Journal of Machine Learning Research (2020) Submitted ; Published

Continuous Treatment Effect Estimation via Generative Adversarial De-confounding

Yunzhe Li* ILLUSIVEMANLYZ@GMAIL.COM
Kun Kuang* KUNKUANG@ZJU.EDU.CN
Zhejiang University

Bo Li LIBO@SEM.TSINGHUA.EDU.CN
Peng Cui CUIP@TSINGHUA.EDU.CN
Tsinghua University

Jianrong Tao HZTAOJIANRONG@CORP.NETEASE.COM
NetEase Fuxi AI Lab

Hongxia Yang YANG.YHX@ALIBABA-INC.COM
Alibaba Group

Fei Wu WUFEI@CS.ZJU.EDU.CN
Zhejiang University

Editor: Thuc Duy Le, Lin Liu, Kun Zhang, Emre Kıcıman, Peng Cui, and Aapo Hyvärinen

Abstract

One fundamental problem in causal inference is the treatment effect estimation in observational studies, and its key challenge is to handle the confounding bias induced by the associations between covariates and treatment variable. In this paper, we study the problem of effect estimation on continuous treatment from observational data, going beyond previous work on binary treatments. Previous work for binary treatment focuses on de-confounding by balancing the distribution of covariates between the treated and control groups with either propensity score or confounder balancing techniques. In the continuous setting, those methods would fail as we can hardly evaluate the distribution of covariates under each treatment status. To tackle the case of continuous treatments, we propose a novel Generative Adversarial De-confounding (GAD) algorithm to eliminate the associations between covariates and treatment variable with two main steps: (1) generating an "calibration" distribution without associations between covariates and treatment by random perturbation; (2) learning sample weight that transfer the distribution of observed data to the "calibration" distribution for de-confounding with a Generative Adversarial Network. Extensive experiments on both synthetic and real-world datasets demonstrate that our algorithm outperforms the state-of-the-art methods for effect estimation of continuous treatment with observational data.

Keywords: Treatment Effect, Continuous Treatment, Generative Adversarial De-confounding, Causal Inference

∗. These authors contributed equally.

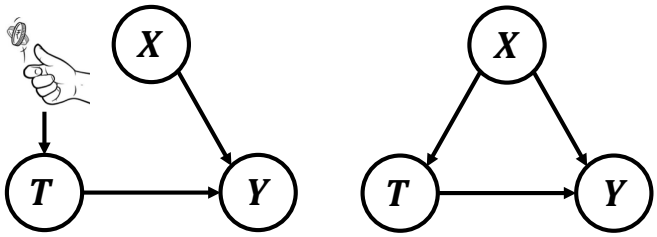
1. Introduction

Causal inference (Holland, 1986), which refers to the process of drawing a conclusion about a causal connection based on the conditions of the occurrence of an effect, is a powerful statistical modeling tool for explanatory analysis. Treatment effect estimation is one fundamental problem in causal inference and gains an essential role for explainable decision making with answering the counterfactual questions (Rubin, 1974; Pearl, 2009), for example, how many doses of a medication will cause better outcomes for patients. Pearl (2009) demonstrates that the gold standard approach for treatment effect estimation is to run a Randomized Controlled Trial (RCT), for example, A/B testing, where the treatment is randomly assigned to units[1] and does not depend on the covariates as shown in Figure 1a. In many real applications, however, fully randomized experiments are always expensive, unethical, or even infeasible (Kohavi and Longbotham, 2011). In this paper, hence, we focus on approximately estimating the treatment effect from off-line data collected from observational studies. In such datasets, the assignment of treatment depends on the covariates as we shown in Figure 1b, leading to confounding bias between treatment and covariates, i.e., $P(T|\mathbf{X}) \neq P(T)$. Therefore, confounding bias removing is the key challenge for treatment effect estimation in observational sturdies.

In literature, many methods have been proposed for effect estimation with binary treatment (treated or control), including matching methods (Kallus, 2019; Liu et al., 2019), propensity score based methods (Rosenbaum and Rubin, 1983; Bang and Robins, 2005; Austin, 2011; Kuang et al., 2017a, 2020a) and confounder balancing techniques (Hainmueller, 2012; Kuang et al., 2017b; Athey et al., 2018; Kuang et al., 2019). The motivation of these methods is to remove the association between treatment and covariates for deconfounding. Matching methods (Liu et al., 2019) proposed to match units with almost the same covariates but different treatment. Inverse of propensity weighting (IPW) (Austin, 2011) attempted to re-weight samples for removing confounding bias between treatment and covariates. Confounder balancing methods (Kuang et al., 2017b) proposed to balance the distribution of covariates between treated and control groups. These methods achieved promising performance in treatment effect estimation (Kuang et al., 2020b), however, all of them focus on the binary treatment and cannot be applied for estimating the causal effect of continuous treatment.

The classical methods for estimating continuous treatment effect are based on regression models, including Y-model (Imbens, 2004; Hill, 2011) to regress outcome Y on the covariates and treatment, T-model (Hirano and Imbens, 2004; Imai and Van Dyk, 2004; Galvao and Wang, 2015; Galagate, 2016) to regress the treatment T on the covariates, and doubly robust methods (Robins and Rotnitzky, 2001) by combining both Y-model and T-model. The performance of these methods entirely relies on the correct specification of their models. Recently, a non-parametric covariate balancing generalized propensity score (Fong et al., 2018) was proposed to minimize the association between the covariates and treatment for deconfounding, and achieved great performance in real applications. However, it is limited by its linear assumption on T-model. Galagate (2016) extended IPW for continuous treatment with considering second moments of covariates, but it needs to assume the linear correlation

1. Units represent the objects of treatment. For example, in medical experiments, the units refer to the patients who take a particular medication.



(a) Randomized Controlled Trial (RCT) (b) Observational Studies

Figure 1: Casual structure for RCT and observational studies, where $\mathbf{X}$ denotes the observed covariates, T refers to the treatment variable, and Y is the outcome. In RCT, the treatment is independent with covariates, while in observational studies, the treatment is affected by the covariates.

between Y and T. Overall, if one have NO prior knowledge on the grounded models, existing methods for continuous treatment cannot fully remove the confounding bias in observational studies, leading to imprecise estimation of continuous treatment effect.

