CIMTx: An R Package for Causal Inference with Multiple Treatments using Observational Data

by Lianyuan Hu and Jiayi Ji

Abstract **CIMTx** provides efficient and unified functions to implement modern methods for causal inferences with multiple treatments using observational data with a focus on binary outcomes. The methods include regression adjustment, inverse probability of treatment weighting, Bayesian additive regression trees, regression adjustment with multivariate spline of the generalized propensity score, vector matching and targeted maximum likelihood estimation. In addition, **CIMTx** illustrates ways in which users can simulate data adhering to the complex data structures in the multiple treatment setting. Furthermore, the **CIMTx** package offers a unique set of features to address the key causal assumptions: positivity and ignorability. For the positivity assumption, **CIMTx** demonstrates techniques to identify the common support region for retaining inferential units using inverse probability of treatment weighting, Bayesian additive regression trees and vector matching. To handle the ignorability assumption, **CIMTx** provides a flexible Monte Carlo sensitivity analysis approach to evaluate how causal conclusions would be altered in response to different magnitude of departure from ignorable treatment assignment.

1 Introduction

Modern comparative effectiveness research (CER) questions often require comparing the effectiveness of multiple treatments on a binary outcome (Hu et al., 2020a). To answer these CER questions, specialized causal inference methods are needed. Methods appropriate for drawing causal inferences about multiple treatments include regression adjustment (RA) (Rubin, 1973; Linden et al., 2016), inverse probability of treatment weighting (IPTW) (Feng et al., 2012; McCaffrey et al., 2013), Bayesian Additive Regression Trees (BART) (Hill, 2011; Hu et al., 2021b, 2020a), regression adjustment with multivariate spline of the generalized propensity score (RAMS) (Hu and Gu, 2021), vector matching (VM) (Lopez and Gutman, 2017) and targeted maximum likelihood estimation (TMLE) (Rose and Normand, 2019). Drawing causal inferences using observational data, however, inevitably requires assumptions. A key causal identification assumption is the *positivity* or sufficient overlap assumption, which implies that there are no values of pre-treatment covariates that could occur only among units receiving one of the treatments (Hu et al., 2020a). Another key assumption requires appropriately conditioning on all pre-treatment variables that predict both treatment and outcome. The pre-treatment variables are known as confounders and this requirement is referred to as the *ignorability* assumption (also as no unmeasured confounding) (Hu et al., 2022b). An important strategy to handle the positivity assumption is to identify a common support region for retaining inferential units. The ignorability assumption can be violated in observational studies, and as a result can lead to biased treatment effect estimates. One widely recognized way to address such concerns is sensitivity analysis (Erik von Elm et al., 2007; Hu et al., 2022b).

The **CIMTx** package provides a suite of functions to easily implement the causal estimation methods, many of which were recently developed (Lopez and Gutman, 2017; Hu et al., 2020a; Hu and Gu, 2021). In addition, **CIMTx** provides strategies to define a common support region to address the positivity assumption using IPTW, BART, VM and implements a flexible Monte Carlo sensitivity analysis approach (Hu et al., 2022b) for unmeasured confounding to address the ignorability assumption. Finally, **CIMTx** offers detailed examples of how to simulate data adhering to the complex structures in the multiple treatment setting. The simulated data can then be used by an analyst to compare the performance of different causal estimation methods. Table 1 summarizes key functionalities of **CIMTx** in comparison to recent R packages designed for causal inference with multiple treatments using observational data. **CIMTx** provides a comprehensive set of functionalities: from simulating data to estimating the causal effects to addressing causal assumptions and elucidating their ramifications. To assist applied researchers and practitioners who work with observational data and wish to draw inferences about the effects of multiple treatments, this article provides a comprehensive illustration of the **CIMTx** package.

Table 1: Comparisons of R packages for causal inference.

R packages	Continuous Outcome	Binary Outcome	Sensitivity Analysis	Identification of Common Support	Design factors	Estimation procedure
CIMTx	✗	✔	✔	✔*	✔	RA, IPTW-SL IPTW-Multinomial IPTW-GBM VM, BART RAMS, TMLE
PSweight	✔	✔	✗	✔	✗	OW, IPTW-SL IPTW-Multinomial IPTW-GBM
twang	✔	✗	✔	✗	✗	IPTW-GBM
WeightIt	✔	✗	✗	✔	✗	CBPS, IPTW-SL IPTW-Multinomial IPTW-GBM,EBCW IPTW-TSBW
CBPS	✔	✔	✗	✗	✗	CBPS
optweight	✔	✗	✗	✗	✗	IPTW-TSBW

✔: the feature is offered in the method; ✗ indicates otherwise; RA: Regression adjustment; IPTW: Inverse probability of treatment weighting; BART: Bayesian additive regression trees; RAMS: Regression adjustment with multivariate spline of generalized propensity score; VM: Vector matching; TMLE: Targeted maximum likelihood estimation; CBPS: Covariate balancing propensity score; OW: Overlap weights; IPTW-Multinomial: Inverse probability of treatment weighting with weight estimated by multinomial logistic regression; IPTW-GBM: Inverse probability of treatment weighting with weight estimated by generalized boosted model; IPTW-SL: Inverse probability of treatment weighting with weight estimated by super learner; IPTW-TSBW: Inverse probability of treatment weighting with targeted stable balancing weights; EBCW: Empirical balancing calibration weights.
*: Identification of Common Support is only for VM, BART and IPTW related methods
References: **PSweight** (Version 1.1.4): Zhou et al. (2020);**twang** (Version 1.6) Ridgeway et al. (2020);**WeightIt** (Version 0.10.2) Greifer (2020);**CBPS** (Version 0.22): Fong et al. (2021); **optweight** (Version 0.2.5): Greifer (2019);

2 Design factors for data simulation

CIMTx provides specific functions to simulate data possessing complex data characteristics of the multiple treatment setting. Seven design factors are considered: (1) sample size, (2) ratio of units across treatment groups, (3) whether the treatment assignment model and the outcome generating model are linear or nonlinear, (4) whether the covariates that best predict the treatment also predict the outcome well, (5) whether the response surfaces are parallel across treatment groups, (6) outcome prevalence, and (7) degree of covariate overlap.

Design factors (1)–(5)

For the data generating process of treatment assignment, consider a multinomial logistic regression model,

$$\ln \frac{P(W=1)}{P(W=T)} = \delta_1 + X\xi_1^L + Q\xi_1^{NL}$$

$$\vdots \tag{1}$$

$$\ln \frac{P(W=T-1)}{P(W=T)} = \delta_{(T-1)} + X\xi_{(T-1)}^L + Q\xi_{(T-1)}^{NL},$$

where Q denotes the nonlinear transformations and higher-order terms of the predictors X. $\xi_1^L, \ldots, \xi_{(T-1)}^L$ are vectors of coefficients for the untransformed versions of the predictors X and $\xi_1^{NL}, \ldots, \xi_{(T-1)}^{NL}$ for the transformed versions of the predictors captured in Q. The intercepts $\delta_1, \ldots, \delta_{(T-1)}$ can be specified to create the corresponding ratio of units across T treatment groups. The T sets of potential response

surfaces can be generated as follows:

$$E[Y(1)|\boldsymbol{X}] = \text{logit}^{-1}\{\tau_1 + \boldsymbol{X}\gamma_1^L + \boldsymbol{Q}\gamma_1^{NL}\}$$
$$\vdots \qquad\qquad (2)$$
$$E[Y(T)|\boldsymbol{X}] = \text{logit}^{-1}\{\tau_T + \boldsymbol{X}\gamma_T^L + \boldsymbol{Q}\gamma_T^{NL}\},$$

where the coefficient setting $\gamma_1^L = \ldots = \gamma_T^L$, $\gamma_1^{NL} = \ldots = \gamma_T^{NL}$ and $\tau_1 \neq \ldots \neq \tau_T$ corresponds to the parallel response surfaces, and by assigning different values to γ_w^L and γ_w^{NL} and setting $\tau_1 = \ldots = \tau_T = 0$, nonparallel response surfaces are generated, which imply treatment effect heterogeneity. Note that the predictors $\boldsymbol{X}$ and the transformed versions of the predictors $\boldsymbol{Q}$ in the treatment assignment model (1) can be different than those in the outcome generating model (2) to create various degrees of alignment. The observed outcomes are related to the potential outcomes through $Y_i = \sum_{w_i \in \mathscr{W}} Y_i(w)$. Covariates $\boldsymbol{X}$ can be generated from user-specified data distributions.

