Treatment Effect Estimation from Small Observational Data

by

Shonosuke Harada

A Doctor Thesis

Kyoto University

Abstract

Recent years have witnessed a significant advancement in data-driven approaches utilizing machine learning techniques, as a result of the increasing diversity of data. These sophisticated approaches have been extensively utilized in a variety of applications such as computer vision, natural language processing, healthcare, economic policy, and education. Despite the remarkable success and significant impact achieved by advanced machine learning methods in the real world, their effective deployment in decision-making remains a challenge in some scenarios. In this regard, the estimation of treatment effects, which seeks to quantify the impact of a particular intervention, holds paramount significance in a wide range of domains for effective decision making.

In this thesis, we discuss treatment effect estimation problems specifically in terms of three key challenges: data scarcity, observational bias, and hidden confounding variables. The first challenge relates to the scarcity of available data, as collecting extensive observational data is often impeded by the associated costs of time, monetary investment, and human effort. The second challenge pertains to the presence of observational bias, which arises due to the decision-maker's policy in assigning treatments based on confounding variables. Besides, we can only observe one realization of possible treatments, namely, the counterfactual nature in treatment effect estimation. These factors result in biased observational data and make naive estimands unreliable. The third challenge is the presence of hidden confounding variables. It is necessary to have all the confounding variables for the accurate treatment effect estimation; however, it may not be feasible to obtain all of them in observational data due to considerations of confidentiality or the expenses incurred in data collection.

To overcome these challenges, we designed several approaches based on advanced machine learning techniques. The first approach involved the utilization of unlabeled data, which are comparatively more attainable, by integrating two innovative concepts from causal inference and semi-supervised learning: matching and label propagation, to leverage the abundance of unlabeled data that is readily accessible. The second strategy involves leveraging a wealth of auxiliary information in the form of graph-structured data associated with treatments. While conventional studies tend to focus on binary or multiple-choice treatments, our approach capitalizes on this auxiliary information in tackling the complex issue of treatment effect estimation, particularly when the number of treatments is substantial. In our third approach, we address the challenge of treatment effect estimation in the presence of hidden confounding variables. Modern advanced generative models such as the Variational Autencoders (VAE) have enabled us to effectively deal with treatment effect estimation problem with hidden confounding variabels. However, a naive application of VAEs may lead to suboptimal performances due to the nature of their loss function. Our analysis demonstrates that this phenomenon also manifests in the context of treatment effect estimation and can give unsatisfactory results. To mitigate this, we drew upon recent theoretical insights on VAEs and introduced an innovative matching approach that incorporates hidden confounding variables. Furthermore, through extensive experiments on synthetic, semi-synthetic, and real-world datasets, we validated the effectiveness of these three approaches.

Acknowledgements

I would like to express my sincere gratitude to the following people who have provided great support throughout my studies. I am truly grateful to my supervisor, Professor Hisashi Kashima for his insightful and invaluable advice. I am greatly thankful to Professor Hidetoshi Shimodaira and Professor Tatsuya Akutsu for their incisive and constructive remarks. I also would like to thank the members of Kashima-Yamada Laboratory for perceptive feedbacks through daily discussions and conversations. Lastly, I would like to express my profound gratitude to my family and friends for their encouragement and heartening support.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

With the exponential advancement of technology and systems, various types of data and information have been stored in numerous fields. Data-driven decision making has emerged as a predominant method for tackling complex problems in real-world scenarios, thereby leading to flourishing of machine learning techniques. In recent years, predictive models incorporating sophisticated machine learning algorithms have seen a considerable increase in development and application across numerous fields, including computer vision [1, 2], natural language processing [3, 4], chemoinformatics, [5, 6, 7] recommendation [8, 9] amongst others. The main goal of constructing such predictive models is to support people in making complex decisions.

However, a simple supervised learning framework that merely predicts a target outcome for an input may not be appropriate for all scenarios. For instance, in the field of healthcare efficacy evaluation, the objective is to assess the impact of drugs on patients. Predicting outcomes when drugs are prescribed to patients is not an effective solution, as recovered patients may have recovered without the drugs, hence administering drugs in such a case is not an efficient decision. Similarly, in the field of advertising, the aim is to predict the increase in consumer purchasing propensity resulting from an advertisement. Similar to the field of healthcare, advertisements are given to those consumers who would have bought the products regardless of

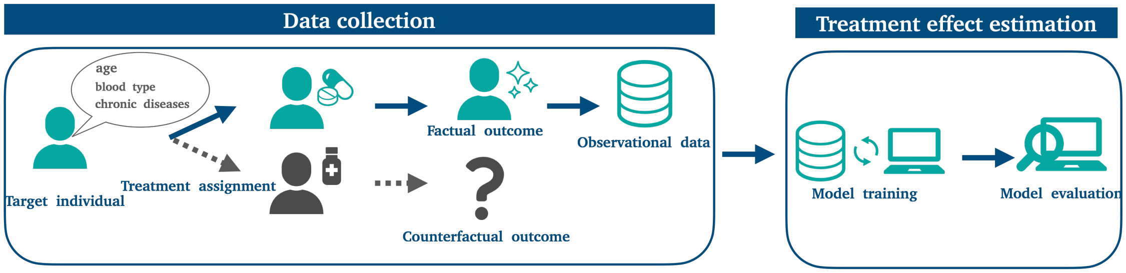


Figure 1.1: An illustrative pipeline for treatment effect estimation from observational data. Treatment is assigned by a decision maker using the target individual's covariates such as age, blood type, and chronic diseases. Then we only observe only one realization (the upper path of data collection procedure). These mechanisms make observational data skewed.

the advertisements, which may not a be feasible action as advertising for such consumers is not effective. Therefore, to make effective decisions in such situations, it is crucial to comprehend the difference in outcomes resulting from changes in the controllable variable, i.e., the treatment effect. Treatment effect estimation has drawn a substantial number of researches for a long time across a wide range of fields. Following the potential outcome framework, known as the Neyman-Rubin potential outcomes framework proposed by Neyman and Rubin [10, 11], a large number of sophisticated methods, including classic matching-based methods [12, 13], tree-based methods [14, 15, 16], have been proposed, and more recently, neural networks [17, 18, 19, 20] have also been applied to treatment effect estimation. Besides the potential outcome framework, a large number of studies also formulated treatment effect based on the Structural Causal Models (SCMs) in the framework proposed by Pearl [21, 22], which also provide tools for modeling treatment effects through potential outcomes, and are quite useful in expressing complex causal relationships between random variables. The SCMs have been successfully employed in causal analyses involving unobserved variables [23, 24, 25, 26]. While sophisticated methods have been developed, several frequently appearing challenges still exist and are difficult to mitigate in practice.

We describe an illustrative pipeline of treatment effect estimation from observational data in Figure 1.1. In addition, we briefly summarize the difficulties in estimating treatment effect from observational data in the real world.

Data scarcity: In practical situations, it is frequently necessary to address treatment effect estimation problems with limited data resources. For instance, the evaluation of the efficacy of medical treatments in the healthcare domain often necessitates the procurement of extensive data through costly experimentation. This not only incurs a significant cost in terms of money and time, but also requires us to work on treatment effect estimation with data scarcity. Furthermore, this scarcity not only poses a critical issue in and of itself, but exacerbates other underlying difficulties as well.

Observational bias: Observational bias is a bias caused by a decision maker or some mechanisms that assign treatment to individuals. This bias is often referred to as selection bias. Owing to the counterfactual nature of treatment effects, where we have access to only the outcomes of one treatment and cannot observe the outcomes of other treatments, observational bias results in data that is susceptible to being skewed. For instance, in the healthcare domain, elderly individuals are more likely to receive medication than younger individuals, thereby leading to an age-based bias in treatment assignment that is non-random. Therefore, naively learning predictive models from skewed observational data cannot produce accurate treatment effects, and consequently necessitating the use of techniques that mitigate observational bias. Additionally, the space of treatments can be larger than binary and consists of multiple or continuous treatments, further complicating the task of bias mitigation.

Hidden confounding variables: Confounding variables are variables that affect both on treatment assignment and outcomes. Most studies often assume the presence of these variables in observational data; however, this assumption is not always met. Some covariates, such as daily diet or income, may be unobtainable due to factors such as cost and privacy considerations. These confounding variables are referred to as "hidden confounding variables", as they are not visible and included in observational data. However, it is not possible to accurately estimate the treatment effect without taking these variables into account. Therefore, inferring hidden confounding variables is essential for accurate treatment effect estimation.

Next, we outline several current and prevalent approaches to address the aforementioned challenges. A wide range of machine learning methods have been devised to tackle the issue of data scarcity. A promising solution in overcoming this challenge is the utilization of unlabeled data, which is often formulated as semi-supervised learning. This approach is particularly useful in situations where obtaining labeled data is cost-prohibitive, while unlabeled data can still be obtained with relative ease. In addition, by taking treatment effect estimation problem as semi-supervised learning problem, it is expected that a robust predictive model can be learned even with limited labeled data.

Secondly, a great deal of effort has also been devoted to mitigate observational bias. The classical matching approach is a widely utilized solution, which compares pairs of individuals with similar covariates who received different treatments and impute counterfactual outcomes [12, 27]. Besides the matching-oriented strategies, recent studies have also demonstrated the efficacy of deep learning-based methods in mitigating bias [18, 17]. The key idea of these deep learning methods is to balance representations between treatment and control group by minimizing the discrepancy of these two groups representations.

Thirdly, the estimation of treatment effects in the presence of hidden confounding variables has garnered substantial attention among researchers [25, 24, 28, 26]. Recently, Variational Autoencoder (VAE) [29] was introduced to the field of treatment effect estimation [24, 28]. The VAE framework enables the inference of hidden confounding variables as latent variables and the estimation of outcomes based on these latent variables.