To fully remove confounding bias in observational studies, we propose a non-parametric data-driven method, named Generative Adversarial De-confounding (GAD) algorithm by sample re-weighting techniques. Specifically, there are two main components in our GAD algorithm, including "calibration" distribution generation and approximation. Firstly, we generate an "calibration" distribution by randomly shuffling the value of each covariate across units, such that each covariate would become independent with the treatment, where the confounding bias are fully removed. Then, we propose a sample weight learning schema on the observed data for approximating the the "calibration" distribution with a Generative Adversarial Network (GAN), achieving de-confounding between continuous treatment and covariates. We validate our GAD algorithm with extensive experiments on both synthetic and real datasets. The experimental results clearly show that our algorithm outperforms the state-of-the-art methods on continuous treatment effect estimation in observational studies.

The main contributions of this paper are summarized as follows:

- We investigate the problem of causal effect estimation with continuous treatment from observational data, going beyond previous work on binary treatments.
- We propose a novel Generative Adversarial De-confounding (GAD) algorithm to learn a sample weight for removing the associations between treatment and covariates, and estimating the causal effect of continuous treatment.
- Extensive experiments on both synthetic and real world datasets demonstrate the superior performance of our proposed algorithms on the problem of continuous treatment effect estimation with observational data.

The rest of this paper is organized as follows. Section 2 reviews the related work. Section 3 gives the notations and formulates our problem. The details of our proposed algorithm for continuous treatment effect estimation are introduced in Section 4. Experimental results and analyses are reported in Section 5. Finally, Section 6 concludes the paper.

2. Related Work

Previous work on treatment effect estimation in observational studies can be categorized by the type of treatment variable as causal effect estimation on binary treatment and continuous treatment.

On binary treatment. The classical method for causal effect estimation on binary treatment is propensity score based methods (Rosenbaum and Rubin, 1983; Bang and Robins, 2005; Chan et al., 2010; Austin, 2011; Kuang et al., 2017a, 2020a). The propensity score was first proposed by Rosenbaum and Rubin (1983), where it was estimated via a logistic regression. Then many other machine learning algorithms (e.g., boosting regression by McCaffrey et al. 2004, bagged CART and neural network by Westreich et al. 2010, LASSO by Chernozhukov et al. 2016) are employed for estimating propensity score. Various methods have been proposed based on propensity score, such as propensity score matching, inverse propensity weighting, doubly robust estimators and data-driven variable decomposition (Bang and Robins, 2005; Chan et al., 2010; Austin, 2011; Kuang et al., 2017a, 2020a). However, these estimators require correct model specification on treatment assignment or precise estimation of the propensity score, which may not be the case in many applications. Moreover, these methods focus on the causal effect estimation on binary treatment.

Bypassing propensity score estimation, recently, researchers proposed direct confounder balancing via sample weight learning (Imai and Ratkovic, 2014; Kuang et al., 2017b; Athey et al., 2018; Kuang et al., 2019). Imai and Ratkovic (2014) introduced covariates balancing propensity score, which models treatment assignment while optimizing covariates balancing. Kuang et al. (2017b, 2019) proposed a differentiated variable balancing algorithm by jointly optimizing sample weight and variable weight. Athey et al. (2018) proposed approximate residual balancing algorithm, which combines outcome modeling using the LASSO with balancing weight constructed to approximately balance covariates between treatment and control groups. These methods achieved good performance in many real applications for treatment effect estimation, but all of these methods also only focused on the problem of binary treatment and cannot be directly applied to continuous treatment.

On continuous treatment. In practice, the most common approach for estimating continuous treatment effect is regression model based, including Y-model (Imbens, 2004; Hill, 2011) and T-model (Hirano and Imbens, 2004; Imai and Van Dyk, 2004; Galvao and Wang, 2015). Y-model method refers to the regression modeling of how the outcome Y relates to covariates and treatment. T-model methods mainly adapted propensity score based approaches to model how the treatment T relates to the covariates, saying modeling treatment assignment mechanism. However, the performance of these methods relies entirely on the correct specification of either the outcome model or the treatment model. By combining Y-model and T-model, many doubly robust estimators (Robins and Rotnitzky, 2001) are proposed and achieved consistent estimation of effects of continuous treatment as long as one of two models is correctly specified and modeled well enough.

Recently, many non-parametric methods (Neugebauer and van der Laan, 2007; Kennedy et al., 2017; Fong et al., 2018) have been proposed to reduce the model dependency for continuous treatment effect estimation. Neugebauer and van der Laan (2007) extended traditional parametric marginal structural model to an nonparametric one and does not require correct specification of a parametric model but instead relies on a working model

Symbol	Definition
n	Sample size
p	Dimension of observed variables
$T \in \mathbb{R}^{n\times 1}$	Treatment
$T' \in \mathbb{R}^{n\times 1}$	Treatment after randomly shuffling
$Y \in \mathbb{R}^{n\times 1}$	Outcome
$\mathbf{X} \in \mathbb{R}^{n\times p}$	Observed variables
$\mathbf{w} \in \mathbb{R}^{n\times 1}$	Sample weight

Table 1: Symbols and definitions.

for precise prediction. Kennedy et al. (2017) developed a kernel smoothing based non-parametric method for doubly robust estimation of continuous treatment effect, allowing for misspecification of either the treatment model or outcome model. Fong et al. (2018) proposed a non-parametric covariate balancing generalized propensity score to minimize the association between the covariates and treatment, however, it only focused on the linear association and would fail if the true T-model is non-linear.

3. Problem and Assumptions

In this paper, we focus on continuous treatment effect estimation based on potential outcome framework (Imbens and Rubin, 2015) as shown in Figure 1b. With the framework, we define a treatment as a random variable T and a potential outcome as $Y(t)$ which corresponds to a specific treatment $T = t$. The continuous treatment of interest can take values in $t \in \mathcal{T}$, where $\mathcal{T}$ is an interval $[t_0, t_1]$. Then, for each unit indexed by $i = 1, 2, \cdots, n$, we observe a treatment T_i, an outcome Y_i^{obs} and a vector of observed variables $X_i \in \mathbb{R}^{p\times 1}$, where the observed outcome Y_i^{obs} of unit i is corresponding to its treatment and denotes as $Y_i^{obs} = Y(T_i)$. The number of units are equal to n and the dimension of all observed variables is p. Table 1 summarized the symbol and definition. In our paper, for any column vector $\mathbf{v} = (v_1, v_2, \cdots, v_m)^T$, let $\|\mathbf{v}\|_\infty = \max(|v_1|, \cdots, |v_m|)$, $\|\mathbf{v}\|_2^2 = \sum_{i=1}^m v_i^2$, and $\|\mathbf{v}\|_1 = \sum_{i=1}^m |v_i|$.