Outcome prevalence

Values for parameters $\tau_1, \ldots, \tau_T$ in model (2) can be chosen to create various outcome prevalence rates. The outcomes are considered rare if the prevalence rate is $< 5\%$.

Covariate overlap

With observational data, it is important to investigate how the sparsity of covariate overlap impacts the estimation of causal effects. We can modify the formulation of the treatment assignment model (1) to adjust the sparsity of overlap by including a multiplier parameter ψ (Hu et al., 2021a) as follows:

$$\ln \frac{P(W=1)}{P(W=T)} = \delta_1 + \boldsymbol{X}\psi\xi_1^L + \boldsymbol{Q}\psi\xi_1^{NL}$$
$$\vdots \qquad\qquad (3)$$
$$\ln \frac{P(W=T-1)}{P(W=T)} = \delta_{(T-1)} + \boldsymbol{X}\psi\xi_{(T-1)}^{NL} + \boldsymbol{Q}\psi\xi_{(T-1)}^{NL},$$

where larger values of ψ correspond to increased sparsity degrees of overlap.

Implementation in CIMTx

We will first demonstrate the functionality of `data_sim()` in **CIMTx** to simulate data in the multiple treatment setting using the above 7 design factors. We first use the `data_sim()` function to simulate a dataset with the following characteristics: (1) sample size = 500, (2) ratio of units = 1:1:1 across three treatment groups, (3) nonlinear treatment assignment and outcome generating models, (4) different predictors for the treatment assignment and outcome generating mechanisms, (5) parallel response surfaces, (6) outcome prevalence = $(0.16, 0.51, 0.75)$ in three treatment groups with an overall rate around 0.5 and (7) moderate covariate overlap. Note that for the design factor (6), we can adjust `tau` to generate rare outcome events.

The outputs of the simulated data object are: (1) `data$covariates` for $\boldsymbol{X}$, (2) `data$w` for treatment indicators, (3) `data$y` for observed binary outcomes, (4) `data$y_prev` for the outcome prevalence rates, (5) `data$ratio_of_units` for the proportions of units in each treatment group, (6) `data$overlap_fig` for the visualization of covariate overlap via boxplots of the distributions of true generalized propensity score (GPS).

```
library(CIMTx)
set.seed(1)
data <- data_sim(
  sample_size = 500, n_trt = 3,
  x = c("rnorm(0, 0.5)",  # x1
        "rbeta(2, .4)",   # x2
        "runif(0, 0.5)",  # x3
        "rweibull(1, 2)", # x4
        "rbinom(1, .4)"),  # x5
  # linear terms in parallel response surfaces
  lp_y = rep(".2*x1 + .3*x2 - .1*x3 - .1*x4 - .2*x5", 3),
```

```
# nonlinear terms in parallel response surfaces
nlp_y = rep(".7*x1*x1  - .1*x2*x3", 3),
align = F,# different predictors used in treatment and outcome models
# linear terms in treatment assignment model
lp_w = c(".4*x1 + .1*x2  - .1*x4 + .1*x5",   # w = 1
         ".2*x1 + .2*x2  - .2*x4 - .3*x5"),  # w = 2
# nonlinear terms in treatment assignment model
nlp_w = c("-.5*x1*x4  - .1*x2*x5", # w = 1
          "-.3*x1*x4 + .2*x2*x5"), # w = 2
tau = c(-1.5, 0, 1.5), delta = c(0.5, 0.5), psi = 1)
```

In this simulated dataset, the ratio of units (data$ratio_of_units) and outcome prevalences (data$y_prev) are:

```
#> w
#>    1    2    3
#> 0.35 0.35 0.30

#>        w y_prev
#> 1      1   0.16
#> 2      2   0.51
#> 3      3   0.75
#> 4 Overall  0.46
```
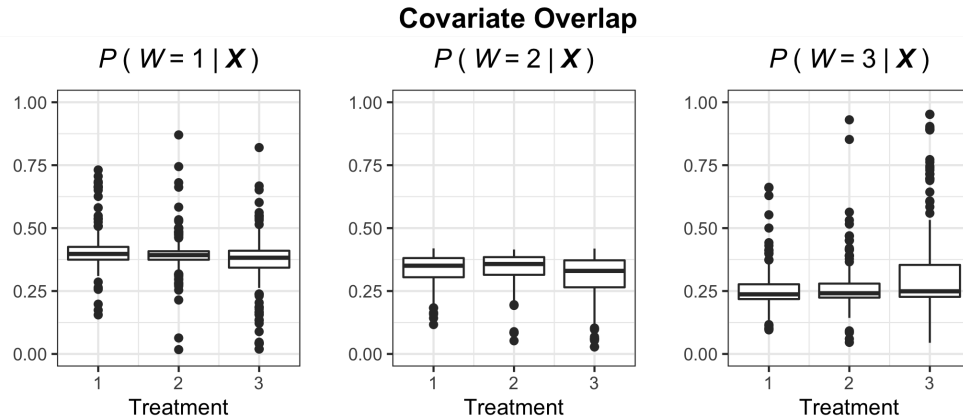
Covariate Overlap

Figure 1: Moderate overlap with psi = 1. Each panel presents boxplots by treatment group of the true generalized propensity score for one of the treatments, $P(W_i = w \mid X = x)$ for every unit in the sample. The left-hand panel presents treatment 1 ($W = 1$), the middle panel presents treatment 2 ($W = 2$), and the right-hand panel presents treatment 3 ($W = 3$).

Figure 1 (data$overlap_fig) shows the distributions of true GPS for each treatment group, suggesting moderate covariate overlap. We can change structures of the simulated data by modifying arguments of the data_sim() function. For example, setting delta = c(1.5,0.5) yields unequal sample sizes across treatment groups with the ratio of unit .6 : .2 : .2. Assigning smaller values to psi can increase overlap: psi = 0.1 corresponds to a strong covariate overlap as shown in Figure 2.
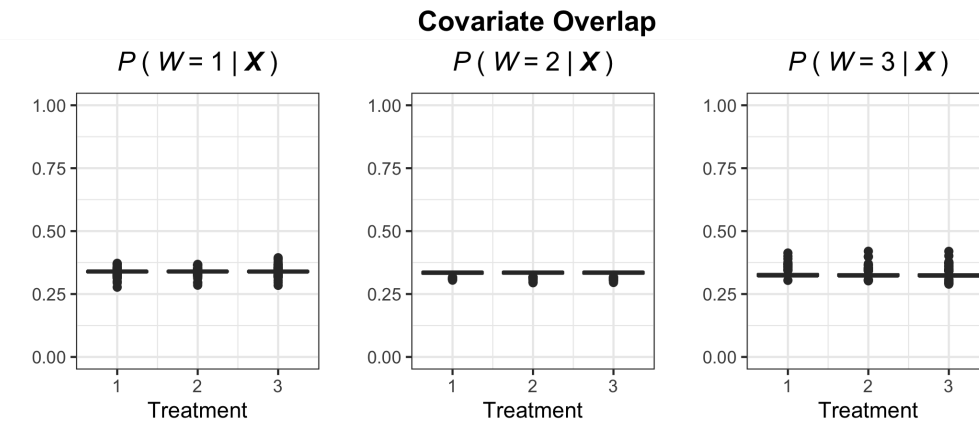
Figure 2: Strong overlap with `psi = 0.1`. Each panel presents boxplots by treatment group of the true generalized propensity score for one of the treatments for every unit in the sample.

3 Methodology and implementation in CIMTx

Estimation of causal effects

Consider an observational study with N individuals, indexed by $i = 1, \ldots, N$, drawn randomly from a target population. Each individual was exposed to one and only one treatment, indexed by W. The goal of this study is to estimate the causal effect of treatment W on a binary outcome Y. There are a total of T possible treatments, and $W_i = w$ if individual i is observed under treatment w, where $w \in \mathcal{W} = \{1, 2, \ldots, T\}$. Pre-treatment measured confounders are indexed by X_i. Under the potential outcomes framework, (Rubin, 1974; Holland, 1986), individual i has T potential outcomes $\{Y_i(1), \ldots, Y_i(T)\}$ under each treatment of $\mathcal{W}$. For each individual, at most one of the potential outcomes is observed – the one corresponding to the treatment to which the individual is exposed. All other potential outcomes are missing, which is known as the fundamental problem of causal inference (Holland, 1986). In general, three standard causal identification assumptions (Rubin, 1980; Hu et al., 2020a) need to be maintained in order to estimate the causal effects from observational data:

(A1) The stable unit treatment value assumption: there is no interference between units and there are no different versions of a treatment.

(A2) Positivity: the GPS for treatment assignment $e(X_i) = P(W_i = 1 \mid X_i)$ is bounded away from 0 and 1.