Based on the outlines regarding several possible solutions to these difficulties,

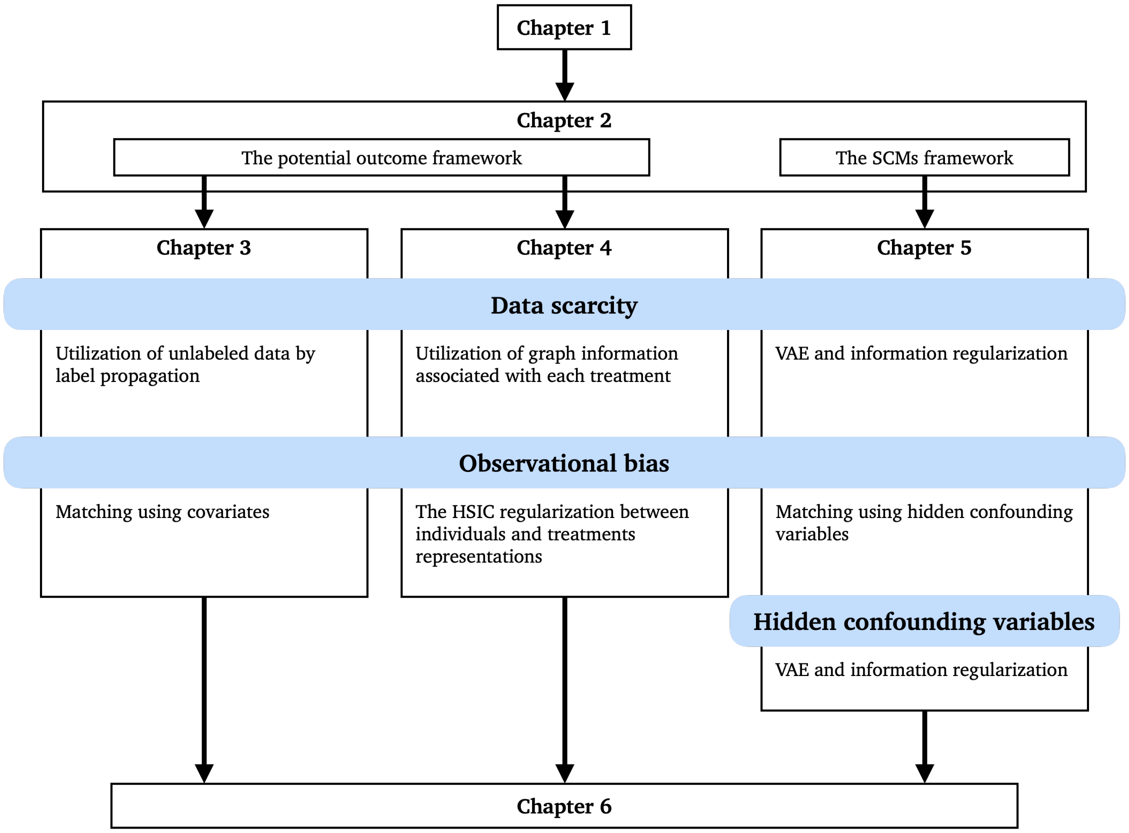

Chapter 1
Chapter 2
The potential outcome framework
The SCMs framework
Chapter 3
Chapter 4
Chapter 5
Data scarcity
Utilization of unlabeled data by label propagation
Utilization of graph information associated with each treatment
VAE and information regularization
Observational bias
Matching using covariates
The HSIC regularization between individuals and treatments representations
Matching using hidden confounding variables
Hidden confounding variables
VAE and information regularization
Chapter 6

Figure 1.2: Dependence structure of the chapters in this thesis. We also briefly state how the three key difficulties are addressed in the each chapter.

we present brief summaries of the remaining chapters below. Dependence structure of chapters in this thesis is described in Figure 1.2. We colorized three challenges tackled in this thesis and outlined the solutions to these challenges.

In Chapter 2, mathematical definitions based on the potential outcome framework [11, 30] and the SCMs framework [21, 22] for treatment effect estimation are detailed. As our approaches are formulated based on either one of them depending on the convenience in formulation, we provide an introduction of several terminologies used in this context.

In Chapter 3, we discuss a method that uses not only labeled data but also unlabeled data. Label scarcity has been one of the vital problems in numerous industries and extensively studied in the field of machine learning [31, 32]. One of the most promising strategies to incorporate unlabeled data is label propagation [33]. Label propagation defines a nearest-neighbor graph that includes unlabeled data based on individual similarity and propagates labels through the graph. Therefore, labels on unlabeled data can be efficiently predicted by their individual neighbours. In addition, the matching method in treatment effect estimation also seeks to find similar individuals to propagate counterfactual outcomes. We combine two ideas from causal inference and semi-supervised learning, namely, matching and label propagation, respectively, to propose *counterfactual propagation,* which is the first semi-supervised treatment effect estimation method. Using a motivating example, we also demonstrate how useful the proposed method is.

In Chapter 4, we examine a method that addresses graph-structured treatments. As previously noted, outcome estimation of treatments for individual targets is a critical aspect of decision-making that is based on causal relationships. While prior studies have primarily considered binary or multiple-choice treatments, some applications feature an extensive array of treatments that are rich in information. In this chapter, we focus on an important class of such cases, namely, the outcome estimation problem of graph-structured treatments such as pharmaceuticals. Given the extensive number of possible treatments and the counterfactual nature of the problem, determining treatment effects from observational data presents a considerable challenge. Our proposed method *GraphITE* extracts the representations of the graph-structured treatments using graph neural networks, and also mitigates the observation biases by using HSIC regularization. By HSIC regularization, we aim to increase the independence of the representations of the targets and the treatments. Owing to its capability of incorporating graph structured treatments, the proposed method also enables us to handle "zero-shot" treatments that are not included in observational data.

In Chapter 5, we address a strategy for estimating treatment effects in the presence of hidden, confounding variables. Despite the assumption that observational data encompasses all confounding variables, it is impracticable to ensure their exhaustive inclusion. Furthermore, due to their associated costs, certain confounding variables may pose challenges in terms of their obtainment. Recently, VAE-based methods have been successfully applied to treatment effect estimation problems. However, a major limitation of VAE-based methods is the lack of theoretical guarantees, as a recent analysis has shown that an optimal solution of VAE may not yield a correct generative function for a particular dataset class [34]. Hence, we opine that this phenomenon may lead to undesirable results if employed naively in this problem. Therefore, we propose an efficient VAE-based method that employs information theory in estimating treatment effect while combining it with a matching technique. To the best of our knowledge, this is the first work that gives the correct treatment effect given an optimal solution using VAE-based methods.

Finally, in Chapter 6, we conclude this thesis and discuss current limitations of the proposed methods. We also suggest promising future directions with regard to treatment effect estimation from small observational data.

Chapter 2

Preliminaries

In this chapter, we present preliminaries with regard to treatment effect. There exist primarily two frameworks for treatment effect estimation: (i) the potential outcome framework [11, 35, 36, 37] and the Structural Causal Models (SCMs) framework [38, 21, 22]. Further, we provide several mathematical definition employed in this context for the both frameworks. Throughout this chapter, let $\mathbf{x}$ denote the covariates including individual features, $t \in \{0, 1\}$ indicate the treatment.

2.1 The potential outcome framework

The potential outcome framework, also known as the Neyman-Rubin causal model, is a framework in which the causal relationship between treatment and an outcome variables is evaluated based on the comparison of potential outcomes under treatment and control conditions. Let y^1, y^0 denote the potential outcomes, which are outcomes under each treatment, of an individual under each treatment. In this framework, Individual Treatment Effect (ITE) is defined as:

$$\mathrm{ITE} := y^1 - y^0. \tag{2.1}$$

Based on several assumptions, we can identify the treatment effect of a group, namely, Average Treatment Effect (ATE) defined as:

$$\text{ATE} := \mathbb{E}[y^1 - y^0]. \tag{2.2}$$

In some applications, particularly in the field of healthcare, the heterogeneity of treatment effects across various subgroups conditioned on covariates such as age, gender, blood type, and health condition $\mathbf{x}$ is a crucial concern and main interest. The heterogeneous treatment effect, often called as Conditional ATE (CATE), is defined as:

$$\text{CATE}(\mathbf{x}) := \mathbb{E}[y^1 - y^0 \mid \mathbf{x}]. \tag{2.3}$$

The estimation of CATE plays an indispensable role in determining and providing the best actions for each individual, and thus making personalized decision making feasible.

2.2 The Structural Causal Models (SCMs) framework

In this chapter, we particularly refer to the SCMs introduced by Pearl [38, 21]. The SCMs refer to framework used to represent and evaluate complex causal relationships among both observed and unobserved variables. For example, structural equation for variable A as a cause of B is expressed as:

$$B := f(A), \tag{2.4}$$

where f is a deterministic function. The SCMs are also employed in modeling potential outcomes. Let y denote the outcomes of an individual. In the SCMs framework proposed by Pearl, the intervention is denoted using $do-$operator. A $do-$operator is utilized to distinguish the conditional distribution, for example, $P(y \mid t = 1)$ and $P(y \mid do(t = 1))$. The former, the probability distribution $P(y \mid t = 1)$ represents the population distribution of y among individuals whose t values are 1. However, the latter $P(y \mid do(t = 1))$ represents the population distribution of y when all the individuals have their t values are fixed at 1. Using $do-$operator, ATE is defined as:

$$\text{ATE} := \mathbb{E}[y \mid do(t = 1)] - \mathbb{E}[y \mid do(t = 0)]. \tag{2.5}$$

Similar to the potential outcome framework, by letting $P(y \mid \mathbf{x}, do(t = 1))$ denote the conditional probability given $\mathbf{x}$ in the distribution by intervention $do(t = 1)$, CATE is defined as:

$$\text{CATE}(\mathbf{x}) := \mathbb{E}[y \mid \mathbf{x}, do(t = 1)] - \mathbb{E}[y \mid \mathbf{x}, do(t = 0)]. \tag{2.6}$$