The important goal of causal inference in observational studies is to evaluate the casual effect of treatment T on outcome Y. In the setting with continuous treatment, the causal effect of treatment can be captured by the *Average Dose Response Function* (ADRF) and *Marginal Treatment Effect Function* (MTEF) (Kreif et al., 2015). The ADRF refers to the expectation of potential outcome $Y(t)$ on each treatment status t over all units. Formally, the ADRF on treatment t is defined as:

$$ADRF(t) = \mathbb{E}[Y_i(t)]. \tag{1}$$

The MTEF represents the effect of increasing the level of treatment on the expected potential outcome over all units. Formally, the MTEF is defined as:

$$MTEF = \frac{\mathbb{E}[Y_i(t)] - \mathbb{E}[Y_i(t - \triangle t)]}{\triangle t}, \tag{2}$$

where $Y_i(t)$ represents the potential outcome of units i with treatment status $T = t$ and $\mathbb{E}(\cdot)$ refers to the expectation function. $\triangle t$ denotes the increasing the level of treatment, for example, with $\triangle t = 1$, MTEF captures the incremental change in the potential outcome, for a unit change in the level of treatment.

The Eq. (1) and Eq. (2) are infeasible because of the counterfactual problem (Chan et al., 2010), that for each unit i with treatment status $T = t$, we can only observe one of the potential outcomes $Y_i(t)$, the other potential outcomes $Y_i(t'), t' \in \mathcal{T} \setminus t$ are unobserved or counterfactual. One can address this counterfactual problem by approximate the unobserved potential outcome. The simplest approach is to directly estimate the ARDF $\mathbb{E}[Y_i(t)]$ on treatment level $T = t$ only over the units with that treatment. However, in observational studies, the treatment is not randomly assigned to units as we shown in Figure 1b, leading to the confounding bias between treatment and covariates (Chan et al., 2010), saying the distribution of covariates would be different over the units with different treatment level.

To address the counterfactual problem and confounding bias issue, throughout this paper, we assume the following standard assumptions (Rosenbaum and Rubin, 1983) are satisfied.

Assumption 1: Stable Unit Treatment Value. Given the observed covariates, the distribution of potential outcome for one unit is assumed to be unaffected by the particular treatment assignment of another unit.

Assumption 2: Unconfoundedness. Given the observed covariates, the distribution of treatment is independent of potential outcome. Formally, $T \perp Y(t)|\mathbf{X}, \forall t \in \mathcal{T}$.

Assumption 3: Overlap. Every unit has a nonzero probability to receive either treatment status when given the observed covariates. Formally, $P(r(T = t, \mathbf{X} = x) > 0) = 1$, where $r(T = t, \mathbf{X} = x) = f_{T|\mathbf{X}}(t|x)$ denotes the conditional density of treatment given covariates.

Under these assumptions, we propose a sample re-weighting technique for removing the confounding bias between treatment T and covariates $\mathbf{X}$. The re-weighting method forms the surrogates of the unobserved potential outcome $Y_i(t)$ over all units by re-weighting units with sample weight $\mathbf{w} \in \mathbb{R}^{n\times 1}$ to make the treatment T become independent with the covariates $\mathbf{X}$. Then, the unobserved potential outcome $Y_i(t)$ over all units can be approximated by the observed outcome $Y_i(t)$ over the units with treatment $T = t$. Finally, with the learned sample weight $\mathbf{w}$, we can approximately estimate the ADRF on each treatment level t by:

$$\widehat{ADRF} = \sum_{i:T_i=t} w_i \cdot Y_i(t). \tag{3}$$

Similarity, we can also approximately estimate the MTEF as:

$$\widehat{MTEF} = \frac{\sum_{i:T_i=t} w_i \cdot Y_i(t) - \sum_{i:T_i=t-\triangle t} w_i \cdot Y_i(t)}{\triangle t}. \tag{4}$$

4. Method

In this section, we give the details of our proposed Generative Adversarial De-confounding (GAD) algorithm for continuous treatment effect estimation in observational studies.

4.1 Generative Adversarial De-confounding Algorithm

To fully remove the confounding bias induced by the dependency between treatment T and covariates $\mathbf{X}$ in observational studies as shown in Figure 1b, we propose to make treatment T become independent with the covariates $\mathbf{X}$ by sample re-weighting, that is our Generative Adversarial De-confounding (GAD) algorithm. In our GAD algorithm, there are two key components: (i) "calibration" distribution generation: Based on the observed data $\mathbf{D}_{obs} = \{T, \mathbf{X}\}$, we generate an "calibration" data $\mathbf{D}_{cal} = \{T, \mathbf{X}'\}$ by changing the distribution of covariates such that $P(T|\mathbf{X}') = P(T)$, namely $T \perp \mathbf{X}'$. (ii) "calibration" distribution approximation: We develop a Generative Adversarial Network to learn a sample weight $\mathbf{w}$ on the observed data $\mathbf{D}_{obs}$ such that the distribution of weighted observed data would be similar even identical with the "calibration" data $\mathbf{D}_{cal}$, formally $\mathbf{w}P(T, \mathbf{X}) = P(T, \mathbf{X}')$. Finally, the learned sample weight $\mathbf{w}$ can guarantee precise estimation on the causal effect of continuous treatment, since it ensures the treatment become independent with the covariates on the weighted observed data, achieving de-confounding between treatment and covariates.

4.1.1 "calibration" distribution generation

In this component, our goal is to generate an "calibration" distribution $\mathbf{D}_{cal} = \{T, \mathbf{X}'\}$, where the treatment T is independent with the covariates $\mathbf{X}'$, ensuring there is no confounding between treatment and covariates.

Proposition 1 *By randomly shuffling the value of each covariate $\mathbf{X}_{\cdot,i}$ over all samples in observed data $\mathbf{D}_{obs} = \{T, \mathbf{X}\}$, the shuffled covariates would become independent with the treatment T if sample size $n \to \infty$.*

The random shuffling process refers to randomly permuting the elements in each covariate $\mathbf{X}_{\cdot,i} \in \mathbb{R}^{n\times 1}$. If $n \to \infty$, the shuffled covariates, denoted as $\mathbf{X}'$, should be independently random variables. Hence, the treatment variable T would be independent with the shuffled covariates $\mathbf{X}'$.