(A3) Ignorability: pre-treatment covariates X_i are sufficiently predictive of both treatment assignment and outcome, $p(W_i \mid Y_i(1), \ldots, Y_i(T), X_i) = p(W_i \mid X_i)$.

The **CIMTx** package addresses assumption (A2) in the section of "Identification of a common support region" and (A3) in the section of "Sensitivity analysis for unmeasured confounding".

Causal effects can be estimated by summarizing functionals of individual-level potential outcomes. For dichotomous outcomes, causal estimands can be the risk difference (RD), odds ratio (OR) or relative risk (RR). For purposes of illustration, we define causal effects based on the RD. Let s_1 and s_2 be two subgroups of treatments such that $s_1, s_2 \subset \mathcal{W}$ and $s_1 \cap s_2 = \emptyset$, and define $|s_1|$ as the cardinality of s_1 and $|s_2|$ of s_2. Two commonly used causal estimands are the average treatment effect (ATE), ATE_{s_1, s_2}, and the average treatment effect on the treated (ATT), for example, among those receiving s_1, $ATT_{s_1|s_1, s_2}$. They are defined as:

$$
\begin{aligned}
ATE_{s_1, s_2} &= E\left[\frac{\sum_{w \in s_1} Y_i(w)}{|s_1|} - \frac{\sum_{w' \in s_2} Y_i(w')}{|s_2|} \right], \\
ATT_{s_1|s_1, s_2} &= E\left[\frac{\sum_{w \in s_1} Y_i(w)}{|s_1|} - \frac{\sum_{w' \in s_2} Y_i(w')}{|s_2|} \middle| W_i \in s_1 \right].
\end{aligned}
\tag{4}
$$

We now introduce six methods implemented in **CIMTx** for estimating the causal effects of multiple treatments: RA, IPTW, BART, RAMS, VM and TMLE.

Regression adjustment Regression adjustment (Rubin, 1973; Linden et al., 2016), also known as model-based imputation (Imbens and Rubin, 2015), uses a regression model to impute missing

potential outcomes: what would have happened to a specific individual had this individual received a treatment to which he or she not exposed. RA regresses the outcomes on treatment and confounders,

$$f(w, \mathbf{X}_i) = E[Y_i \mid W_i = w, \mathbf{X}_i] = \text{logit}^{-1}\left\{\beta_0 + \beta_1 w + \boldsymbol{\beta}_2^\top \mathbf{X}_i\right\}, \tag{5}$$

where β_0 is the intercept, β_1 is the coefficient for treatment and $\boldsymbol{\beta}_2$ is a vector of coefficients for covariates $\mathbf{X}_i$. From the fitted regression model (5), the missing potential outcomes for each individual are imputed using the observed data. The causal effects can be estimated by contrasting the imputed potential outcomes between treatment groups. **CIMTx** implements RA with the Bayesian logistic regression model via the bayesglm() function of the **arm** package. For the ATE effects, we first average the L predictive posterior draws $\{f^l(w, \mathbf{X}_i), l = 1, \ldots, L\}$ over the empirical distribution of $\{\mathbf{X}_i\}_{i=1}^{N}$, and for the ATT effects using s_1 as the reference group, over the empirical distribution of $\{\mathbf{X}_i\}_{i:W_i \in s_1}$. We then take the difference of the averaged values between two treatment groups $w \in s_1$ and $w' \in s_2$. Inferences about treatment effect can be obtained based on the L posterior average treatment effects. The 95% credible interval is calculated using the 2.5th percentile and the 97.5th percentile of the posterior draws (Kruschke, 2014).

In our package **CIMTx**, we can specify method = "RA" and estimand = "ATE" in the ce_estimate() function to get the ATE effects via RA:

```
ra_ate_res <- ce_estimate(y = data$y, x = data$covariates, w = data$w,
                          method = "RA", estimand = "ATE", ndpost = 100)
```

The estimates, standard errors and 95% confidence intervals for the causal estimands would be printed using the summary() generic function:

```
summary(ra_ate_res)

#> $ATE12
#>       EST   SE LOWER UPPER
#> RD -0.28 0.04 -0.35 -0.21
#> RR  0.39 0.07  0.27  0.54
#> OR  0.26 0.06  0.17  0.40


#> $ATE13
#>       EST   SE LOWER UPPER
#> RD -0.60 0.06 -0.69 -0.47
#> RR  0.23 0.05  0.15  0.34
#> OR  0.07 0.02  0.03  0.13


#> $ATE23
#>       EST   SE LOWER UPPER
#> RD -0.31 0.04 -0.39 -0.25
#> RR  0.60 0.04  0.52  0.67
#> OR  0.25 0.05  0.17  0.34
```

Specifying estimand = "ATT" and setting reference_trt will get us the ATT effects:

```
ra_att_res <- ce_estimate(y = data$y, x = data$covariates,w = data$w, method = "RA",
                          estimand = "ATT", ndpost = 100, reference_trt = 1)
summary(ra_att_res)

#> $ATT12
#>       EST   SE LOWER UPPER
#> RD -0.28 0.05 -0.37 -0.18
#> RR  0.40 0.09  0.25  0.57
#> OR  0.27 0.08  0.16  0.44


#> $ATT13
#>       EST   SE LOWER UPPER
#> RD -0.59 0.06 -0.67 -0.46
#> RR  0.24 0.06  0.14  0.38
#> OR  0.07 0.03  0.03  0.13
```

Inverse probability of treatment weighting The idea of IPTW was originally introduced by Horvitz and Thompson (1952) in survey research to adjust for imbalances in sampling pools. Weighting

methods have been extended to estimate the causal effect of a binary treatment in observational studies, and more recently reformulated to accommodate multiple treatments (Imbens, 2000; Feng et al., 2012; McCaffrey et al., 2013). When interest is in estimating the pairwise ATE for treatment groups s_1 and s_2, a consistent estimator of ATE_{s_1,s_2} is given by the weighted mean,

$$\widehat{ATE}_{s_1,s_2} = \frac{\sum_{i=1}^{N} Y_i I(W_i \in s_1)/|s_1|}{\sum_{i=1}^{N} I(W_i \in s_1) r(W_i, \boldsymbol{X_i})} - \frac{\sum_{i=1}^{N} Y_i I(W_i \in s_2)/|s_2|}{\sum_{i=1}^{N} I(W_i \in s_2) r(W_i, \boldsymbol{X_i})}, \tag{6}$$

where $r(w, \boldsymbol{X}_i)$ is the weights satisfying $r(w, \boldsymbol{X}_i) = 1/P(W_i = w \mid \boldsymbol{X}_i)$, and $I(\cdot)$ is the indicator function. The **CIMTx** package provides three ways in which the weights can be estimated: (i) multinomial logistic regression (Feng et al., 2012), (ii) generalized boosted model (GBM) (McCaffrey et al., 2013), and (iii) super learner (Van der Laan et al., 2007). A challenge with IPTW is low GPS can result in extreme weights, which may yield erratic causal estimates with large sample variances (Little, 1988; Kang et al., 2007). This issue is increasingly likely as the number of treatments increases. Weight trimming or truncation can alleviate the issue of extreme weights (Cole and Hernán, 2008; Lee et al., 2011)). **CIMTx** provides an argument for users to choose the percentile at which the weights should be truncated. We briefly describe the three weight estimators.

(i) The multinomial logistic regression model for treatment assignment is as follows:

$$P(W_i = w|\boldsymbol{X}_i) = \frac{e^{\boldsymbol{\alpha}'_w \boldsymbol{X}_i}}{1 + e^{\boldsymbol{\alpha}'_1 \boldsymbol{X}_i} + \ldots + e^{\boldsymbol{\alpha}'_{T-1} \boldsymbol{X}_i}},$$

where $\boldsymbol{\alpha}'_w$ is a vector of coefficients for $\boldsymbol{X}_i$ corresponding to treatment w, and can be estimated by using an iterative procedure such as generalized iterative scaling or iteratively reweighted least squares.

(ii) GBM uses machine learning to flexibly model the relationships between treatment assignment and covariates. It does this by growing a series of boosted classification trees to minimize an exponential loss function. This process is effective for fitting nonlinear treatment models characterized by curves and interactions. The procedure of estimating the GPS can be tuned to find the GPS model producing the best covariate balance between treatment groups.

(iii) Super learner is an algorithm that creates the optimally weighted average of several machine learning models. The machine learning models can be specified via the SL.library argument of the **SuperLearner** package. This approach has been proven to be asymptotically as accurate as the best possible prediction algorithm that is included in the library (Van der Laan et al., 2007).