If unobserved confounding variables, namely, hidden confounding variables, exist, we need to infer them from observational data. We describe two examples of graphical causal models in Figure 2.1. In Figure 2.1(a), we can naively give potential outcomes given $\mathbf{x}$ and t as:

$$p(y \mid \mathbf{x}, do(t)) = p(y \mid \mathbf{x}, t). \tag{2.7}$$

However, in Figure 2.1(b),

$$p(y \mid \mathbf{x}, do(t)) \neq p(y \mid \mathbf{x}, t) \tag{2.8}$$

because $\mathbf{x}$ does not have a direct effect on y, and $\mathbf{z}$ is not used for prediction even though it has a direct effect on y. To mitigate this problem, we need conditioning

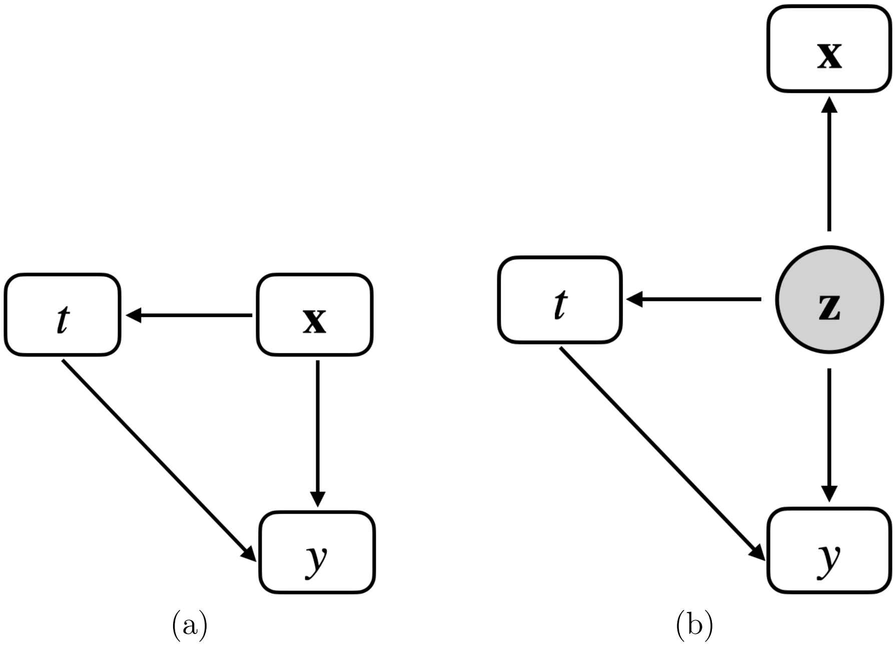


Figure 2.1: Graphical models. While (a) represents a graphical model in which variable $\mathbf{x}$ includes all the confounding variables, (b) represents a graphical model in which variable $\mathbf{x}$ does not include confounding variables and unobserved variable $\mathbf{z}$ includes all the confounding variables.

on hidden confounding variables $\mathbf{z}$ as:

$$p(y \mid \mathbf{z}, do(t)) = p(y \mid \mathbf{z}, t). \quad (\because \text{The definition of Pearl's } do\text{-calculus.}) \tag{2.9}$$

Consequently, if $\mathbf{z}$ is recovered from $\mathbf{x}$, y can be predicted as:

$$\begin{align} p(y \mid \mathbf{x}, do(t)) &= \int_z p(y \mid \mathbf{z}, do(t)) p(\mathbf{z} \mid \mathbf{x}) d\mathbf{z} \tag{2.10} \\ &= \int_z p(y \mid \mathbf{z}, t) p(\mathbf{z} \mid \mathbf{x}) d\mathbf{z}. \tag{2.11} \end{align}$$

In Chapter 3 and 4, we formulate problems following the potential outcome framework. In contrast, the SCMs facilitate the expression of complex causal relationships between variables, and are useful particularly in the formulation of treatment effect estimates in the presence of unobserved confounding variables. Hence, we formulate a problem following the SCMs framework in Chapter 5.

Chapter 3

Counterfactual Propagation for Semi-Supervised Individual Treatment Effect Estimation

3.1 Introduction

One of the important roles of predictive modeling is to support decision making related to taking particular actions in responses to situations. The recent advances of in the machine learning technologies have significantly improved their predictive performance. However, most predictive models are based on passive observations and do not aim to predict the causal effects of actions that actively intervene in environments. For example, advertisement companies are interested not only in their customers' behavior when an advertisement is presented, but also in the causal effect of the advertisement, in other words, the change it causes on their behavior. There has been a growing interest in moving from this passive predictive modeling to more active causal modeling in various domains, such as education [15], advertisement [39, 13], economic policy [40], and health care [41].

Taking an action toward a situation generally depends on the expected improvement in the outcome due to the action.

This is often called the *individual treatment effect (ITE)* [11] and is defined as the difference between the outcome of taking the action and that of *not* taking the action. An intrinsic difficulty in ITE estimation is that ITE is defined as the difference between the factual and counterfactual outcomes [42, 21, 11]; in other words, the outcome that we can actually observe is either of the one when we take an action or the one when we do not, and it is physically impossible to observe both. To address the counterfactual predictive modeling from observational data, various techniques including matching [12], inverse-propensity weighting [43], instrumental variable methods [44], and more modern deep learning-based approaches have been developed [18, 17]. For example, in the matching method, matching pairs of instances with similar covariate values and different treatment assignments are determined. The key idea is to consider the two instances in a matching pair as the counterfactual instance of each other so that we can estimate the ITE by comparing the pair.

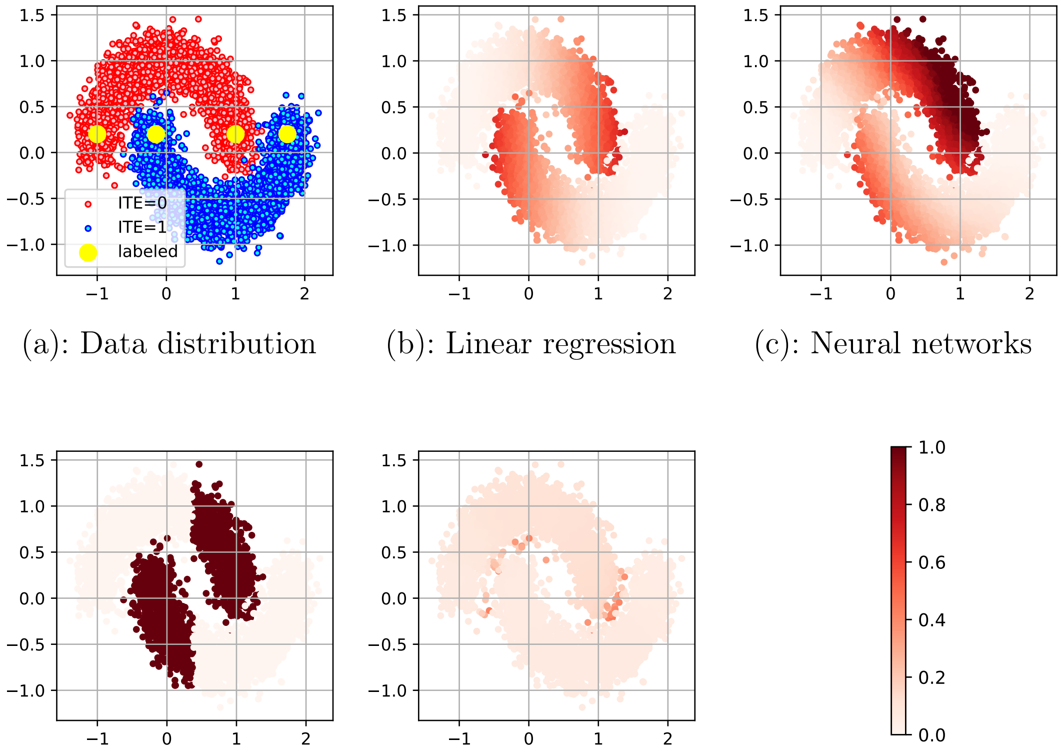
Another difficulty in ITE estimation is data scarcity. For ITE estimation, we need some labeled instances whose treatments (i.e., whether or not an action was taken on the instance) and their outcomes (depending on the treatments) as well as their covariates are given. However, collecting such labeled instances can be quite costly in terms of time and money, or owing to other reasons, such as physical and ethical constraints [45, 46]. Consequently, ITE estimation from scarcely labeled data is an essential requirement in many situations.

In the ordinary predictive modeling problem, a promising option to the scarcity of labeled data is semi-supervised learning that exploits unlabeled instances only with covariates because it is relatively easy to obtain such unlabeled data. A typical solution is the graph-based label propagation method [33, 47, 48], which makes predictions for unlabeled instances based on the assumption that instances with similar covariate values are likely to have a same label.

In this study, we consider a semi-supervised ITE estimation problem. The proposed solution called *counterfactual propagation* is based on the resemblance between the matching method in causal inference and the graph-based semi-supervised learning method called label propagation. We consider a weighted graph over both labeled instances with treatment outcomes and unlabeled instances with no outcomes, and estimate ITEs using the smoothness assumption of the outcomes and the ITEs.

The proposed idea is illustrated in Fig. 3.1. Fig. 3.1(a) describes the two-moon shaped data distribution. We consider a binary treatment and binary outcomes. The blue points indicate the instances with a positive ITE (= 1), where the outcome is 1 if the treatment is 1 and 0 if the treatment is 0. The red points indicate the instances with zero ITE (= 0); their outcomes are always 1 irrespective of the treatments. We have only four labeled data instances shown as yellow points, whose observed (treatment, outcome) pairs are $(0,1),(1,1),(1,1),(0,0)$ from left to right. Since the amount of labeled data is considerably limited, supervised methods relying only on labeled data fail to estimate the ITEs. Figures 3.1(b), (c), (d) show the ITE estimation errors by the standard two-model approach using different base learners, which show poor performance. In contrast, the proposed approach exploits unlabeled data to find connections between the red points and those between the blue points to estimate the correct ITEs (Fig. 3.1(e)).

We propose an efficient learning algorithm assuming the use of a neural network as the base model, and conduct experiments using semi-synthetic real-world datasets to demonstrate that the proposed method estimates the ITEs more accurately than baselines when the labeled instances are limited.