Therefore, we can obtain an "calibration" data $\mathbf{D}_{cal} = \{T, \mathbf{X}'\}$ under Proposition 1, where the confounding bias between the treatment T and covariates $\mathbf{X}'$ are fully removed.

Need to note that the "calibration" data is meaningless except for its non-confounding or independence property between its treatment and covariates. Many other methods can also be employed for generating an "calibration" data, we leave it in future work.

4.1.2 "calibration" distribution approximation

In this component, we aim to adjust the distribution of observed data $\mathbf{D}_{obs} = \{T, \mathbf{X}\}$ by sample weighting such that with the identical distribution of the "calibration" data $\mathbf{D}_{cal} = \{T, \mathbf{X}'\}$, resulting in the treatment become independent with the covariates in the adjusted observed data.

Inspired by the immense success of Generative Adversarial Network (GAN) (Goodfellow et al., 2014) in producing simulated data that highly resembles the distribution of real-world samples, we propose a novel framework that leverages the objective of GAN to the task of

generating weight for ensuring the distribution of adjusted observed data has the identical distribution of the "calibration" one[2].

To be self-contained, we briefly revisit the key idea of GAN (Goodfellow et al., 2014). The goal of GAN is to learn a generative model $g(\cdot)$ of an unknown distribution $\mathcal{D}_{data}$ using a class of discriminators $d(\cdot)$ to gauge the similarity between data distributions. The GAN framework can be described as a game between the generator $g(\cdot)$ and the discriminator $d(\cdot)$, where the generator $g(\cdot)$ simulates data $g(z)$ with an input random variable z from a predefined distribution $\mathcal{D}_z$, then the discriminator $d(\cdot)$ attempts to bridge the distribution between the simulated data $g(z)$ and real samples s in $\mathcal{D}_{data}$ by minimizing the expected classification error in the real and simulated samples as:

$$L(g, d) = \mathbb{E}_{s\sim\mathcal{D}_{data}}[l(d(s), 1)] + \mathbb{E}_{z\sim\mathcal{D}_z}[l(d(g(z), 0)], \tag{5}$$

where $l(\cdot)$ is the loss function. Given the discriminator model $d(\cdot)$, the generator $g(\cdot)$ attempts to maximize the expected error with following objective function to find:

$$g^\star = \arg\max_g (\min_d L(g, d)). \tag{6}$$

In our problem, we employ the generator $g(\cdot)$ to optimize a sample weight vector $\mathbf{w} = (w_1, w_2, \cdots, w_n)$ to adjust the distribution of observed data $\mathbf{D}_{obs} = \{T, \mathbf{X}\}$ such that the discriminator $d(\cdot)$ cannot distinguish the adjusted observed distribution and the "calibration" distribution by minimizing the expected classification error in the adjusted observed and "calibration" samples as:

$$\begin{aligned} L(\mathbf{w}, d) = &\mathbb{E}_{(t,x)\sim\mathbf{D}_{cal}}[l(d(t, x), 1)] \\ &+ \mathbb{E}_{(t,x)\sim\mathbf{D}_{obs}}[w_{(t,x)} \cdot l(d(t, x), 0)], \\ &s.t. \quad \mathbb{E}_{(t,x)\sim\mathbf{D}_{obs}}[w_{(t,x)}] = 1, \mathbf{w} \succeq 0, \end{aligned} \tag{7}$$

where $w_{(t,x)}$ refers to the sample weight related to the sample (t, x) in the observed data, and $l(\cdot)$ is the loss function. The term $\mathbb{E}_{(t,x)\sim\mathbf{D}_{obs}}[w_{(t,x)}] = 1$ avoids all sample weight to be $zero$, and $\mathbf{w} \succeq 0$ constrains each sample weight to be non-negative. Given the discriminator model $d(\cdot)$, the generator $g(\cdot)$ attempts to maximize the expected error with following objective function to find:

$$\hat{\mathbf{w}} = \arg\max_{\mathbf{w}} (\min_d L(\mathbf{w}, d)). \tag{8}$$

Following the objective function in Eq. (7) we know only the term $\mathbb{E}_{(t,x)\sim\mathbf{D}_{obs}}[w_{(t,x)} \cdot l(d(t, x), 0)]$ is related to the parameter $\mathbf{w}$. Then to optimize $\mathbf{w}$ with discriminator $d(\cdot)$ fixed, we could either maximize $\mathbb{E}_{(t,x)\sim\mathbf{D}_{obs}}[w_{(t,x)} \cdot l(d(t, x), 0)]$ with gradient ascending methods, or instead choose to minimize $\mathbb{E}_{(t,x)\sim\mathbf{D}_{obs}}[-w_{(t,x)} \cdot l(d(t, x), 0)]$ or $\mathbb{E}_{(t,x)\sim\mathbf{D}_{obs}}[w_{(t,x)} \cdot l(d(t, x), 1)]$ with gradient descending methods as metioned in Goodfellow et al. (2014). In practice, we switch 0/1 labels for two data distributions, resulting the following loss functions for both $\mathbf{w}$ and discriminator $d(\cdot)$ to minimize alternately:

$$L_d(\mathbf{w}, d) = L(\mathbf{w}, d)$$

2. Other methods could also be applied for generating sample weights with a "calibaration" distribution, we leave comparison among these methods for future work.

$$
\begin{aligned}
L_w(\mathbf{w}, d) = & \mathbb{E}_{(t,x)\sim\mathbf{D}_{cal}}[l(d(t,x),0)] \\
& + \mathbb{E}_{(t,x)\sim\mathbf{D}_{obs}}[w_{(t,x)} \cdot l(d(t,x),1)], \\
s.t. \quad & \mathbb{E}_{(t,x)\sim\mathbf{D}_{obs}}[w_{(t,x)}] = 1, \mathbf{w} \succeq 0,
\end{aligned} \tag{9}
$$

The details of our GAD algorithm is summarized in Algorithm 1 in Appendix A.

Finally, with the optimized sample weight $\mathbf{w}$ by our GAD algorithm, we can estimate the ADRF with Eq. (3) and MTEF with Eq. (4).

5. Experiment

In this section, we evaluate the effectiveness of our proposed method on both synthetic and real-world datasets.