IPTW can be implemented in **CIMTx** by setting a specific method and estimand. For IPTW estimators, variance can be estimated via a robust sandwich-type variance estimator or a bootstrap variance estimator. In practice, a bootstrap variance estimator is often recommended. (Austin, 2016). The following shows the code to estimate ATE using IPTW with weights estimated by multinomial logistic regression.

```
iptw_multi_res <- ce_estimate(y = data$y, x = data$covariates , w = data$w,
                              method = "IPTW-Multinomial", estimand = "ATE")
```

We can estimate the ATE effects with weights estimated by super learner and GBM by changing the argument of method to "IPTW-SL","IPTW-GBM" respectively. We can then estimate the causal effects and bootstrap confidence intervals by setting boot = TRUE.

```
iptw_sl_trim_ate_res <- ce_estimate(y = data$y, x = data$covariates , w = data$w,
                                    method = "IPTW-SL", estimand = "ATE",
                                    sl_library =  c("SL.glm", "SL.glmnet", "SL.rpart"),
                                    trim_perc = c(0.05,0.95), boot = TRUE,
                                    nboots = 100, verbose_boot = F)
```

```
summary(iptw_sl_trim_ate_res)
#> $ATE12
#>      EST   SE LOWER UPPER
#> RD -0.34 0.05 -0.42 -0.24
#> RR  0.33 0.07  0.19  0.48
#> OR  0.20 0.06  0.10  0.33

#> $ATE13
#>      EST   SE LOWER UPPER
#> RD -0.59 0.05 -0.67 -0.46
```

```
#> RR  0.22 0.05  0.13  0.34
#> OR  0.07 0.02  0.04  0.13


#> $ATE23
#>       EST   SE LOWER UPPER
#> RD -0.25 0.05 -0.34 -0.15
#> RR  0.67 0.06  0.57  0.79
#> OR  0.34 0.09  0.21  0.54
```

Bayesian additive regression trees BART (Chipman et al., 2010) is a likelihood-based machine learning model and has been adapted into causal inference settings in recent years (Hill, 2011; Hu et al., 2020a; Hu and Gu, 2021; Hu et al., 2021a,c). For a binary outcome, BART uses the probit regression

$$f(w, \boldsymbol{X}_i) = E[Y_i | W_i = w, \boldsymbol{X}_i] = \Phi\left\{ \sum_{j=1}^{J} g_j(w, \boldsymbol{X}_i; T_j, M_j) \right\}, \tag{7}$$

where Φ is the the standard normal cumulative distribution function, (T_j, M_j) indexes a single subtree model in which T_j denotes the regression tree and M_j is a set of parameter values associated with the terminal nodes of the jth regression tree, $g_j(w, \boldsymbol{X}_i; T_j, M_j)$ represents the mean assigned to the node in the jth regression tree associated with covariate value $\boldsymbol{X}_i$ and treatment level w, and the number of regression trees J is considered to be fixed and known. BART uses regularizing priors for (T_j, M_j) to keep the impact of each tree small. Although the prior distributions can be specified via the `ce_estimate()` function of **CIMTx**, the default priors tend to work well and require little modification in many situations (Hill, 2011; Hu et al., 2020a,b). The details of prior specification and Bayesian backfitting algorithm for posterior sampling can be found in Chipman et al. (2010). The posterior inferences about the treatment effects can be drawn in a similar way as described in the Regression adjustment section.

Setting `method = "BART"` and specifiying the `estimand = "ATE"` or `estimand = "ATT"` of the `ce_estimate()` function implements the BART method.

```
bart_res <- ce_estimate(y = data$y, x = data$covariates, w = data$w, method = "BART",
                        estimand = "ATT", ndpost=100, reference_trt = 1)

summary(bart_res)

#> $ATT12
#>       EST   SE LOWER UPPER
#> RD -0.38 0.07 -0.51 -0.25
#> RR  0.47 0.08  0.31  0.61
#> OR  0.21 0.07  0.10  0.35


#> $ATT13
#>       EST   SE LOWER UPPER
#> RD -0.56 0.07 -0.69 -0.43
#> RR  0.38 0.07  0.24  0.50
#> OR  0.06 0.03  0.02  0.13
```

Regression adjustment with multivariate spline of GPS For a binary outcome, the number of outcome events can be small. The estimation of causal effects is challenging with rare outcomes because the great majority of units contribute no information to explaining the variability attributable to the differential treatment regimens in the health outcomes (Hu and Gu, 2021). Franklin et al. (2017) found that regression adjustment on propensity score using one nonlinear spline performed best with respect to bias and root-mean-squared-error in estimating treatment effects. Hu and Gu (2021) proposed RAMS, which accommodates multiple treatments by using a nonlinear spline model for the outcome that is additive in the treatment and multivariate spline function of the GPS as the following:

$$f(W_i, \boldsymbol{X}_i) = E[Y_i | W_i, \boldsymbol{X}_i] = \text{logit}^{-1}\left\{ \boldsymbol{\beta} W_i + h(\boldsymbol{R}(\boldsymbol{X}_i), \boldsymbol{\phi}) \right\}, \tag{8}$$

where $h(\cdot)$ is a spline function of the GPS indexed by $\boldsymbol{\phi}$ and $\boldsymbol{\beta} = [\beta_1, \dots, \beta_T]^\top$ are regression coefficients associated with the treatment W_i. The dimension of the spline function $h(\cdot)$ depends on the number of treatments T. Confidence intervals of treatment effect estimates can be obtained using nonparametric bootstrap for RAMS (Hu and Gu, 2021).

In **CIMTx**, RAMS is implemented using the gam() function with tensor product smoother te() between treatments from the **mgcv** package. Treatment effects can then be estimated by averaging and contrasting the predicted $\hat{f}(w, X_i)$ between treatment groups. The RAMS can be called by setting method = "RAMS-Multinomial" and specifying the estimand estimand = "ATE" or estimand = "ATT".

```
rams_multi_res <- ce_estimate(y = data$y, x = data$covariates, w = data$w,
                              method = "RAMS-Multinomial", estimand = "ATE",
                              boot = TRUE, nboots = 100, verbose_boot = F)
```

Vector matching Lopez and Gutman (2017) proposed the VM algorithm, which matches individuals with similar vector of the GPS. VM obtains matched sets using a combination of k-means clustering and one-to-one matching with replacement within each cluster strata. Currently, VM is only designed to estimate the ATT effects. In **CIMTx** , VM is implemented via method = "VM". The **CIMTx** does not provide confidence intervals for treatment effect estimates because the authors of this method, Lopez and Gutman (2017), did not provide an approach to estimate the sampling variance of the VM estimator.

To implement VM in **CIMTx**, we set the reference group reference_trt = 1, the number of clusters to form using k-means clustering n_cluster = 3.

```
vm_res <- ce_estimate(y = data$y, x = data$covariates, w = data$w, method = "VM",
                      estimand = "ATT", reference_trt = 1, n_cluster = 3)
```

The number of matched individuals is also stored in the output list:

```
vm_res$number_matched
```

```
#> 158
```

Targeted maximum likelihood estimation TMLE is a doubly robust approach that combines outcome estimation, IPTW estimation, and a targeting step to optimize the parameter of interest with respect to bias and variance. Rose and Normand (2019) implemented TMLE to estimate the ATE effects of multiple treatments. **CIMTx** calls the R package **tmle** to implement TMLE for the ATE effects. As suggested by Rose and Normand (2019), nonparametric bootstrap is used in **CIMTx** to obtain the confidence interval of the treatment effect estimate.

Calling method = "TMLE" implements TMLE in **CIMTx**. We use nonparametric bootstrap to estimate the 95% confidence intervals by setting boot = TRUE and nboots = 100.

```
tmle_res_boot <- ce_estimate(y = data$y, x = data$covariates, w = data$w, nboots = 100,
                             method = "TMLE", estimand = "ATE", boot = TRUE,
                             sl_library = c("SL.glm", "SL.glmnet", "SL.rpart"))
```

```
summary(tmle_res)
```

```
#> $ATE12
#> EST    SE LOWER UPPER
#> RD -0.36 0.04 -0.45 -0.29
#> RR  0.30 0.05  0.21  0.39
#> OR  0.17 0.04  0.11  0.24

#> $ATE13
#> EST    SE LOWER UPPER
#> RD -0.60 0.04 -0.67 -0.51
#> RR  0.20 0.03  0.15  0.28
#> OR  0.06 0.02  0.04  0.10

#> $ATE23
#> EST    SE LOWER UPPER
#> RD -0.24 0.05 -0.34 -0.14
#> RR  0.68 0.06  0.57  0.79
#> OR  0.34 0.09  0.21  0.55
```

Identification of a common support region

Turning to causal identification assumptions. If the positivity assumption (A2) is violated, problems can arise when extrapolating over the areas of the covariate space where common support does not exist. It is important to define a common support region to which the causal conclusions can be generalized. In **CIMTx**, the identification of a common support region is offered in three methods: IPTW, VM and BART.