(a): Data distribution (b): Linear regression (c): Neural networks

(d): kNN (e): **Proposed method**

Figure 3.1: Illustrative example using (a) two-moon dataset. Each moons has a constant ITE either of 0 and 1. Only two labeled instances are available for each moon, denoted by yellow points, whose observed $(\text{treatment}, \text{outcome})$ pairs are $(0,1),(1,1),(1,1),(0,0)$ from left to right. Figures (b), (c), and (d) show the ITE estimation error (PEHE) by the standard two-model approach using different base models suffered from the lack of labeled data. The deeper-depth color indicates larger errors. The proposed semi-supervised method (e) successfully exploits the unlabeled data to estimate the correct ITEs.

3.2 Semi-supervised ITE estimation problem

We start with the problem setting of the semi-supervised treatment effect estimation problem. Suppose we have N labeled instances and M unlabeled instances. (We usually assume $N \ll M$.) The set of labeled instances is denoted by $\{(\mathbf{x}_i, t_i, y_i^{t_i})\}_{i=1}^{N}$, where $x_i \in \mathbb{R}^D$ is the covariates of the i-th instance, $t_i \in \{0,1\}$ is the treatment applied to instance i, and $y_i^{t_i}$ is its outcome. Note that for each instance i, either $t_i = 0$ or $t_i = 1$ is realized; accordingly, either y_i^0 or y_i^1 is available. The unobserved outcome is called a counterfactual outcome. The set of unlabeled instances is denoted by $\{(\mathbf{x}_i)\}_{i=N+1}^{N+M}$, where only the covariates are available.

Our goal is to estimate the ITE for each instance. Following the Rubin-Neyman potential outcomes framework [11, 10], the ITE for instance i is defined as $\tau_i = y_i^1 - y_i^0$ exploiting both the labeled and unlabeled sets. Note that τ_i is not known even for the labeled instances, and we want to estimate the ITEs for both the labeled and unlabeled instances.

We make typical assumptions in ITE estimation in this study. i.e., (i) stable unit treatment value: the outcome of each instance is not affected by the treatment assigned to other instances; (ii) unconfoundedness: the treatment assignment to an instance is independent of the outcome given covariates (confounder variables); (iii) overlap: each instance has a positive probability of treatment assignment.

3.3 Proposed method

We propose a novel ITE estimation method that utilizes both the labeled and unlabeled instances. The proposed solution called *counterfactual propagation* is based on the resemblance between the matching method in causal inference and the graph-based semi-supervised learning method.

3.3.1 Matching

Matching is a popular solution to address the counterfactual outcome problem. Its key idea is to consider two similar instances as the counterfactual instance of each other so that we can estimate the causal effect by comparing the pair. More concretely, we define the similarity w_{ij} between two instances i and j, as that defined between their covariates; for example, we can use the Gaussian kernel.

$$w_{ij} = \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{\sigma^2}\right). \tag{3.1}$$

The set of (i, j) pairs with w_{ij} being larger than a threshold and satisfying $t_i \neq t_j$ are found and compared as counterfactual pairs. Note that owing to definition of the matching pair, the matching method only uses labeled data.

3.3.2 Graph-based semi-supervised learning

Graph-based semi-supervised learning methods assume that the nearby instances in a graph are likely to have similar outputs. For a labeled dataset $\{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$ and an unlabeled dataset $\{\mathbf{x}_i\}_{i=N+1}^{N+M}$, their loss functions for standard predictive modeling typically look like

$$L(f) = \sum_{i=1}^{N} l(y_i, f(\mathbf{x}_i)) + \lambda \sum_{i,j=1}^{N+M} w_{ij} \left(f(\mathbf{x}_i) - f(\mathbf{x}_j)\right)^2, \tag{3.2}$$

where f is a prediction model, l is a loss function for the labeled instances, and λ is a hyper-parameter. The second term imposes “smoothness" of the model output over the input space characterized by w_{ij} that can be considered as the weighted adjacency matrix of a weighted graph; it can be seen the same as that used for matching (3.1).

The early examples of graph-based methods include label propagation [33] and manifold regularization [47]. More recently, deep neural networks have been used as the base model f [48].

3.3.3 Treatment effect estimation using neural networks

We build our ITE estimation model based on the recent advances of deep-learning approaches for ITE estimation, specifically, the treatment-agnostic representation network (TARNet) [18] that is a simple but quite effective model. TARNet shares common parameters for both treatment instances and control instances to construct representations but employs different parameters in its prediction layer, which is given as:

$$f(\mathbf{x}_i, t_i) = \begin{cases} \Theta_1^\top g\left(\Theta^\top x_i\right) & (t_i = 1) \\ \Theta_0^\top g\left(\Theta^\top x_i\right) & (t_i = 0) \end{cases}, \tag{3.3}$$

where Θ is the parameters in the representation learning layer and Θ_1, Θ_0 are those in the prediction layers for treatment and controlled instances, respectively. The g is a non-linear function such as ReLU. One of the advantages of TARNet is that joint representations learning and separate prediction functions for both treatments enable more flexible modeling.

3.3.4 Counterfactual propagation

It is evident that the matching method relies only on labeled data, while the graph-based semi-supervised learning method does not address ITE estimation; however, they are quite similar because they both use instance similarity to interpolate the factual/counterfactual outcomes or model predictions as mentioned in Section 3.3.2. Our idea is to combine the two methods to propagate the outcomes and ITEs over the matching graph assuming that similar instances would have similar outcomes.

Our objective function consists of three terms, $L_\mathrm{s}, L_\mathrm{o}, L_\mathrm{e}$, given as

$$L(f) = L_\mathrm{s}(f) + \lambda_\mathrm{o} L_\mathrm{o}(f) + \lambda_\mathrm{e} L_\mathrm{e}(f), \tag{3.4}$$

where λ_o and λ_e are the regularization hyper-parameters. We employ TARNet [18] as the outcome prediction model $f(\mathbf{x}, t)$. The first term in the objective function (3.4) is a standard loss function for supervised outcome estimation; we specifically employ the squared loss function as

$$L_s(f) = \sum_{i=1}^{N} (y_i^{t_i} - f(\mathbf{x}_i, t_i))^2. \tag{3.5}$$

Note that it relies only on the observed outcomes of the treatments that are observed in the data denoted by t_i.

The second term L_o is the outcome propagation term:

$$L_\mathrm{o}(f) = \sum_t \sum_{i,j=1}^{N+M} w_{ij}((f(\mathbf{x}_i, t) - f(\mathbf{x}_j, t))^2. \tag{3.6}$$

Similar to the regularization term (3.2) in the graph-based semi-supervised learning, this term encourages the model to output similar outcomes for similar instances by penalizing the difference between their outcomes. This regularization term allows the model to propagate outcomes over a matching graph. If two nearby instances have different treatments, they interpolate the counterfactual outcome of each other, which compares the factual and (interpolated) counterfactual outcomes to estimate the ITE. The key assumption behind this term is the smoothness of outcomes for each treatment over the covariate space. While w_{ij} indicates the adjacency between nodes i and j in the graph-based regularization, it can be considered as a matching between the instances i and j in the treatment effect estimation problem. Even though traditional matching methods have only rely on labeled instances, we combine matching with graph-based regularization which also utilizes unlabeled instances. This regularization enables us to propagate the outcomes for each treatment over the matching graph and mitigate the counterfactual problem.

The third term L_e is the ITE propagation term defined as

$$L_\mathrm{e}(f) = \sum_{i,j=1}^{N+M} w_{ij}(\hat{\tau}_i - \hat{\tau}_j)^2, \tag{3.7}$$

where $\hat{\tau}_i$ is the ITE estimate for instance i:

$$\hat{\tau}_i = f(\mathbf{x}_i, 1) - f(\mathbf{x}_i, 0).$$

$L_\mathrm{e}(f)$ imposes the smoothness of the ITE values in addition to that of the outcomes imposed by outcome propagation (3.6). In comparison to the standard supervised learning problems, where the goal is to predict the outcomes, as stated in Section 3.2, our objective is to predict the ITEs. This term encourages the model to output similar ITEs for similar instances. We expect that the outcome propagation and ITE propagation terms are beneficial especially when the available labeled instances are limited while there is an abundance of unlabeled instances, similar to semi-supervised learning.

3.3.5 Estimation algorithm

As mentioned earlier, we assume the use of neural networks as the specific choice of the outcome prediction model f based on the recent successes of deep neural networks in causal inference. For computational efficiency, we apply a sampling approach to optimizing Eq. (3.4). Following the existing method [48], we employ the Adam optimizer [49], which is based on stochastic gradient descent to train the model in a mini-batch manner.

Algorithm 1 describes the procedure of model training, which iterates two steps until convergence. In the first step, we sample a mini-batch consisting of b_1 labeled instances to approximate the supervised loss (3.5). In the second step, we compute the outcome propagation term and the ITE propagation terms using a mini-batch consisting of b_2 instance pairs. Note that in order to make the model more flexible, we can employ different regularization parameters for the treatment outcomes and the control outcomes. The b_1 and b_2 are considered as hyper-parameters; the details are described in Section 3.4. In practice, we optimize only the supervised loss for the first several epochs, and decrease the strength of regularization as training proceeds, in order to guide efficient training [48].