5.1 Baseline Methods

We implement or use the following baseline methods for comparison. Parameter settings are as default, unless otherwise specified.

- *Inverse Probability Weighting (IPW)* (Robins et al., 2000): This method estimates conditional probability $P(T_i|\mathbf{X_i})$ by regressing treatment T on covariates $\mathbf{X}$, then uses it to generate sample weight. Both unstablized ($IPW_{unstable} = \frac{1}{P(T_i|\mathbf{X}_i)}$) and stablized ($IPW_{stable} = \frac{P(T_i)}{P(T_i|\mathbf{X}_i)}$) versions are evaluated. Performance of IPW largely relies on estimation of $P(T_i|\mathbf{X}_i)$. Thus, it's not attractive in most real-world applications.

- *Inverse Second-Moment Weighting (ISMW)* (Galagate, 2016): This method is an extension of IPW with second-moment. Under linear assumption of Y-T relation, $ISMW$ generates sample weight matrix in closed form as $\mathbb{E}(B_i B_i^T|\mathbf{X}_i)^{-1}$, where $B_i = [1, t_i]^T$. However, if Y-T relation is more complex, $ISMW$ might be less attractive due to its restriction to the means of higher-order terms.

- *Covariate-Balancing Generalized Propensity Score (CBGPS)* (Fong et al., 2018): Based on Generalized Propensity Score, this method adapts covariate balancing condition for continuous treatment that $\mathbb{E}(P(T_i|\mathbf{X}_i)T_i\mathbf{X}_i) = \mathbb{E}(T_i)\mathbb{E}(\mathbf{X}_i) = 0$, where $\mathbf{X}$ and T are centralized and orthogonalized in preprocessing.

5.2 Evaluation Metrics

In synthetic experiments, we evaluate the performance based on three metrics:

- Bias(MTEF): mean absolute error of MTEF estimation over all samples
- RMSE(MTEF): rooted mean squared error of MTEF estimation over all samples
- RMSE(ADRF): rooted mean squared error of ADRF estimation over all samples

Normally, MTEF-based metrics are more important than ADRF-based in synthetic experiments, as it eliminates effect of intercept which involves means of covariates and noise.

5.3 Experiments on Synthetic Data

In this section, we introduce data generation process for synthetic datasets, and demonstrate the effectiveness of our proposed weighting method, with extensive experiments.

5.3.1 Dataset

The process of generating synthetic datasets basically follows Fong et al. (2018) with slight modification, where we set sample size $n = 2000$ and the dimension of observed variables $p = 10$. We first generate covariates $\mathbf{X} = (x_1, x_2, \ldots, x_p)$ independently with **Standard Normal** distribution as:

$$x_1, x_2, \ldots, x_p \overset{i.i.d}{\sim} N(0, 1)$$

Then we generate treatment T and outcome Y generally as:

$$T = f(\mathbf{X}) + \epsilon_t, \quad Y = g(\mathbf{X}) + \mu(T) + \epsilon_y$$

where $f(\mathbf{X}) = \sum_{j=1}^{p} \alpha_{mod(j,10)} \cdot x_j$, $g(\mathbf{X}) = \sum_{j=1}^{p} \beta_{mod(j,10)} \cdot x_j$, $\alpha = [1, 1, 0.2, 0.2, 0.2, 0, 0, 0, 0, 0]$, and $\epsilon_t \sim N(0, 2)$. Function $mod(a, b)$ returns the modulus after division of a by b. β, $\mu(T)$ and ϵ_y varies under different settings with considering the relation (linear and non-linear) between Y and T, and between Y and $\mathbf{X}$:

YT-linear:

$$\mu(T) = T \quad and \quad \epsilon_y \sim N(0, 5)$$

YT-nonlinear:

$$\mu(T) = T^2 + T, \ \epsilon_y \sim N(0, 9) \quad and \quad g(\mathbf{X}) = 2g(\mathbf{X})$$

YX-linear:

$$\beta = [0, 1, 0, 0.1, 0.1, 0.1, 0, 0, 0, 0]$$

YX-nonlinear:

$$\beta = [0, 2, 0, 0.5, 0.5, 0.5, 0, 0, 0, 0] \quad and$$

$$x_j = I(mod(j, 10) = 1) \cdot x_j^2 + I(mod(j, 10) \neq 1) \cdot x_j$$

By combining different YX relations and YT relations, we could evaluate all methods under 4 different settings which cover a large variety of common cases. In simulation, we know the ground-truth ADRF and MTEF as:

YT-linear:

$$ADRF(T) = T + \mathbb{E}(g(X)) \quad and \quad MTEF = 1$$

YT-nonlinear:

$$ADRF(T) = T^2 + T + 2\mathbb{E}(g(X)) \quad and \quad MTEF = 2\mathbb{E}(T) + 1$$

Then, we evaluate the ADRF and MTEF with our algorithm, comparing with baselines.

Setting	Method	$n = 2000, p = 10$		
		BIAS_{MTEF}	RMSE_{MTEF}	RMSE_{ADRF}
YX-linear, YT-linear	OLS	0.153(0.044)	0.153(0.044)	0.392(0.102)
	$\text{IPW}_{unstable}$	0.141(0.082)	0.141(0.082)	0.467(0.184)
	IPW_{stable}	0.070(0.081)	0.070(0.081)	0.239(0.194)
	ISMW	0.049(0.026)	0.049(0.026)	**0.163(0.071)**
	CBGPS	0.063(0.069)	0.063(0.069)	0.223(0.171)
	Our	**0.043(0.036)**	**0.043(0.036)**	0.176(0.082)
YX-linear, YT-nonlinear	OLS	0.310(0.078)	0.332(0.079)	0.816(0.181)
	$\text{IPW}_{unstable}$	0.286(0.126)	0.337(0.150)	0.885(0.430)
	IPW_{stable}	0.211(0.137)	0.252(0.159)	0.589(0.362)
	ISMW	1.023(0.521)	1.050(0.520)	2.569(1.310)
	CBGPS	0.195(0.126)	0.237(0.152)	0.558(0.334)
	Our	**0.167(0.072)**	**0.207(0.091)**	**0.471(0.144)**
YX-nonlinear, YT-linear	OLS	0.353(0.066)	0.353(0.066)	0.882(0.159)
	$\text{IPW}_{unstable}$	0.179(0.089)	0.179(0.089)	0.582(0.177))
	IPW_{stable}	0.100(0.070)	0.100(0.070)	0.295(0.174)
	ISMW	**0.050(0.027)**	**0.050(0.027)**	**0.206(0.087)**
	CBGPS	0.096(0.067)	0.096(0.067)	0.294(0.138)
	Our	0.068(0.032)	0.068(0.032)	0.291(0.179)
YX-nonlinear, YT-nonlinear	OLS	0.753(0.120)	0.871(0.145)	1.982(0.320)
	$\text{IPW}_{unstable}$	0.360(0.123)	0.418(0.133)	1.071(0.398)
	IPW_{stable}	0.280(0.144)	0.338(0.181)	0.738(0.372)
	ISMW	1.689(0.817)	1.777(0.798)	4.335(2.002)
	CBGPS	0.267(0.106)	0.317(0.127)	**0.714(0.271)**
	Our	**0.230(0.147)**	**0.282(0.185)**	1.067(0.347)