For IPTW, one strategy is weight truncation, by which extreme weights that fall outside a specified range limit of the weight distribution are set to the range limit. This functionality is offered in **CIMTx** via the `trim_perc` argument. `trim_perc`, which can take two values – one for the lower- and one for the upper-percentile of the weight distribution for trimming. Figure 3 shows the distributions of the weights estimated by the three methods before and after weight trimming at the 5% and 95% of the weight distribution.

```
plot(iptw_multi_res, iptw_sl_res, iptw_gbm_res, iptw_multi_trim_res,
    iptw_sl_trim_res, iptw_gbm_trim_res)
```
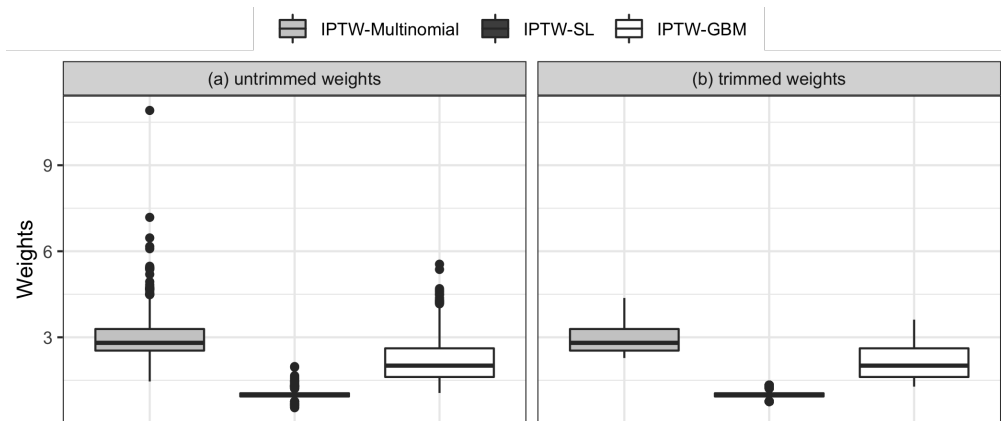

Figure 3: Distributions of the inverse probability of treatment weights estimated by multinomial logistic regression, super learner and generalized boosted models. Panel (a) shows results before weight trimming. Panel (b) displays results after trimming the weights at 5% and 95% of the distribution. Super learner and the generalized boosted models produced less extreme weights compared to multinomial logistic regression.

For VM, Lopez and Gutman (2017) proposed a rectangular support region defined by the maximum value of the smallest GPS and the minimum value of the largest GPS among the treatment groups. Individuals that fall outside the region are discarded from the causal analysis. This feature is automatically implemented with "VM" in **CIMTx**.

For BART, Hu et al. (2020a) supplied BART with a strategy to identify a common support region for retaining inferential units, which is to discard individuals with a large variability in their predicted potential outcomes. Specifically, for the ATT effects, any individual i with $W_i = w$ will be discarded if

$$s_i^{fw'} > \max_j\{s_j^{fw}\}, \forall j : W_j = w, w' \neq w \in \mathcal{W}, \tag{9}$$

where s_j^{fw} and $s_i^{fw'}$ respectively denote the standard deviation of the posterior distribution of the potential outcomes under treatment $W = w$ and $W = w'$, for a given sample j. For the ATE effects, the discarding rule in equation (9) is applied to each treatment group. Users can implement the discarding rule by setting the `discard` argument in **CIMTx**. Using $ATT_{1|1,2}$ as an example, 5 (`bart_dis_res$n_discard`) individuals in the reference group $w = 1$ were discarded from the simulated data.

```
bart_dis_res <- ce_estimate(y = data$y, x = data$covariates, w = data$w,
                            method = "BART", estimand = "ATT", discard = TRUE,
                            ndpost = 100, reference_trt = 1)
```

Sensitivity analysis for unmeasured confounding

The violation of the ignorability assumption (A3) can lead to biased treatment effect estimates. Sensitivity analysis is useful in gauging how much the causal conclusions will be altered in response to different magnitude of departure from the ignorability assumption. **CIMTx** implements a new flexible sensitivity analysis approach developed by Hu et al. (2022b). This approach first defines a confounding function for any pair of treatments (w, w') as

$$c(w, w', \boldsymbol{x}) = E\left[Y(w) \mid W = w, \boldsymbol{X} = \boldsymbol{x}\right] - E\left[Y(w) \mid W = w', \boldsymbol{X} = \boldsymbol{x}\right]. \tag{10}$$

The confounding function, also viewed as the sensitivity parameter in a sensitivity analysis, directly represents the difference in the mean potential outcomes $Y(w)$ between those treated with $W = w$ and those treated with $W = w'$, who have the same level of $\boldsymbol{x}$. If the ignorability assumption holds, the confounding function will be zero for all $w \in \mathcal{W}$. When treatment assignment is not ignorable, the unmeasured confounding is present and the causal effect estimates using measured $\boldsymbol{X}$ will be biased. Hu et al. (2022b) derived the form of the resultant bias as:

$$\begin{aligned} \text{Bias}(w, w') = {} & - p_w c(w', w, \boldsymbol{x}) + p_{w'} c(w, w', \boldsymbol{x}) \\ & - \sum_{l:l \in \mathcal{W} \setminus \{w, w'\}} p_l \left\{ c(w', l, \boldsymbol{x}) - c(w, l, \boldsymbol{x}) \right\}, \end{aligned} \tag{11}$$

where $p_w = P(W = w \mid \boldsymbol{X} = \boldsymbol{x})$, $w \neq w' \in \mathcal{W} = \{1, \dots, T\}$.

Table 2 demonstrates the plausible assumptions about the confounding functions and their interpretations. There are three ways in which we can specify the prior for the confounding functions: (i) point mass prior; (ii) re-analysis over a range of point mass priors (tipping point); (iii) full prior with uncertainty specified. Since the new sensitivity analysis approach was developed within the Bayesian framework, strategy (iii) offers an advantage of incorporating the statistical uncertainty due to sampling and the uncertainty about the values of the sensitivity parameters. In strategy (i), a fixed value is assumed for the sensitivity parameter. Strategy (ii) expands on strategy (i) and examines how the causal conclusion would change when a range of values are assumed for the sensitivity parameter. We will demonstrate all three cases of prior specifications with sa() function in **CIMTx** package. Hu et al. (2022b) further discussed (a) strategies to specify the confounding functions that represent our prior beliefs about the degrees of unmeasured confounding via the remaining variability in the outcomes unexplained by measured $\boldsymbol{X}$ (Hogan et al., 2014); and (b) ways in which the causal effects can be estimated adjusting for the presumed degree of unmeasured confounding.

Table 2: Interpretation of assumed priors on $c(w, w', \boldsymbol{x})$ and $c(w', w, \boldsymbol{x})$ for causal estimands based on the risk difference, assuming the outcome is an adverse event.

Prior assumption $c(w, w', \boldsymbol{x})$	$c(w', w, \boldsymbol{x})$	Interpretation and implications of the assumptions
> 0	< 0	Unhealthier individuals are treated with w.
< 0	> 0	Contrary to the above interpretation, unhealthier individuals are treated with w'.
< 0	< 0	The observed treatment allocation between w' and w is beneficial relative to the alternative which reverses treatment assignment for everyone.
> 0	> 0	Contrary to the above interpretation, the observed treatment allocation between w' and w is undesirable relative to the alternative which reverses treatment assignment for everyone.

The proposed sensitivity analysis algorithm proceeds with the following steps (Hu et al., 2022b):

1. Fit a multinomial probit BART model (Kindo et al., 2016) $f^{\text{MBART}}(A \mid X)$ to estimate the GPS, $p_l \equiv P(W = l \mid X = x) \; \forall l \in \mathcal{W}$, for each individual.
2. **for** $w \leftarrow 1$ to T **do**
    Draw M_1 GPS $\tilde{p}_{l1}, \dots, \tilde{p}_{lM_1}, \forall l \neq w \wedge l \in \mathcal{W}$ from the posterior predictive distribution of $f^{\text{MBART}}(W \mid \boldsymbol{X})$ for each individual.
    for $m \leftarrow 1$ to M_1 **do**
        Draw M_2 values $\eta^*_{lm1}, \dots, \eta^*_{lmM_2}$ from the prior distribution of each of the confounding functions $c(w, l, \boldsymbol{x})$, for each $l \neq j \wedge l \in \mathcal{W}$.
    end for
    end for

3. Compute the adjusted outcomes, $Y_i^{CF} \equiv Y_i - \sum_{l \neq w}^{T} P(W_i = l \mid X_i = x) c(w, l, x)$, for each treatment w, for each of $M_1 M_2$ draws of $\{\tilde{p}_{l1}, \eta_{l11}^*, \ldots, \eta_{l1M_2}^*, \ldots, \tilde{p}_{lM_1}, \eta_{lM_11}^*, \ldots, \eta_{lM_1M_2}^*; l \neq w \wedge l \in \mathscr{W}\}$.