Algorithm 1: Counterfactual propagation

```
Input: labeled instances $\{(\mathbf{x}_i, t_i, y_i^{t_i})\}_{i=1}^{N}$, unlabeled instances $\{(\mathbf{x}_i)\}_{i=N+1}^{N+M}$, a
similarity matrix $w = (w_{ij})$, and mini-batch sizes $b_1, b_2$.
Output: estimated outcome(s) for each treatment $\hat{y}_i^1$ and/or $\hat{y}_i^0$ using
Eq.( 3.3 ).
while not converged do
    # Approximating the supervised loss
    Sample $b_1$ instances $\{(\mathbf{x}_i, t_i, y_i)\}$ from the labeled instances
    Compute the supervised loss (3.5) for the $b_1$ instances
    # Approximating propagation terms
    Sample $b_2$ pairs of instances $\{(\mathbf{x}_i, \mathbf{x}_j)\}$
    Compute the outcome propagation terms $\lambda_{\mathrm{o}} L_{\mathrm{o}}$ for $b_2$ pairs of instances
    Sample $b_2$ pairs of instances $\{(\mathbf{x}_i, \mathbf{x}_j)\}$
    Compute the ITE propagation terms $\lambda_{\mathrm{e}} L_{\mathrm{e}}$ of $b_2$ pairs of instances
    Update the parameters to minimize $L_s + \lambda_o L_o + \lambda_e L_e$ for the sampled
      instances
end
```

3.4 Experiments

We test the effectiveness of the proposed semi-supervised ITE estimation method in comparison with various supervised methods, especially when the available labeled data are strictly limited. We first conduct experiments using two semi-synthetic datasets based on public real datasets. We also design some experiments varying the magnitude of noise on outcomes to explore how the noisy outcomes affect the proposed method. Our implementation is available on Github.[1]

3.4.1 Datasets

Owing to the counterfactual nature of ITE estimation, we rarely access real-world datasets including ground truth ITEs, and therefore cannot directly evaluate ITE estimation methods like the standard supervised learning methods using cross-validation. Therefore, following the existing work [17], we employ two semi-synthetics datasets whose counterfactual outcomes are generated through simulations. Refer

[1] `https://github.com/SH1108/CounterfactualPropagation`

to the original papers for the details on outcome generations [15, 17].

News dataset

is a dataset including opinions of media consumers for news articles [17]. It contains 5,000 news articles and outcomes generated from the NY Times corpus[2]. Each article is consumed on desktop ($t = 0$) or mobile ($t = 1$) and it is assumed that media consumers prefer to read some articles on mobile than desktop. Each article is generated by a topic model and represented in the bag-of-words representation. The size of the vocabulary is 3,477.

IHDP dataset

is a dataset created by randomized experiments called the Infant Health and Development Program (IHDP) [15] to examine the effect of special child care on future test scores. It contains the results of 747 subjects (139 treated subjects and 608 control subjects) with 25 covariates related to infants and their mothers. Following the existing studies [17, 18], the ground-truth counterfactual outcomes are simulated using the NPCI package [50].

3.4.2 Experimental settings

Since we are particularly interested in the situation when the available labeled data are strictly limited, we split the data into a training dataset, validation dataset, and a test dataset by limiting the size of the training data. We change the ratio of the training to investigate the performance; we use $10\%, 5\%$, and 1% of the whole data from the News dataset, and use $40\%, 20\%$, and 10% of those from the IHDP dataset for the training datasets. The rest 80% and 10% of the whole News data are used for test and validation, respectively. Similarly, 50% and 10% of the whole IHDP

[2] `https://archive.ics.uci.edu/ml/datasets/Bag+of+Words`

dataset are used for test and validation, respectively. We report the average results of 10 trials on the News dataset and 50 trials on the IHDP dataset.

In addition to the evaluation under labeled data scarcity, we also test the robustness against label noises. As pointed out in previous studies, noisy labels in training data can severely deteriorate predictive performance, especially in semi-supervised learning. Following the previous work [15, 17], we add the noise $\epsilon \sim \mathcal{N}(0, c^2)$ to the observed outcomes in the training data, where $c \in \{1, 3, 5, 7, 9\}$. In this evaluation, we use 1% of the whole data as the training data for the News dataset and 10% for the IHDP dataset, respectively, since we are mainly interested in label-scarce situations.

The hyper-parameters are tuned based on the prediction loss using the observed outcomes on the validation data. We calculate the similarities between the instances by using the Gaussian kernel; we select σ^2 from $\{5 \times 10^{-3}, 1 \times 10^{-3}, \ldots, 1 \times 10^2, 5 \times 10^2\}$, and select λ_o and λ_e from $\{1 \times 10^{-3}, 1 \times 10^{-2}, \ldots, 1 \times 10^2\}$. Because the scales of treatment outcomes and control outcomes are not always the same, we found scaling the regularization terms according to them is beneficial; specifically, we scale the regularization terms with respect to the treatment outcomes, the control outcomes, and the treatment effects by $\alpha = 1/\sigma^2_{y^1}, \beta = 1/\sigma^2_{y^0}$, and $\gamma = 1/(\sigma^2_{y^1} + \sigma^2_{y^0})$, respectively. We apply principal component analysis to reduce the input dimensions before applying the Gaussian kernel; we select the number of dimensions from $\{2, 4, 6, 8, 16, 32, 64\}$. The learning rate is set to 1×10^{-3} and the mini-batch sizes b_1, b_2 are chosen from $\{4, 8, 16, 32\}$.

As the evaluation metrics, we report the *Precision in Estimation of Heterogeneous Effect (PEHE)* used in the previous research [15]. PEHE is the estimation error of individual treatment effects, and is defined as

$$\epsilon_{\text{PEHE}} = \frac{1}{N+M} \sum_{i=1}^{N+M} (\tau_i - \hat{\tau}_i)^2.$$

Following the previous studies [18, 51], we evaluate the predictive performance for

Table 3.1: The performance comparison of different methods on News dataset. The † indicates that our proposed method (CP) performs statistically significantly better than the baselines by the paired t-test ($p < 0.05$). The bold results indicate the best results in terms of the average.

$\sqrt{\epsilon_{\text{PEHE}}}$	**News 1%**		**News 5%**		**News 10%**	
Method	labeled	unlabeled	labeled	unlabeled	labeled	unlabeled
Ridge-1	$^{\dagger}4.494_{\pm 1.116}$	$^{\dagger}4.304_{\pm 0.988}$	$^{\dagger}4.666_{\pm 1.0578}$	$^{\dagger}3.951_{\pm 0.954}$	$^{\dagger}4.464_{\pm 1.082}$	$^{\dagger}3.607_{\pm 0.943}$
Ridge-2	$2.914_{\pm 0.797}$	$^{\dagger}2.969_{\pm 0.814}$	$^{\dagger}2.519_{\pm 0.586}$	$^{\dagger}2.664_{\pm 0.614}$	$^{\dagger}2.560_{\pm 0.558}$	$^{\dagger}2.862_{\pm 0.621}$
Lasso-1	$^{\dagger}4.464_{\pm 1.082}$	$^{\dagger}3.607_{\pm 0.943}$	$^{\dagger}4.466_{\pm 1.058}$	$^{\dagger}3.367_{\pm 0.985}$	$^{\dagger}4.464_{\pm 1.0822}$	$^{\dagger}3.330_{\pm 0.984}$
Lasso-2	$^{\dagger}3.344_{\pm 1.022}$	$^{\dagger}3.476_{\pm 1.038}$	$^{\dagger}2.568_{\pm 0.714}$	$^{\dagger}2.848_{\pm 0.751}$	$^{\dagger}2.269_{\pm 0.628}$	$^{\dagger}2.616_{\pm 0.663}$
kNN	$^{\dagger}3.678_{\pm 1.250}$	$^{\dagger}3.677_{\pm 1.246}$	$^{\dagger}3.351_{\pm 1.004}$	$^{\dagger}3.434_{\pm 1.018}$	$^{\dagger}3.130_{\pm 0.752}$	$^{\dagger}3.294_{\pm 0.766}$
PSM	$^{\dagger}3.713_{\pm 1.149}$	$^{\dagger}3.662_{\pm 1.127}$	$^{\dagger}3.363_{\pm 0.901}$	$^{\dagger}3.500_{\pm 0.961}$	$^{\dagger}3.260_{\pm 0.734}$	$^{\dagger}3.526_{\pm 0.832}$
RF	$^{\dagger}4.494_{\pm 1.116}$	$^{\dagger}3.691_{\pm 0.878}$	$^{\dagger}4.466_{\pm 1.058}$	$^{\dagger}2.975_{\pm 0.874}$	$^{\dagger}4.464_{\pm 1.082}$	$^{\dagger}2.657_{\pm 0.682}$
CF	$^{\dagger}3.691_{\pm 1.082}$	$^{\dagger}3.607_{\pm 0.943}$	$^{\dagger}3.196_{\pm 0.901}$	$^{\dagger}3.215_{\pm 0.910}$	$^{\dagger}3.101_{\pm 0.806}$	$^{\dagger}3.129_{\pm 0.818}$
TARNET	$^{\dagger}3.166_{\pm 0.742}$	$^{\ddagger}3.160_{\pm 0.722}$	$^{\dagger}2.670_{\pm 0.796}$	$^{\dagger}2.666_{\pm 0.773}$	$^{\dagger}2.589_{\pm 0.894}$	$^{\dagger}2.598_{\pm 0.869}$
CFR	$2.908_{\pm 0.752}$	$2.925_{\pm 0.746}$	$^{\dagger}2.590_{\pm 0.772}$	$^{\dagger}2.546_{\pm 0.796}$	$^{\dagger}2.570_{\pm 0.519}$	$^{\dagger}2.451_{\pm 0.547}$
CP (proposed)	$\mathbf{2.844}_{\pm \mathbf{0.683}}$	$\mathbf{2.823}_{\pm \mathbf{0.656}}$	$\mathbf{2.310}_{\pm \mathbf{0.430}}$	$\mathbf{2.446}_{\pm \mathbf{0.471}}$	$\mathbf{2.003}_{\pm \mathbf{0.393}}$	$\mathbf{2.153}_{\pm \mathbf{0.436}}$

labeled instances and unlabeled instances separately. Note that, although we observe the factual outcomes of the labeled data, their true ITEs are still unknown because we cannot observe their counterfactual outcomes.