Table 2: Results on synthetic datasets with sample size $n = 2000$, feature dimension $p = 10$. The value in bracket refers to corresponding standard deviations of 10 times experiments. The smaller of these metrics, the better.

5.3.2 Results

To evaluate the performance of our proposed algorithm on continuous treatment effect estimation, we carry out experiments for 10 times independently for each setting. Based on the estimated ADRF and MTEF, we report Bias(MTEF), RMSE(MTEF) and RMSE(ADRF), and their standard deviation (SD) over 10 times experiments in Tables 2. From these results, we have following observations and analysis:

- Model based regression method, OLS, cannot precisely estimate the causal effect of continuous treatment even the model is correctly specified, since it ignores the confounding bias between treatment and covariates.

- With constraints on the variance of weight, IPW_{stable} achieves better performance than $IPW_{unstable}$ across most settings. Moreover, with considering the second moments, $ISMW$ obtains the best performance among IPW based methods under setting with YT-linear. However, in the setting with YT-nonlinear, the performance of $ISMW$ is very poor and even worse than OLS, since it entirely relies on the linear assumption between T and Y.

- By directly minimizing the association between treatment and covariates, CBGPS obtains a good performance across all settings. However, it's still worse than our method, since it only considers the linear association between treatment and covariates.
- Our algorithm, by directly making treatment become independent with covariates, achieves significant improvements over the baselines in different settings, especially on MTEF-based metrics. Under setting YT-nonlinear where the assumptions in baselines are violated, our GAD algorithm, a non-parametric method, almost obtains the best performance. Under setting YT-linear, our algorithm can also achieve comparable results with the best baseline.

5.4 Real-world Data: *TWINS*

Considering that few real-world datasets with continuous treatment contain ground-truth of causal effect. Like previous most work on continuous treatment (Kallus and Zhou, 2018), we perform a semi-simulation on *TWINS*, a dataset previously used in binary or categorical treatment research for evaluation.

5.4.1 Dataset

TWINS is a dataset commonly used in binary treatment research, which totally contains data of over 70,000 twins. The treatment of this dataset is to be the light one or not when born. Originally, the treatment is generated from a continuous variable, *born weight*. The dataset also includes 50 covariates recording information of parents, which are almost the same for a pair of twins.

To conduct semi-simulation on *TWINS* dataset, we first filter dataset by limiting weight under 2 kilogram. Data of 4,821 pairs of twins are left for further experiments. We set the difference between *born weight* with 2 kilogram as treatment T in our experiment. To ensuring the ground-truth, we propose to semi-simulate the outcome variable Y from treatment and covariates to represent the risk of death after born. We reorganize a few columns of covariates according to twins identity, such as *birth order*. Also, we concatenate original binary treatment to covariates. From observation of dataset, as weight difference increases, death rate over dataset population also increases. Thus, we can generate outcome as follows with different settings of Y-T relations:

YT-linear:

$$Y = 4 \cdot T - 40 + \mathbf{X}\gamma + \epsilon$$

YT-nonlinear:

$$Y = 0.15 \cdot T^2 + T - 20 + \mathbf{X}\gamma + \epsilon$$

where $\gamma \in \mathbb{R}^{p\times 1}$ and $\gamma_i \sim N(0, 0.25)$, $\epsilon \sim N(0, 2.25)$. Then we can get the ground-truth ADRF and MTEF as **YT-linear:**

$$ADRF(T) = 4T - 40 + \mathbb{E}(\mathbf{X}_{i,\cdot}\gamma) \;\; and \;\; MTEF = 4$$

YT-nonlinear:

$$Y_{ADRF} = 0.15 \cdot T^2 + T - 20 + \mathbb{E}(\mathbf{X}_{i,\cdot}\gamma) \;\; and \;\; MTEF = 0.3 \cdot T + 1$$

Setting	Method	*TWINS*		
		BIAS_{MTEF}	RMSE_{MTEF}	RMSE_{ADRF}
YT-linear	OLS	0.125(0.082)	0.125(0.082)	0.569(0.371)
	$\text{IPW}_{unstable}$	0.480(0.424)	0.480(0.424)	3.844(2.120)
	IPW_{stable}	0.818(0.360)	0.818(0.360)	4.899(1.450)
	ISMW	**0.007(0.005)**	**0.007(0.005)**	0.299(0.188)
	CBGPS	0.043(0.040)	0.043(0.040)	0.620(0.378)
	Our	0.049(0.048)	0.049(0.048)	**0.283(0.183)**
YT-nonlinear	OLS	0.208(0.079)	0.236(0.089)	0.686(0.350)
	$\text{IPW}_{unstable}$	1.385(0.757)	1.532(0.890)	5.506(2.061)
	IPW_{stable}	1.693(1.599)	1.878(1.849)	6.982(4.453)
	ISMW	0.165(0.062)	0.181(0.069)	0.962(0.214)
	CBGPS	0.187(0.137)	0.216(0.158)	0.683(0.380)
	Our	**0.127(0.039)**	**0.144(0.046)**	**0.383(0.091)**

Table 3: Results on TWINS dataset.