4. Fit a BART model to each of $M_1 \times M_2$ sets of observed data with the adjusted outcomes Y^{CF}.

5. Estimate the combined adjusted causal effects and uncertainty intervals by pooling posterior samples across model fits arising from the $M_1 \times M_2$ data sets.

We now demonstrate the Monte Carlo sensitivity analysis approach for unmeasured confounding (Hu et al., 2022b). We first simulate a small dataset in a simple causal inference setting. There are two binary confounders: X_1 is measured and X_2 is unmeasured.

```
set.seed(111)
data_SA <- data_sim(
  sample_size = 100, n_trt = 3,
  x = c("rbinom(1, .5)",  # x1: measured confounder
        "rbinom(1, .4)"), # x2: unmeasured confounder
  lp_y = rep(".2*x1+2.3*x2", 3),# parallel response surfaces
  nlp_y = NULL,
  align = F, # w model is not the same as the y model
  lp_w = c("0.2 * x1 + 2.4 * x2",  # w = 1
           "-0.3 * x1 - 2.8 * x2"),# w = 2
  nlp_w = NULL,
  tau = c(-2, 0, 2),  delta = c(0, 0), psi = 1)
```

Next we implement the sensitivity analysis algorithm step-by-step.

1. Estimate the GPS for each individual. Specifically, we fit a multinomial probit BART model regressing treatment assignment on covariates, using mbart2() function from **BART** package. We set the number of posterior draws for the GPS (m1) to 50.

```
m1 <- 50; sample_gap <- 10
w_model <- BART::mbart2(x.train = data_SA$covariates, y.train = data_SA$w,
                        ndpost = m1 * sample_gap)
```

2. Then we draw the GPS for each individual from the fitted multinomial probit BART model.

```
gps <- array(w_model$prob.train[seq(1, m1 * sample_gap, sample_gap),],
             dim = c(m1,  # 1st dimension is M1
                     length(unique(data_SA$w)), # 2nd dimension is w
                     dim(data_SA$covariates)[1])) # 3rd dimension is sample size
dim(gps)

#> 50 3 100
```

The output of the posterior GPS is a three-dimensional array. The first dimension is the number of posterior draws for the GPS (M_1). The second dimension is the number of treatment W, and the third dimension is the total sample size.

3. Specify the prior distributions and the number of draws (M_2) for the confounding functions $c(w, w', x)$. In this illustrative simulation example, we use the true values of the confounding functions within each stratum of x_1. This represents the strategy (i) point mass prior.

```
x1 <- data_SA$covariates[, 1, drop = F]
x2 <- data_SA$covariates[, 2, drop = F] # x2 as the unmeasured confounder
w <- data_SA$w
x1w_data <- cbind(x1, w)
Y1 <- data_SA$y_true[, 1]
Y2 <- data_SA$y_true[, 2]
Y3 <- data_SA$y_true[, 3]
y <- data_SA$y
# Calculate the true confounding functions within x1 = 1 stratum
c_1_x1_1 <- mean(Y1[w == 1 & x1 == 1]) - mean(Y1[w == 2 & x1 == 1]) # c(1,2)
c_2_x1_1 <- mean(Y2[w == 2 & x1 == 1]) - mean(Y2[w == 1 & x1 == 1]) # c(2,1)
c_3_x1_1 <- mean(Y2[w == 2 & x1 == 1]) - mean(Y2[w == 3 & x1 == 1]) # c(2,3)
c_4_x1_1 <- mean(Y1[w == 1 & x1 == 1]) - mean(Y1[w == 3 & x1 == 1]) # c(1,3)
c_5_x1_1 <- mean(Y3[w == 3 & x1 == 1]) - mean(Y3[w == 1 & x1 == 1]) # c(3,1)
c_6_x1_1 <- mean(Y3[w == 3 & x1 == 1]) - mean(Y3[w == 2 & x1 == 1]) # c(3,2)
```

```
c_x1_1 <- cbind(c_1_x1_1, c_2_x1_1, c_3_x1_1, c_4_x1_1, c_5_x1_1,
                c_6_x1_1)# True confounding functions among x1 = 1
```

The true values of the confounding functions within the stratum $x_1 = 0$ can be calculated in a similar way.

```
c_1_x1_0 <- mean(Y1[w == 1 & x1 == 0]) - mean(Y1[w == 2 & x1 == 0])# c(1,2)
c_2_x1_0 <- mean(Y2[w == 2 & x1 == 0]) - mean(Y2[w == 1 & x1 == 0])# c(2,1)
c_3_x1_0 <- mean(Y2[w == 2 & x1 == 0]) - mean(Y2[w == 3 & x1 == 0])# c(2,3)
c_4_x1_0 <- mean(Y1[w == 1 & x1 == 0]) - mean(Y1[w == 3 & x1 == 0])# c(1,3)
c_5_x1_0 <- mean(Y3[w == 3 & x1 == 0]) - mean(Y3[w == 1 & x1 == 0])# c(3,1)
c_6_x1_0 <- mean(Y3[w == 3 & x1 == 0]) - mean(Y3[w == 2 & x1 == 0])# c(3,2)
c_x1_0 <- cbind(c_1_x1_0, c_2_x1_0, c_3_x1_0, c_4_x1_0, c_5_x1_0, c_6_x1_0)
```

The true values of the confounding functions within the stratum of x_1 can be calculated using the helper function `true_c_fun_cal()` in our package.

```
true_c_fun <- true_c_fun_cal(x = x1, w = w)
```

4. Calculate the confounding function adjusted outcomes with the drawn values of GPS and confounding functions.

```
i <- 1; j <- 1
ycf <- ifelse(
  x1w_data[, "w"] == 1 & x1 == 1,
  # w = 1, x1 = 1
  y - (c_x1_1[i, 1] * gps[j, 2, ] + c_x1_1[i, 4] * gps[j, 3, ]),
  ifelse(
    x1w_data[, "w"] == 1 & x1 == 0,
    # w = 1, x1 = 0
    y - (c_x1_0[i, 1] * gps[j, 2, ] + c_x1_0[i, 4] * gps[j, 3, ]),
    ifelse(
      x1w_data[, "w"] == 2 & x1 == 1,
      # w = 2, x1 = 1
      y - (c_x1_1[i, 2] * gps[j, 1, ] + c_x1_1[i, 3] * gps[j, 3, ]),
      ifelse(
        x1w_data[, "w"] == 2 & x1 == 0,
        # w = 2, x1 = 0
        y - (c_x1_0[i, 2] * gps[j, 1, ] + c_x1_0[i, 3] * gps[j, 3, ]),
        ifelse(
          x1w_data[, "w"] == 3 & x1 == 1,
          # w = 3, x1 = 1
          y - (c_x1_1[i, 5] * gps[j, 1, ] + c_x1_1[i, 6] * gps[j, 2, ]),
          # w = 3, x1 = 0
          y - (c_x1_0[i, 5] * gps[j, 1, ] + c_x1_0[i, 6] * gps[j, 2, ])
        )
      )
    )
  )
)
```

5. Use the adjusted outcomes to estimate the causal effects.

```
bart_mod_sa <- BART::wbart(x.train = x1w_data, y.train = ycf, ndpost = 1000)
predict_1_ate_sa <- BART::pwbart(cbind(x1, w = 1), bart_mod_sa$treedraws)
predict_2_ate_sa <- BART::pwbart(cbind(x1, w = 2), bart_mod_sa$treedraws)
predict_3_ate_sa <- BART::pwbart(cbind(x1, w = 3), bart_mod_sa$treedraws)
RD_ate_12_sa <- rowMeans(predict_1_ate_sa - predict_2_ate_sa)
RD_ate_23_sa <- rowMeans(predict_2_ate_sa - predict_3_ate_sa)
RD_ate_13_sa <- rowMeans(predict_1_ate_sa - predict_3_ate_sa)
predict_1_att_sa <- BART::pwbart(cbind(x1[w == 1,], w = 1), bart_mod_sa$treedraws)
predict_2_att_sa <- BART::pwbart(cbind(x1[w == 1,], w = 2), bart_mod_sa$treedraws)
RD_att_12_sa <- rowMeans(predict_1_att_sa - predict_2_att_sa) # w=1 is the reference
```

Repeat steps 3 and 4 $M_1 \times M_2$ times to form $M_1 \times M_2$ datasets with adjusted outcomes. The uncertainty intervals are estimated by pooling the posteriors across the $M_1 \times M_2$ model fits.