3.4.3 Baselines

We compare the proposed method with several existing supervised ITE estimation approaches. (i) Linear regression (Ridge, Lasso) is the ordinary linear regression models with ridge regularization or lasso regularization. We consider two variants: one that includes the treatment as a feature (denoted by 'Ridge-1' and 'Lasso-1'), and the other with two separated models for treatment and control (denoted by 'Ridge-2' and 'Lasso-2'). (ii) k-nearest neighbors (kNN) is a matching-based method that predicts the outcomes using nearby instances. (iii) Propensity score matching with logistic regression (PSM) [43] is a matching-based method using

Table 3.2: The performance comparison of different methods on IHDP dataset. The † indicates that our proposed method (CP) performs statistically significantly better than the baselines by the paired t-test ($p < 0.05$). The bold results indicate the best results in terms of average.

$\sqrt{\epsilon_{\text{PEHE}}}$	**IHDP 10%**		**IHDP 20%**		**IHDP 40%**	
Method	labeled	unlabeled	labeled	unlabeled	labeled	unlabeled
Ridge-1	$^\dagger 5.484_{\pm 8.825}$	$^\dagger 5.696_{\pm 7.328}$	$^\dagger 5.067_{\pm 8.337}$	$^\dagger 4.692_{\pm 6.943}$	$^\dagger 4.80_{\pm 8.022}$	$^\dagger 4.448_{\pm 6.874}$
Ridge-2	$^\dagger 3.426_{\pm 5.692}$	$^\dagger 3.357_{\pm 5.177}$	$^\dagger 2.918_{\pm 4.874}$	$^\dagger 2.918_{\pm 4.730}$	$^\dagger 2.605_{\pm 4.314}$	$^\dagger 2.639_{\pm 4.496}$
Lasso-1	$^\dagger 6.685_{\pm 10.655}$	$^\dagger 6.408_{\pm 9.900}$	$^\dagger 6.435_{\pm 10.147}$	$^\dagger 6.2446_{\pm 9.639}$	$^\dagger 6.338_{\pm 9.704}$	$^\dagger 6.223_{\pm 9.596}$
Lasso-2	$^\dagger 3.118_{\pm 5.204}$	$^\ddagger 3.292_{\pm 5.725}$	$^\dagger 2.684_{\pm 4.428}$	$^\dagger 2.789_{\pm 4.731}$	$^\dagger 2.512_{\pm 4.075}$	$^\dagger 2.571_{\pm 4.379}$
kNN	$^\dagger 4.457_{\pm 6.957}$	$^\dagger 4.603_{\pm 6.629}$	$^\dagger 4.023_{\pm 6.193}$	$^\dagger 4.370_{\pm 6.244}$	$^\dagger 3.623_{\pm 5.316}$	$^\dagger 4.109_{\pm 5.936}$
PSM	$^\dagger 6.506_{\pm 10.077}$	$^\dagger 6.982_{\pm 10.672}$	$^\dagger 6.277_{\pm 9.708}$	$^\dagger 7.209_{\pm 11.077}$	$^\dagger 6.065_{\pm 9.362}$	$^\dagger 7.181_{\pm 9.362}$
RF	$^\dagger 6.924_{\pm 10.620}$	$^\dagger 5.356_{\pm 8.790}$	$^\dagger 6.854_{\pm 10.471}$	$^\dagger 4.845_{\pm 8.241}$	$^\dagger 6.928_{\pm 10.396}$	$^\dagger 4.549_{\pm 7.822}$
CF	$^\dagger 5.389_{\pm 8.736}$	$^\dagger 5.255_{\pm 8.070}$	$^\dagger 4.939_{\pm 7.762}$	$^\dagger 4.955_{\pm 7.503}$	$^\dagger 4.611_{\pm 7.149}$	$^\dagger 4.764_{\pm 7.448}$
TARNET	$^\dagger 3.827_{\pm 5.315}$	$^\dagger 3.664_{\pm 4.888}$	$^\dagger 2.770_{\pm 3.617}$	$^\dagger 2.770_{\pm 3.542}$	$^\dagger 2.005_{\pm 2.447}$	$^\dagger 2.267_{\pm 2.825}$
CFR	$^\dagger 3.461_{\pm 5.1444}$	$^\dagger 3.292_{\pm 4.619}$	$^\dagger 2.381_{\pm 3.126}$	$^\dagger 2.403_{\pm 3.080}$	$1.572_{\pm 1.937}$	$1.815_{\pm 2.204}$
CP (proposed)	$\mathbf{2.427}_{\pm \mathbf{3.189}}$	$\mathbf{2.652}_{\pm \mathbf{3.469}}$	$\mathbf{1.686}_{\pm \mathbf{1.838}}$	$\mathbf{1.961}_{\pm \mathbf{2.343}}$	$\mathbf{1.299}_{\pm \mathbf{1.001}}$	$\mathbf{1.485}_{\pm \mathbf{1.433}}$

the propensity score estimated by a logistic regression model. We also compared the proposed method with tree models such as (iv) random forest (RF) [14] and its causal extension called (v) causal forest (CF) [16]. In CF, trees are trained to predict propensity score and leaves are used to predict treatment effects. (vi) TARNet [18] is a deep neural network model that has shared layers for representation learning and different layers for outcome prediction for treatment and control instances. (vii) Counterfactual regression (CFR) [18] is a state-of-the-art deep neural network model based on balanced representations between treatment and control instances. We use the Wasserstein distance.

3.4.4 Results and discussions

We discuss the performance of the proposed method compared with the baselines by changing the size of labeled datasets, and then investigate the robustness against the label noises.

We first see the experimental results for different sizes of labeled datasets and sensitivity to the choice of the hyper-parameters that control the strength of label propagation. Tables 3.1 and 3.2 show the PEHE values by different methods for the News dataset and the IHDP dataset, respectively. Overall, our proposed method exhibits the best ITE estimation performance for both labeled and unlabeled data in both of the datasets; the advantage is more significant in the News dataset. The News dataset is a relatively high-dimensional dataset represented using a bag of words. The two-model methods such as Ridge-2 and Lasso-2 perform well in spite of their simplicity, and in terms of regularization types, the Lasso-based methods perform relatively better due to the high-dimensional nature of the dataset.

The proposed method also performs the best in the IHDP dataset; however, the performance gain is rather moderate, as shown by the no statistical significance against CFR [18] with the largest 40%-labeled data, which is the most powerful baseline method. The reason for the moderate improvements is probably because of

Table 3.3: Investigation of the contributions by the outcome propagation and the ITE propagation in the proposed method. The upper table shows the results for the News dataset, and the lower for the IHDP dataset. The $\lambda_o = 0$ and $\lambda_e = 0$ indicate the proposed method (CP) with only the ITE propagation and the outcome propagation, respectively, The † indicates that our proposed method (CP) performs statistically significantly better than the baselines by the paired t-test ($p < 0.05$). The bold numbers indicate the best results in terms of the average.

$\sqrt{\epsilon_{\text{PEHE}}}$	**News 1%**		**News 5%**		**News 10%**	
Method	labeled	unlabeled	labeled	unlabeled	labeled	unlabeled
CP ($\lambda_o = 0$)	$\mathbf{2.812}_{\pm\mathbf{651}}$	$\mathbf{2.806}_{\pm\mathbf{0.598}}$	$^{\dagger}2.527_{\pm 0.474}$	$\dagger 2.531_{\pm 0.523}$	$^{\dagger}2.400_{\pm 0.347}$	$^{\dagger}2.410_{\pm 0.450}$
CP ($\lambda_e = 0$)	$2.879_{\pm 0667}$	$2.885_{\pm 0.609}$	$2.351_{\pm 0.450}$	$2.483_{\pm 0.481}$	$\mathbf{1.996}_{\pm\mathbf{0.338}}$	$2.221_{\pm 0.455}$
CP	$2.844_{\pm 0.683}$	$2.823_{\pm 0.656}$	$\mathbf{2.310}_{\pm\mathbf{0.430}}$	$\mathbf{2.446}_{\pm\mathbf{0.471}}$	$2.003_{\pm 0.393}$	$\mathbf{2.153}_{\pm\mathbf{0.436}}$

$\sqrt{\epsilon_{\text{PEHE}}}$	**IHDP 10%**		**IHDP 20%**		**IHDP 40%**	
Method	labeled	unlabeled	labeled	unlabeled	labeled	unlabeled
CP ($\lambda_o = 0$)	$^{\dagger}2.883_{\pm 3.708}$	$^{\dagger}3.004_{\pm 4.071}$	$^{\dagger}1.972_{\pm 1.930}$	$^{\dagger}2.144_{\pm 2.465}$	$1.574_{\pm 1.392}$	$1.674_{\pm 1.874}$
CP ($\lambda_e = 0$)	$2.494_{\pm 3.201}$	$2.698_{\pm 3.461}$	$1.728_{\pm 2.194}$	$1.977_{\pm 2.450}$	$1.344_{\pm 1.383}$	$1.585_{\pm 1.923}$
CP	$\mathbf{2.427}_{\pm\mathbf{3.189}}$	$\mathbf{2.652}_{\pm\mathbf{3.469}}$	$\mathbf{1.686}_{\pm\mathbf{1.838}}$	$\mathbf{1.961}_{\pm\mathbf{2.343}}$	$\mathbf{1.299}_{\pm\mathbf{1.001}}$	$\mathbf{1.485}_{\pm\mathbf{1.433}}$

the difficulty in defining appropriate similarities among instances, because the IHDP dataset has various types of features including continuous variables and discrete variables. The traditional baselines such as Ridge-1, Lasso-1, k-NN matching, and the tree-based models show limited performance; in contrast, the deep learning based methods such as TARNet and CFR demonstrate remarkable performance. Generally, the performance gain by the proposed method is larger on labeled data than on unlabeled data.

Our proposed method has two different propagation terms, the outcome propagation term and the ITE propagation term, as regularizers for semi-supervised learning. Table 3.3 investigates the contributions by the different propagation terms. The proposed method using the both propagation terms (denoted by CP) shows better results than the one only with the ITE propagation denoted by CP ($\lambda_o = 0$); on the other hand, the improvement over the one only with the outcome regularization is marginal. This observation implies the outcome propagation contributes more to the predictive performance than the ITE propagation.