5.4.2 Results

We report the results in Table 3. Though we can only carry out semi-simulation on real dataset, the hidden T-X relation is still a major challenge to tackle for methods based on generalized propensity score or other methods requiring a T-model. Thus, $IPW_{unstable}$ and IPW_{stable} fail on causal effect estimation on continuous treatment due to possibly misspecified T-model and inaccurate estimation on generalized propensity score as demonstrated in Table 3. Similar to the results on synthetic data, under the setting with YT-linear, *ISMW* achieves the best performance among baselines since its assumptions are satisfied. By directly make treatment become independent with covariates, our method achieves comparable result with *ISMW*, and significantly better than other methods. Under the setting with YT-nonlinear, our algorithm obtains the best performance with a significant improvement than baselines, since our method is non-parametric and can guarantee the de-confounding between treatment and covariates.

6. Conclusion

In this paper, we focus on the problem of causal effect estimation on continuous treatment in observational studies. We argue that traditional methods for continuous treatment effect estimation are basically regression model based, hence, their performance entirely relies on the correctly specified models or some impractical assumptions. Hence, we proposed a non-parametric method, Generative Adversarial De-confoudning (GAD) algorithm to remove the confounding bias between treatment and covariates for precisely estimation on continuous treatment effect. In our GAD algorithm, we proposed a Generative Adversarial Network based de-confounding algorithm to generative sample weight for making treatment become independent with covariates. We proved that the learned sample weight from our GAD algorithm can fully remove the confounding bias from empirical experiments. The experimental results on both synthetic and real world datasets show that our GAD algorithm outperforms the baselines for causal effect estimation on continuous treatment in observational studies.

Acknowledgments

We thank all the reviewers for constructive feedbacks that helped improve the paper. This research was supported in part by the Fundamental Research Funds for the Central Universities; National Key Research and Development Program of China No. 2018AAA0101900.

Appendix A. Our proposed Generative Adversarial De-confounding (GAD) algorithm

In GAD algorithm, steps 1-4 is for generating the "calibration" distribution $\mathbf{D}_{cal} = \{T, \mathbf{X}'\}$, and steps 5-11 is for approximating the "calibration" distribution by learning sample weight.

Algorithm 1 Generative Adversarial De-confounding

Input: Observed Data $\mathbf{D}_{obs} = \{T, \mathbf{X}\}$, stopping criterion $h(\mathbf{D}_{obs}, \mathbf{D}_{target}, \mathbf{w})$, optimizer for discriminator, $SGD(\theta, L_d(\mathbf{w}, d))$, and optimizer for $\mathbf{w}$, $Ranger(\mathbf{w}, L_w(\mathbf{w}, d))$

Output: sample weight $\mathbf{w}$

1: **for** $i = 1, 2, \cdots, p$ **do**
2: Generating shuffled covariate $\mathbf{X}'_{\cdot,i}$ by randomly permuting the elements in $\mathbf{X}_{\cdot,i}$
3: **end for**
4: Generate target data $\mathbf{D}_{cal} = \{T, \mathbf{X}'\}$
5: Initialize sample weight $\mathbf{w}^0 = [1, 1, \cdots, 1]$
6: Initialize discrimator $d(\cdot)$ with parameter θ^0
7: Initialize the iteration variable $t \leftarrow 0$
8: **repeat**
9: $t \leftarrow t + 1$
10: Update $\theta^t \leftarrow SGD(\theta^{t-1}, L_d(\mathbf{w}^{t-1}, d))$
11: Update sample weight $\mathbf{w}^t \leftarrow Ranger(\mathbf{w}^{t-1}, L_w(\mathbf{w}^{\mathbf{t-1}}, d))$
12: Limit mean of sample weight $\mathbf{w}_i^t \leftarrow n\mathbf{w}_i^t / \sum_{i=1}^n \mathbf{w}_i^t, \quad i = 1, 2, \cdots, n$
13: **until** $h(\mathbf{D}_{obs}, \mathbf{D}_{cal}, \mathbf{w}^t)$ satisfied or max iteration is reached
14: **return** sample weight $\mathbf{w}$

Appendix B: Implementation of Baselines and Our Algorithm on Synthetic Data.

We implement both versions of *IPW*, and *ISMW* as baseline methods. As for both versions of *CBGPS*, we use the R-Package 'CBPS' to carry out experiments on both synthetic and real-world datasets. All baseline methods calculate weight first, then use *Weighted Least Square* (WLS) to estmiate ADRF and MTEF by regressing Y on T.

The core part of *IPW* and *ISMW* is to estimate $P(T_i)$, $P(T_i|\mathbf{X}_i)$ and $\mathbb{E}(B_i B_i^T|\mathbf{X}_i)$. For $P(T_i)$, we estimate a **Normal** distribution with sample mean and variance of T as its parameter. For $P(T_i|\mathbf{X}_i)$, we estimate a **Normal** distribution per sample. To generate its parameter, we first fit a T-model $t(\mathbf{X}_i)$ by linearly regressing T on $\mathbf{X}$. Then we take $\hat{T}_i = t(\mathbf{X}_i)$ as its mean, $\frac{1}{N-p}\sum_{i=1}^N (T_i - \hat{T}_i^2)$ as its variance, where p is degree of freedom. For $\mathbb{E}(B_i B_i^T|\mathbf{X}_i)$, as it only involves $\mathbb{E}(T_i|\mathbf{X}_i)$ and $Var(T_i|\mathbf{X}_i)$, we take the fitted value of T-model as $\mathbb{E}(T_i|\mathbf{X}_i)$ and estimated residual variance as $Var(T_i|\mathbf{X}_i)$.

Parameters of *CBGPS* mostly remain default in all experiments, except that iteration of *CBGPS* sets to 10000.

Our method uses a neural network with two hidden layers as discriminator, each layer has 512 hidden units. Mish is used as activation function, and dropout layer with keep

probability = 0.5 is applied to last hidden layer. We use SGD with learning rate $lr = 1e^{-3}$ as optimizer of discriminator, Ranger (a combination of RAdam and Look-Ahead) with learing rate $lr = 3e^{-4}$, betas = $(0.0, 0.9)$, internal step $k = 5$ as optimizer of sample weight. We use *cross-entropy* as loss function, and optimize both sample weight and discriminator almost the same way as GAN does. Optimization is performed on full-sample rather than mini-batch, considering global feature of sample weight. We simply divide sample weight by their sum after each step, to keep restriction on sum of weight.

Appendix C: Implementation of Baselines and Our Algorithm on TWINS Data.