The sa() function implements the sensitivity analysis approach while fitting the $M_1 \times M_2$ models using parallel computation.

```
sa_res <- sa(m1 = 50, m2 = 1, x = x1, y = data_SA$y, w = data_SA$w, ndpost = 100,
             estimand = "ATE", prior_c_function =  true_c_fun, nCores = 3)
```

```
summary(sa_res)
```

```
#>            EST   SE LOWER UPPER
#> ATE_RD12 -0.44 0.10 -0.64 -0.23
#> ATE_RD13 -0.58 0.11 -0.80 -0.36
#> ATE_RD23 -0.14 0.12 -0.38  0.10
```

We compare the sensitivity analysis results to the naive estimators where we ignore the unmeasured confounder X_2, and to the results where we had access to X_2.

```
bart_with_x2_res <- ce_estimate(y = data_SA$y, x = cbind(x1, x2), w = data_SA$w,
                                method = "BART", estimand = "ATE", ndpost = 100)
bart_without_x2_res <- ce_estimate(y = data_SA$y, x = x1, w = data_SA$w,
                                method = "BART", estimand = "ATE", ndpost = 100)
```

Figure 4 compares the estimates of $ATE_{1,2}$, $ATE_{2,3}$ and $ATE_{1,3}$ from the three analyses. The sensitivity analysis estimators are similar to the results that could be achieved had the unmeasured confounder X_2 been made available.
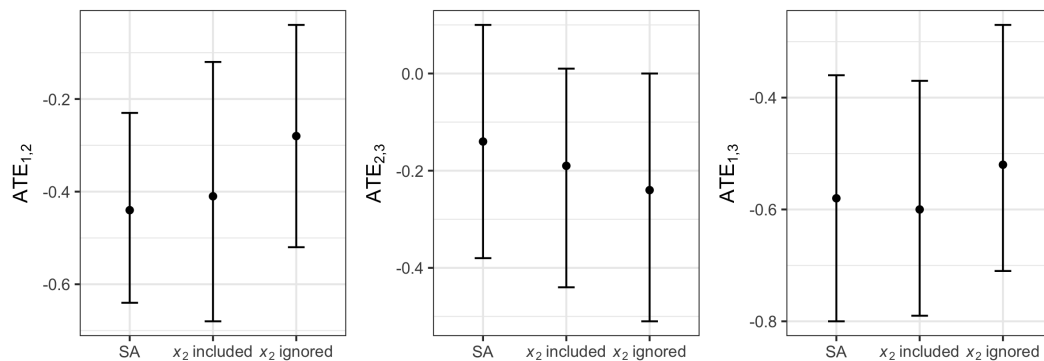

Figure 4: Estimates and 95% credible intervals for $ATE_{1,2}$, $ATE_{2,3}$ and $ATE_{1,3}$

We can also conduct the sensitivity analysis for the ATT effects by setting estimand = "ATT".

```
sa_att_res <- sa(m1 = 50, m2 = 1, x = x1, y = data_SA$y, w = data_SA$w, ndpost = 100,
             estimand = "ATT", prior_c_function =  true_c_fun, nCores = 1,
             reference_trt = 1)
```

```
summary(sa_att_res)
```

```
#>            EST   SE LOWER UPPER
#> ATT_RD12 -0.42 0.10 -0.63 -0.22
#> ATT_RD13 -0.57 0.11 -0.79 -0.35
```

Finally, we demonstrate the sa() function in a more complex data setting with 3 measured confounders and 2 unmeasured confounders.

```
set.seed(1)
data_SA_2 <- data_sim(
  sample_size = 100, n_trt = 3,
  x = c( "rnorm(0, 0.5)",  # x1
         "rbeta(2, .4)",   # x2
         "runif(0, 0.5)",  # x3
         "rweibull(1, 2)", # x4 as one of the unmeasured confounders
         "rbinom(1, .4)" ),  # x5 as one of the unmeasured confounders
  lp_y = rep(".2*x1 + .3*x2 - .1*x3 - 1.1*x4 - 1.2*x5", 3),
  nlp_y  = rep(".7*x1*x1  - .1*x2*x3", 3), # parallel response surfaces
  align = FALSE,
```

```
  lp_w =  c(".4*x1 + .1*x2 - 1.1*x4 + 1.1*x5",  # w = 1
            ".2*x1 + .2*x2 - 1.2*x4 - 1.3*x5"), # w = 2,
  nlp_w = c("-.5*x1*x4 - .1*x2*x5",  # w = 1
            "-.3*x1*x4 + .2*x2*x5"), # w = 2,
  tau = c(0.5,-0.5,0.5), delta = c(0.5,0.5), psi = 2)
```

We have demonstrated the strategy (i) point mass prior, and now show how strategy (ii) re-analysis over a range of point mass priors and (iii) full prior with uncertainty specified can be used. For strategy (ii), we can specify the grid of the specific confounding function using seq() function. In the following example, we will set $c(1,3)$ as a grid of 5 negative numbers from -0.6 to 0 with an increment of 0.15, and set $c(3,1)$ as a grid of 5 positive numbers from 0 to 0.6 with an increment of 0.15. This specification encodes our belief that unhealthier (suppose the outcome is death) individuals are treated with treatment option 3 (see Table 2) because those receiving $w = 1$ would have lower probability of death to either treatment. The other confounding functions are drawn from a uniform distribution (strategy (iii)).

```
c_grid <- c("runif(-.6, 0)", "runif(0,.6)", "runif(-.6,0)", # c(1,2), c(2,1), c(2,3)
            "seq(-.6, 0,.15)", "seq(0,.6,.15)", "runif(0,.6)") # c(1,3), c(3,1), c(3,2)
SA_grid_res <- sa(y = data_SA_2$y, w = data_SA_2$w, x = data_SA_2$covariates[,-c(4,5)],
                  prior_c_function = c_grid, m1 = 1, nCores = 3, estimand = "ATE")
```

The sensitivity analysis results can be visualized via a contour plot. Figure 5 shows how the estimate of $ATE_{1,3}$ would change under different values of the two confounding functions $c(1,3,x)$ and $c(3,1,x)$. Under the assumption that unhealthier patients are treated with $w = 3$, when the effect of unmeasured confounding increases (moving up along the $-45°$ line), the beneficial treatment effect associated with $w = 3$ becomes more pronounced evidenced by larger estimates of $ATE_{1,3}$.
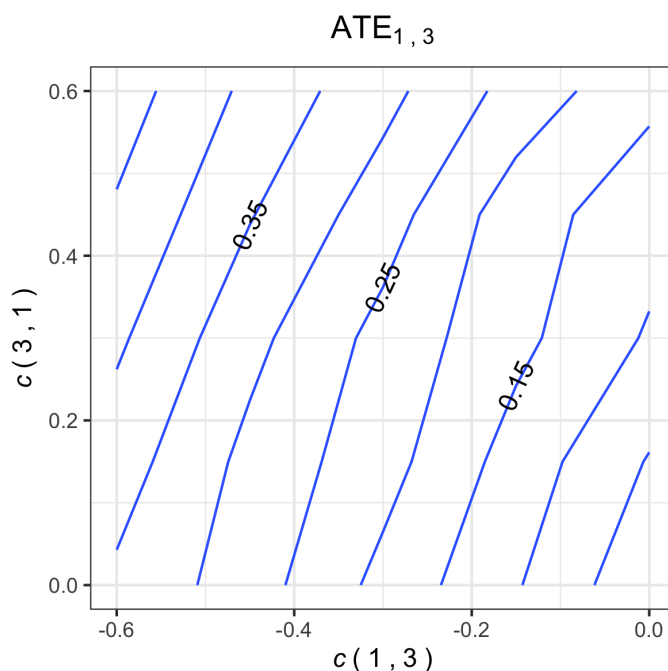
```
plot(SA_grid_res, ATE = "1,3")
```


Figure 5: Contour plot of the confounding function adjusted $ATE_{1,3}$. The blue lines report the adjusted causal effect estimates corresponding to pairs of values for $c(1,3)$ and $c(3,1)$ spaced on a grid $(-0.6,0) \times (0,0.6)$ incremented by 0.15, and under the prior distributions: $c(1,2), c(2,3) \sim U(-0.6,0); c(2,1), c(2,3) \sim U(0,0.6)$.