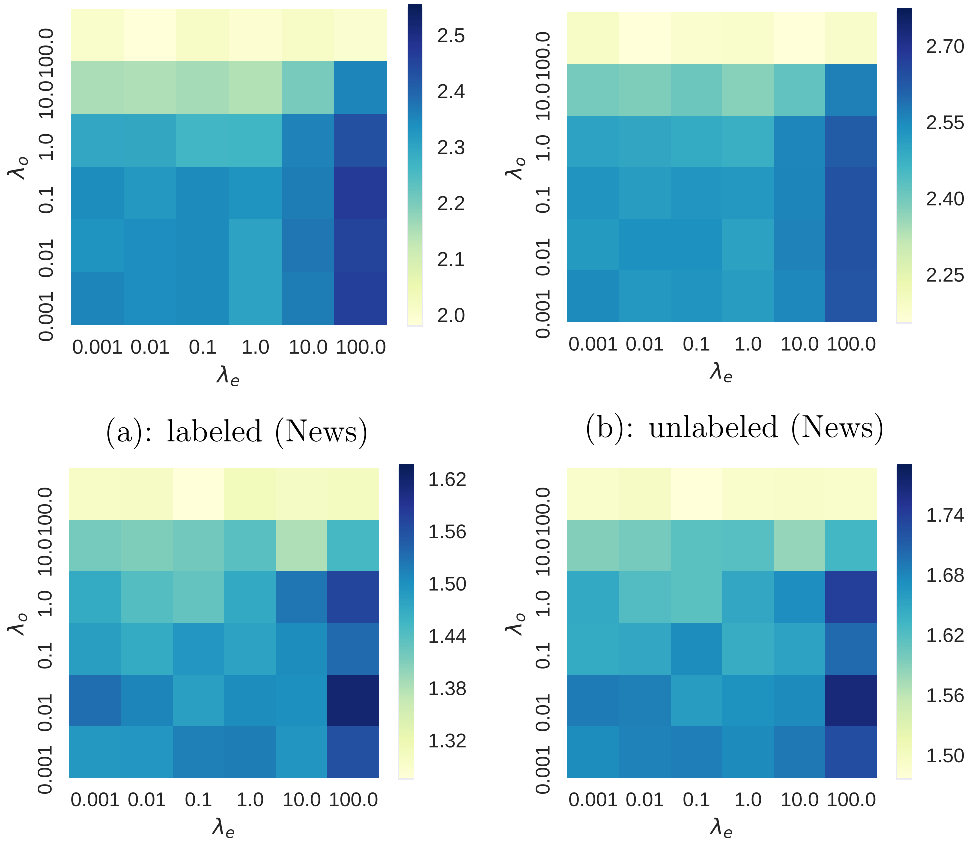
We also examine the sensitivity of the performance to the regularization hyper-parameters. Fig. 3.2 reports the results using 40% and 10% of the whole data as the training data of the News and IHDP datasets, respectively. The proposed method seems rather sensitive to the strength of the regularization terms, particularly on the IHDP dataset, which suggests that the regularization parameters should be carefully tuned using validation datasets in the proposed method. In our experimental observations, slight changes in the hyper-parameters sometimes caused significant changes of predictive performance. We admit the hyper-parameter sensitivity is one of the current limitations in the proposed method and efficient tuning of the hyper-parameters should be addressed in future.

Finally, we compare the proposed method with the state-of-the-art methods by varying the magnitude of noises added to the outcomes. Fig 3.3 shows the performance comparison in terms of $\sqrt{\epsilon_{\mathrm{PEHE}}}$. Note that the results when $c = 1$ correspond to the previous results in Tables 3.1 and 3.2 . The proposed method stays tolerant of relatively small magnitude of noises; however, with larger label noises, it suffers more from wrongly propagated outcome information than the baselines. This is consistent with the previous studies reporting the vulnerability of semi-supervised learning methods against label noises [52, 53, 54, 55].

3.5 Related work

3.5.1 Treatment effect estimation

Treatment effect estimation has been one of the major interests in causal inference and widely studied in various domains. Matching [12, 56] is one of the most basic and commonly used treatment effect estimation techniques. It estimates the counterfactual outcomes using its nearby instances, whose idea is similar to that of



(a): labeled (News)

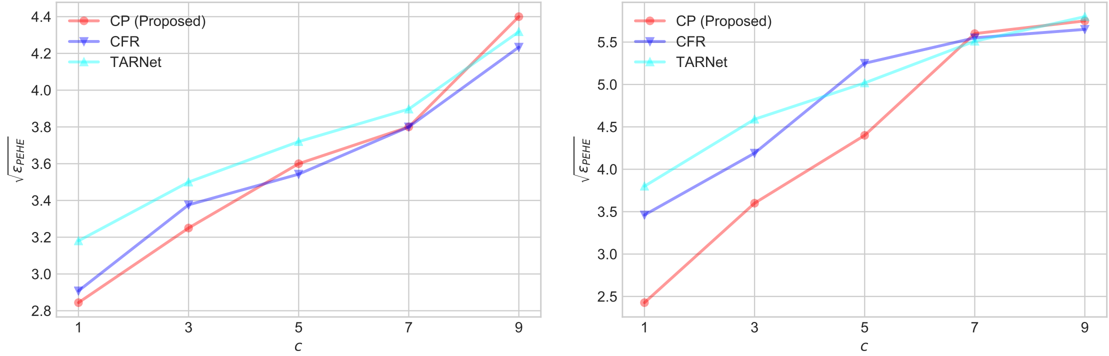
(b): unlabeled (News)

(c): labeled (IHDP)

(d): unlabeled (IHDP)

Figure 3.2: Sensitivity of the results to the hyper-parameters. The colored bars indicate $\sqrt{\epsilon_{\text{PEHE}}}$ for (a)(b) the News dataset and (c)(d) the IHDP dataset when using the largest size of labeled data. The deeper-depth color indicates larger errors. It is observed that the proposed method is somewhat sensitive to the choice of hyper-parameters, especially, the strength of outcome regularizations (λ_o).

graph-based semi-supervised learning. Both methods assume that similar instances in terms of covariates have similar outcomes. To mitigate the curse of dimensionality and selection bias in matching, the propensity matching method relying on the one-dimensional propensity score was proposed [43, 57]. The propensity score is the probability of an instance to get a treatment, which is modeled using probabilistic models like logistic regression, and has been successfully applied in various domains to estimate treatment effects unbiasedly [58]. Tree-based methods such as regression trees and random forests have also been well studied for this problem [59, 16]. One



(a): News

(b): IHDP

Figure 3.3: Performance comparisons for different levels of noise c added to the labels on (a) News dataset and (b) IHDP dataset. Note that the results when $c = 1$ correspond to the previous resuls (Tables 3.1 and 3.2).

of the advantages of such models is that they can build quite expressive and flexible models to estimate treatment effects. Recently, deep learning-based methods have been successfully applied to treatment effect estimation [18, 17]. Balancing neural networks (BNNs) [17] aim to obtain balanced representations of a treatment groups and a control group by minimizing the discrepancy between them, such as the Wasserstein distance [18]. Most recently, some studies have addressed causal inference problems on network-structured data [60, 61, 62]. Alvari et al. applied the idea of manifold regularization using users activities as causality-based features to detect harmful users in social media [61]. Guo et al. considered treatment effect estimation on social networks using graph convolutional balancing neural networks [60]. In contrast with their work assuming the network structures are readily available, we do not assume them and considers matching network defined using covariates.

3.5.2 Semi-supervised learning

Semi-supervised learning, which exploits both labeled and unlabeled data, is one of the most popular approaches, especially in scenarios when only limited labeled data can be accessed [63, 64]. Semi-supervised learning has many variants, and because it is almost impossible to refer to all of them, we mainly review the graph-based regularization methods, known as label propagation or manifold regularization [33, 47, 48]. Utilizing a given graph or a graph constructed based on instance proximity, graph-based regularization encourages the neighbor instances to have similar labels or outcomes [33, 47]. Such idea is also applied to representation learning in deep neural networks [48, 65, 66, 67, 53]; they encourage nearby instances not only to have similar outcomes, but also have similar intermediate representations, which results in remarkable improvements from ordinary methods. One of the major drawbacks of semi-supervised approaches is that label noises in training data can be quite harmful; therefore, a number of studies managed to mitigate the performance degradation [52, 68, 54, 55].

One of the most related work to our present study is graph-based semi-supervised prediction under sampling biases of labeled data [69]. The important difference between this work and ours is that they do not consider intervention and we do not consider the sampling biases of labeled data.

3.6 Conclusion

We addressed the semi-supervised ITE estimation problem. In comparison to the existing ITE estimation methods that only rely on labeled instances including treatment and outcome information, we proposed a novel semi-supervised ITE estimation method that also utilizes unlabeled instances. The proposed method called counterfactual propagation is built on two ideas from causal inference and semi-supervised learning, namely, matching and label propagation, respectively; accordingly, we devised an efficient learning algorithm. Experimental results using the semi-simulated real-world datasets revealed that our methods performed better in comparison to several strong baselines when the available labeled instances are limited. However, this method had issues related to reasonable similarity design and hyper-parameter tuning.

One of the possible future directions is to make use of balancing techniques such as the one used in CFR [18], which can be also naturally integrated into our model. Our future work also includes addressing the biased distribution of labeled instances. As mentioned in Related work, we did not consider such sampling biases for labeled data. Some debiasing techniques [69] might also be successfully integrated into our framework. In addition, robustness against noisy outcomes under semi-supervised learning framework is still the open problem and will be addressed in the future.

Chapter 4

GraphITE: Estimating Individual Effects of Graph-structured Treatments

4.1 Introduction

Estimating causal effects of treatments for individual targets, which is often referred to as individual treatment effect (ITE), is an important foundation for efficient decision making based on observational data. The scope of applications of causal inference ranges across a wide range of fields, including medicine, education, and economic policy. The main difficulties in estimating ITE are (i) the counterfactual nature of observational data, that is, only the outcome of an actual treatment is observed and (ii) biases in observational data due to biases in past treatment decisions. To address these difficulties, various statistical techniques have been developed, including matching [12], inverse propensity score weighting [43], instrumental variable methods [44], as well as more modern representation learning approaches [18, 17].