As on *TWINS* the conducted semi-simulation is similar to synthetic experiments, Implementation details are almost the same as those in synthetic experiments except for a few changes. Iteration of *CBGPS* is set to 20000 rather than 10000. For our method, we adjust number of hidden units from 512 to 256 with fixed number of hidden layers. Z-score standardization is applied to covariates as data preprocessing. To achieve better performance, we keep shuffling covariates along every dimension during training, rather than one-time shuffling used in experiments on synthetic datasets.

References

Susan Athey, Guido W. Imbens, and Stefan Wager. Approximate residual balancing: debiased inference of average treatment effects in high dimensions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(4):597–623, 2018.

Peter C. Austin. An introduction to propensity score methods for reducing the effects of confounding in observational studies. *Multivariate behavioral research*, 46(3):399–424, 2011.

Heejung Bang and James M. Robins. Doubly robust estimation in missing data and causal inference models. *Biometrics*, 61(4):962–973, 2005.

David Chan, Rong Ge, Ori Gershony, Tim Hesterberg, and Diane Lambert. Evaluating online ad campaigns in a pipeline: causal models at scale. In *KDD*, pages 7–16, 2010.

Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, et al. Double machine learning for treatment and causal parameters. *arXiv preprint arXiv:1608.00060*, 2016.

Christian Fong, Chad Hazlett, Kosuke Imai, et al. Covariate balancing propensity score for a continuous treatment: Application to the efficacy of political advertisements. *The Annals of Applied Statistics*, 12(1):156–177, 2018.

Douglas Galagate. *CAUSAL INFERENCE WITH A CONTINUOUS TREATMENT AND OUTCOME: ALTERNATIVE ESTIMATORS FOR PARAMETRIC DOSE-RESPONSE FUNCTIONS WITH APPLICATIONS.* PhD thesis, 2016.

Antonio F. Galvao and Liang Wang. Uniformly semiparametric efficient estimation of treatment effects with a continuous treatment. *Journal of the American Statistical Association*, 110(512):1528–1542, 2015.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

Jens Hainmueller. Entropy balancing for causal effects: A multivariate reweighting method to produce balanced samples in observational studies. *Political Analysis*, 20(1):25–46, 2012.

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

Keisuke Hirano and Guido W. Imbens. The propensity score with continuous treatments. *Applied Bayesian modeling and causal inference from incomplete-data perspectives*, 226164:73–84, 2004.

Paul W. Holland. Statistics and causal inference. *Journal of the American statistical Association*, 81(396):945–960, 1986.

Kosuke Imai and Marc Ratkovic. Covariate balancing propensity score. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 76(1):243–263, 2014.

Kosuke Imai and David A. Van Dyk. Causal inference with general treatment regimes: Generalizing the propensity score. *Journal of the American Statistical Association*, 99 (467):854–866, 2004.

Guido W. Imbens. Nonparametric estimation of average treatment effects under exogeneity: A review. *Review of Economics and statistics*, 86(1):4–29, 2004.

Guido W. Imbens and Donald B. Rubin. *Causal inference in statistics, social, and biomedical sciences*. Cambridge University Press, 2015.

Nathan Kallus. Generalized optimal matching methods for causal inference. *The Journal of Machine Learning Research (forthcoming)*, 2019.

Nathan Kallus and Angela Zhou. Policy evaluation and optimization with continuous treatments. In *International Conference on Artificial Intelligence and Statistics*, pages 1243–1251, 2018.

Edward H. Kennedy, Zongming Ma, Matthew D. McHugh, and Dylan S. Small. Nonparametric methods for doubly robust estimation of continuous treatment effects. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(4):1229–1245, 2017.

Ron Kohavi and Roger Longbotham. Unexpected results in online controlled experiments. *ACM SIGKDD Explorations Newsletter*, 12(2):31–35, 2011.

Noémi Kreif, Richard Grieve, Iván Díaz, and David Harrison. Evaluation of the effect of a continuous treatment: a machine learning approach with an application to treatment for traumatic brain injury. *Health economics*, 24(9):1213–1228, 2015.

Kun Kuang, Pen Cui, Bo Li, Meng Jiang, Shiqiang Yang, and Fei Wang. Treatment effect estimation with data-driven variable decomposition. In *AAAI*, 2017a.

Kun Kuang, Peng Cui, Bo Li, Meng Jiang, and Shiqiang Yang. Estimating treatment effect in the wild via differentiated confounder balancing. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 265–274, 2017b.

Kun Kuang, Pen Cui, Bo Li, Meng Jiang, Yashen Wang, Fei Wu, and Shiqiang Yang. Treatment effect estimation via differentiated confounder balancing and regression. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 14:1 – 25, 2019.

Kun Kuang, Peng Cui, Hao Zou, Bo Li, Jianrong Tao, Fei Wu, and Shiqiang Yang. Data-driven variable decomposition for treatment effect estimation. *IEEE Transactions on Knowledge and Data Engineering*, (01):1–1, 2020a. ISSN 1558-2191.

Kun Kuang, Lian Li, Zhi Geng, Lei Xu, Kun Zhang, Beishui Liao, Huaxin Huang, Peng Ding, Wang Miao, and Zhichao Jiang. Causal inference. *Engineering*, 6(3):253 – 263, 01 2020b. ISSN 2095-8099.

Yameng Liu, Aw Dieng, Sudeepa Roy, Cynthia Rudin, and Alexander Volfovsky. Interpretable almost matching exactly for causal inference. *AISTATS*, 2019.

Daniel F. McCaffrey, Greg Ridgeway, and Andrew R. Morral. Propensity score estimation with boosted regression for evaluating causal effects in observational studies. *Psychological methods*, 9(4):403, 2004.

Romain Neugebauer and Mark van der Laan. Nonparametric causal effects based on marginal structural models. *Journal of Statistical Planning and Inference*, 137(2):419–434, 2007.

Judea Pearl. *Causality*. Cambridge university press, 2009.

James M. Robins, Miguel Angel Hernan, and Babette Brumback. Marginal structural models and causal inference in epidemiology, 2000.

JM. Robins and A. Rotnitzky. Comment on inference for semiparametric models: Some questions and an answer, by pj bickel and j. kwon. *Statistica Sinica*, 11:920–936, 2001.

Paul R. Rosenbaum and Donald B. Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 1983.

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

Daniel Westreich, Justin Lessler, and Michele Jonsson Funk. Propensity score estimation: neural networks, support vector machines, decision trees (cart), and meta-classifiers as alternatives to logistic regression. *Journal of clinical epidemiology*, 63(8):826–833, 2010.