4   Discussion

We contribute a comprehensive R package **CIMTx** suitable for causal analysis of observational data with multiple treatments and a binary outcome. In this package, we introduce six methods for the

estimation of causal effects, including both the classical approaches and machine learning based methods. Drawing causal inference from non-experimental data inevitably involves structural causal assumptions. **CIMTx** offers a unique set of features to address two key assumptions: positivity and ignorability, using appropriate estimation procedures. Additionally, the **CIMTx** package provides guidance to readers on how to simulate data possessing the data characteristics in the multiple treatment setting. Detailed step-by-step examples are provided to demonstrate all methods. The current version of the **CIMTx** package focuses on binary outcomes. For future research, developing methods and R packages for causal inferences with more complex outcomes such as censored survival outcomes (Hu et al., 2022a) could be a worthwhile contribution.

Bibliography

P. C. Austin. Variance estimation when using inverse probability of treatment weighting (IPTW) with survival analysis. *Statistics in Medicine*, 35(30):5642–5655, 2016. [p219]

H. A. Chipman, E. I. George, and R. E. McCulloch. BART: Bayesian additive regression trees. *The Annals of Applied Statistics*, 4(1):266–298, 2010. [p220]

S. R. Cole and M. A. Hernán. Constructing inverse probability weights for marginal structural models. *American Journal of Epidemiology*, 168(6):656–664, 2008. [p219]

M. Erik von Elm, D. G. Altman, M. Egger, et al. The Strengthening the Reporting of Observational Studies in Epidemiology (STROBE) statement: guidelines for reporting observational studies. *Annals of Internal Medicine*, 147(8):573—-577, 2007. [p213]

P. Feng, X.-H. Zhou, Q.-M. Zou, M.-Y. Fan, and X.-S. Li. Generalized propensity score for estimating the average treatment effect of multiple treatments. *Statistics in Medicine*, 31(7):681–697, 2012. [p213, 219]

C. Fong, M. Ratkovic, and K. Imai. *CBPS: Covariate Balancing Propensity Score*, 2021. URL https://CRAN.R-project.org/package=CBPS. R package version 0.22. [p214]

J. M. Franklin, W. Eddings, P. C. Austin, E. A. Stuart, and S. Schneeweiss. Comparing the performance of propensity score methods in healthcare database studies with rare outcomes. *Statistics in Medicine*, 36(12):1946–1963, 2017. [p220]

N. Greifer. *optweight: Targeted Stable Balancing Weights Using Optimization*, 2019. URL https://CRAN.R-project.org/package=optweight. R package version 0.2.5. [p214]

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. [p214]

J. L. Hill. Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics*, 20(1):217–240, 2011. [p213, 220]

J. W. Hogan, M. J. Daniels, and L. Hu. A bayesian perspective on assessing sensitivity to assumptions about unobserved data. In G. Molenberghs, G. Fitzmaurice, M. G. Kenward, A. Tsiatis, and G. Verbeke, editors, *Handbook of Missing Data Methodology*, chapter 18, pages 405–434. Boca Raton, FL: CRC Press, 2014. [p223]

P. W. Holland. Statistics and causal inference. *Journal of the American Statistical Association*, 81(396): 945–960, 1986. [p217]

D. G. Horvitz and D. J. Thompson. A generalization of sampling without replacement from a finite universe. *Journal of the American Statistical Association*, 47(260):663–685, 1952. [p218]

L. Hu and C. Gu. Estimation of causal effects of multiple treatments in healthcare database studies with rare outcomes. *Health Services and Outcomes Research Methodology*, 21(3):287–308, 2021. [p213, 220]

L. Hu, C. Gu, M. Lopez, J. Ji, and J. Wisnivesky. Estimation of causal effects of multiple treatments in observational studies with a binary outcome. *Statistical Methods in Medical Research*, 29(11): 3218–3234, 2020a. [p213, 217, 220, 222]

L. Hu, B. Liu, J. Ji, and Y. Li. Tree-based machine learning to identify and understand major determinants for stroke at the neighborhood level. *Journal of the American Heart Association*, 9(22):e016745, 2020b. [p220]

L. Hu, J. Ji, and F. Li. Estimating heterogeneous survival treatment effect in observational data using machine learning. *Statistics in Medicine*, 40(21):4691–4713, 2021a. [p215, 220]

L. Hu, J.-Y. Joyce Lin, and J. Ji. Variable selection with missing data in both covariates and outcomes: Imputation and machine learning. *Statistical Methods in Medical Research*, 30(12):2651–2671, 2021b. [p213]

L. Hu, J.-Y. J. Lin, K. Sigel, and M. Kale. Estimating heterogeneous survival treatment effects of lung cancer screening approaches: A causal machine learning analysis. *Annals of Epidemiology*, 62:36–42, 2021c. [p220]

L. Hu, J. Ji, R. D. Ennis, and J. W. Hogan. A flexible approach for causal inference with multiple treatments and clustered survival outcomes. *arXiv preprint*, 2022a. arXiv:2202.08318. [p228]

L. Hu, J. Zou, C. Gu, J. Ji, M. Lopez, and M. Kale. A flexible sensitivity analysis approach for unmeasured confounding with multiple treatments and a binary outcome with application to SEER-Medicare lung cancer data. *The Annals of Applied Statistics*, 16(2):1014–1037, 2022b. [p213, 223, 224]

G. W. Imbens. The role of the propensity score in estimating dose-response functions. *Biometrika*, 87(3): 706–710, 2000. [p219]

G. W. Imbens and D. B. Rubin. *Causal inference in statistics, social, and biomedical sciences*. Cambridge University Press, 2015. [p217]

J. D. Kang, J. L. Schafer, et al. Demystifying double robustness: A comparison of alternative strategies for estimating a population mean from incomplete data. *Statistical Science*, 22(4):523–539, 2007. [p219]

B. P. Kindo, H. Wang, and E. A. Peña. Multinomial probit bayesian additive regression trees. *Stat*, 5(1): 119–131, 2016. [p223]

J. Kruschke. *Doing Bayesian data analysis: A tutorial with R, JAGS, and Stan*. Academic Press, 2014. [p218]

B. K. Lee, J. Lessler, and E. A. Stuart. Weight trimming and propensity score weighting. *PLOS One*, 6 (3):e18174, 2011. [p219]

A. Linden, S. D. Uysal, A. Ryan, and J. L. Adams. Estimating causal effects for multivalued treatments: a comparison of approaches. *Statistics in Medicine*, 35(4):534–552, 2016. [p213, 217]

R. J. Little. Missing-data adjustments in large surveys. *Journal of Business & Economic Statistics*, 6(3): 287–296, 1988. [p219]

M. J. Lopez and R. Gutman. Estimation of causal effects with multiple treatments: a review and new ideas. *Statistical Science*, 32(3):432–454, 2017. [p213, 221, 222]

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. [p213, 219]

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. [p214]

S. Rose and S.-L. Normand. Double robust estimation for multiple unordered treatments and clustered observations: Evaluating drug-eluting coronary artery stents. *Biometrics*, 75(1):289–296, 2019. [p213, 221]

D. B. Rubin. The use of matched sampling and regression adjustment to remove bias in observational studies. *Biometrics*, pages 185–203, 1973. [p213, 217]

D. B. Rubin. Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of educational Psychology*, 66(5):688, 1974. [p217]

D. B. Rubin. Randomization analysis of experimental data: The Fisher randomization test comment. *Journal of the American Statistical Association*, 75(371):591–593, 1980. [p217]

M. J. Van der Laan, E. C. Polley, and A. E. Hubbard. Super learner. *Statistical Applications in Genetics and Molecular Biology*, 6(1), 2007. [p219]

T. Zhou, G. Tong, F. Li, L. Thomas, and F. Li. *PSweight: Propensity Score Weighting for Causal Inference with Observational Studies and Randomized Trials*, 2020. URL https://CRAN.R-project.org/package=PSweight. R package version 1.1.2. [p214]

Liangyuan Hu
Rutgers University School of Public Health
Department of Biostatistics and Epidemiology
683 Hoes Lane West
Piscataway, NJ 08854, United States of America
liangyuan.hu@rutgers.edu

Jiayi Ji
Rutgers University School of Public Health
Department of Biostatistics and Epidemiology
683 Hoes Lane West
Piscataway, NJ 08854, United States of America
jj869@sph.rutgers.edu