D. E. Faries, and Z. Kadziola. Propensity score matching and subclassification in observational studies with multi-level treatments. *Biometrics*, 72(4):1055–1065, 2016. doi: 10.1111/biom.12505. [p282, 286, 291]

S. Yang, E. Lorenzi, G. Papadogeorgou, D. M. Wojdyla, F. Li, and L. E. Thomas. Propensity score weighting for causal subgroup analysis. *Statistics in Medicine*, 2021. [p296]

K. Yoshida, S. Hernández-Díaz, D. H. Solomon, J. W. Jackson, J. J. Gagne, R. J. Glynn, and J. M. Franklin. Matching weights to simultaneously compare three treatment groups comparison to three-way matching. *Epidemiology*, 28(3):387–395, 2017. doi: 10.1097/EDE.0000000000000627. [p282, 284]

K. Yoshida, D. Solomon, S. Haneuse, S. Kim, E. Patorno, S. Tedeschi, H. Lyu, T. Franklin, J.M.and Stürmer, S. Hernández-Díaz, and R. Glynn. Multinomial extension of propensity Score trimming methods: A simulation study. *American Journal of Epidemiology*, 183(3):609–616, 2018. doi: 10.1093/aje/kwy263. [p282, 284, 286, 291, 292]

S. Zeng, F. Li, R. Wang, and F. Li. Propensity Score weighting for covariate adjustment in randomized clinical trials. *Statistics in Medicine*, 40(4):842–858, 2020. [p296]

Y. Zhou, R. A. Matsouaka, and L. Thomas. Propensity core weighting under limited overlap and model misspecification. *Statistical Methods in Medical Research*, 29(12):3721–3756, 2020. doi: 10.1177/0962280220940334. [p282, 284]

J. R. Zubizarreta and Y. Li. **sbw**: *Stable Balancing Weights for Causal Inference and Estimation with Incomplete Outcome Data*, 2020. URL https://CRAN.R-project.org/package=sbw. R package version 1.1.1. [p283]

Tianhui Zhou
Department of Biostatistics and Bioinformatics
Duke University School of Medicine
2424 Erwin Road, Suite 1105
Durham, NC 27705, United States of America
E-mail: tianhui.zhou@duke.edu

Guangyu Tong
Department of Biostatistics
Yale University School of Public Health
300 George St, Suite 511
New Haven, CT 06510, United States of America
E-mail: guangyu.tong@yale.edu
⋆*T. Zhou and G. Tong contributed equally to this work and are considered co-first authors.*

Fan Li
Department of Statistical Science
Duke University
122 Old Chemistry Building
Durham, NC 27705, United States of America
E-mail: fl35@duke.edu

Laine E. Thomas
Department of Biostatistics and Bioinformatics
Duke University School of Medicine
2424 Erwin Road, Suite 1105
Durham, NC 27705, United States of America
E-mail: laine.thomas@duke.edu

Fan Li
Department of Biostatistics
Yale University School of Public Health
135 College St, Suite 200
New Haven, CT 06510, United States of America
E-mail: fan.f.li@yale.edu