Most previous studies dealt with a binary or relatively small number of treatments. However, in some scenarios, the number of treatments can be considerably

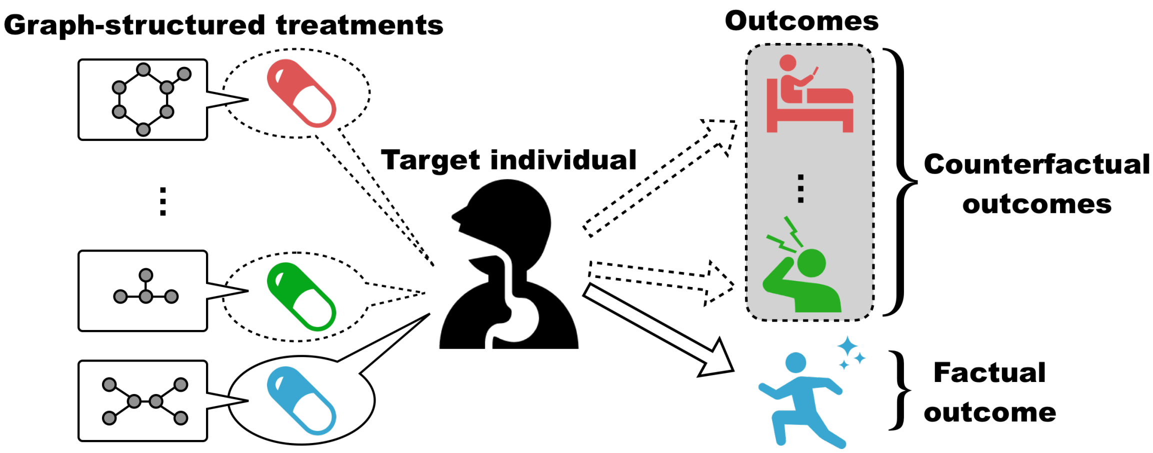


Figure 4.1: Individual effect estimation problem of graph-structured treatments. The possible treatments, i.e., drugs, are associated with graphs representing their molecular structures. In observational data, only one treatment is applied to the target individual; consequently, only the factual outcome is observed, while the counterfactual outcomes for the other treatments are not. Our goal is to predict the outcomes of all treatments for future targets.

larger. For example, when modeling drug effects on target cells, the number of candidate drugs (i.e., treatments) can be huge, while the number of observations per drug can be quite small due to the high cost of clinical trials [70, 71]. Each drug is composed of a bunch of atoms, such as carbon, oxygen, and nitrogen, and the number of drugs composed by the combination of atoms are substantially huge. A similar situation can also occur in online advertisements [72]. This scarcity of data exacerbates the aforementioned problems. More seriously, some treatments that never appeared in the observational data, such as new drugs or new ads, may appear for the first time during the test phase. Despite the significant importance of estimating unobserved treatment effects in various applications, existing methods are not capable of dealing with such “zero-shot" treatment effect estimation.

In this study, we consider the individual treatment effect estimation problem with a large number of treatments, for which there is no definitive existing solution due to extremely sparse observations. To solve the problem, we focus on auxiliary information that accompanies the treatments. Such auxiliary information is sometimes available in applications. In the drug effect example, each drug is a chemical compound with its own molecular structure, which can be represented as a graph (Figure 4.1), and it is expected to take advantage of structural patterns contained in the graph structure. The rich structural information of graphs allows us to transfer useful information for predicting outcomes from treatments with many observations to those with less observations, even to “zero-shot" treatments that have not been seen before. Therefore, our challenge in this paper is to incorporate the rich graph structure information of treatments into our treatment effect estimation model, and provide an effective learning method to mitigate biases in sparse observational data.

We propose GraphITE (pronounced “graphite”), which is an outcome prediction model for graph-structured treatments based on biased observational data. It is built upon the recent significant advances in learning representations using graph neural networks (GNNs) [66, 5]. Bias mitigation with a large (possibly infinite) number of treatments is another issue because most existing frameworks [18, 73, 72] are not designed for such cases. To reduce the treatment selection bias depending on the individual target, GraphITE finds representations of the target and treatment that are as independent of each other as possible. This is achieved by Hilbert-Schmidt Independence Criterion (HSIC) regularization, which was recently proposed by Lopez et al. [74]; we extend their framework to exploit the treatment features extracted by GNNs and give theoretical justification on how reducing biases over the representation space extracted from graph space leads to unbiased results. Our formulation makes it possible to reduce the selection bias caused by complex graph structure information, even for the zero-shot treatments that cannot be handled by existing frameworks.

We conduct experiments on two real datasets: one with a relatively small number of treatments and one with over 100 treatments. The results show that the graph structures contribute to improving the predictive performance and that the HSIC regularization is robust to the presence of selection bias.

The contributions of this study can be summarized as follows:

- We propose GraphITE, an outcome prediction model for graph-structured treatments, that can exploit auxiliary information of treatments to deal with a large number of treatments and “zero-shot" treatments.

- In order to train GraphITE from biased observational data, we extend HSIC regularization to cases where treatments have features, and give theoretical justification on how HSIC regularization making use of representations extracted from graph space contributes to mitigating biases.

- Experiments on two real-world datasets empirically demonstrate the benefits of GraphITE for biased observational data and zero-shot treatments.

4.2 Related work

Treatment effect estimation.

Treatment effect estimation is a practically important task and has been widely studied in various fields ranging from healthcare [75] and economy [40] to education [76].

One of the typical solutions is the matching method [12, 56], which compares the outcomes for pairs with similar covariates but different treatments. The propensity score, which is the probability of a target individual receiving a treatment, is introduced to mitigate the curse of dimensionality and selection bias [43]. Tree-based methods, such as Causal Forest [16] and BART [59, 15], have also been proposed and shown promising performances.

Recently, representation learning based on deep neural networks has been successfully applied to treatment effect estimation and outperformed traditional methods [18, 17, 51]. They encourage the representations of treatment and control groups to get closer to each other to reduce selection bias. In addition, confounding variables are extracted by an additional neural network that predicts treatment assignments [19]. Generative adversarial neural networks (GAN) [77] were also successfully applied to ITE estimation [78, 20]; their key idea is to train a predictive model (i.e., a generator) whose outcomes are difficult to distinguish between factual outcomes and counterfactual outcomes.

Most previous studies have focused on binary treatments, and extensions to multiple types of treatments, especially high numbers, are key research directions. There have been several approaches designed for multiple treatments [73, 16, 59]; however, most of them are limited to a relatively small number of treatments, making it difficult to consider more than a few dozen treatments. Saini et al. [72] whose motivation was somewhat similar to ours, considered combinatorial treatments; however, their focus was on a large number of combinations made from a small number of treatments, whereas we focus on many single treatments with the help of information on the treatments.

Extensions to real-valued treatments are also important for real-world applications, such as estimation of appropriate drug dosages [79, 37]. Wang et al. [80] proposed an interesting approach to learn input representations that cannot distinguish real-valued domains. Lopez et al. [74] considered total-ordered treatment spaces. They proposed HSIC regularization for dealing with biased observational data; the theory of our proposed GraphITE is based on their theoretical framework, but we extend the implications of their framework to representation learning of treatments with rich features.

Graph neural networks.

Graph-structured data is one of the most popular data structures and has been widely employed in various domains such as social network analysis, citation analysis, and chemical informatics. The GNN is one of the most successful deep neural network architectures owing to the practical importance of graph-structured data, and it has significantly improved the performance on various graph-structured data analysis tasks, such as node classification [66], graph classification [6, 5], and link prediction [81], beyond conventional methods [82, 83]. In the field of chemoinformatics, GNNs have particularly flourished and played an important role in predicting molecule properties [6, 84, 5], finding interactions between chemical objects [7], and generating desirable and unique molecules [85, 86]. GraphITE also relies on their powerful ability to extract features from graph-structured treatments.

Theoretical analysis of the expressive power of GNNs has been of great interest to researchers, for example, in their invertibility [87, 88, 86, 89]. GraphITE theoretically requires this property although it does not hold for most practical GNNs; however, a non-invertible GNN shows satisfactory performance in practice, as shown in the experimental section.

Recently, several studies have considered causal inference in *graph-structured input domains* [60, 61, 62, 90, 91], where the input space has a graph structure representing proximal relations among target individuals. However, to the best of our knowledge, no study has explored treatment effect estimation with *graph-structured treatments*, which is at the intersection of the above two topics of practical importance.

4.3 Problem definition

We consider the problem of estimating the outcomes of treatments with graph structures from biased observational data. Let $\mathcal{D} = \{(x_i, t_i, y_i^{t_i})\}_{i=1}^N \in \mathcal{X} \times \mathcal{T} \times \mathcal{Y}$ be a biased observational dataset, where $x_i \in \mathcal{X}$ is the covariate vector of the i-th target individual, $t_i \in \mathcal{T}$ is the treatment performed on the target individual, and $y_i^{t_i}$ is the outcome.[1] We assume the covariate space $\mathcal{X} = \mathbb{R}^D$, treatment space $\mathcal{T} = \{1, 2, \ldots, |\mathcal{T}|\}$, and outcome space $\mathcal{Y} = \mathbb{R}$. In addition to $\mathcal{D}$, we assume each treatment $j \in \mathcal{T}$ is associated with a graph $G_j = (\mathcal{V}_j, \mathcal{E}_j)$, where $\mathcal{V}_j$ denotes a set of nodes and $\mathcal{E}_j \subseteq \mathcal{V}_j \times \mathcal{V}_j$ denotes a set of edges. We denote the set of the treatment graphs by $\mathcal{G} = \{G_j\}_{j=1}^{|\mathcal{T}|}$. Our goal is to, given $\mathcal{D}$ and $\mathcal{G}$, estimate an outcome

[1] Owing to the counterfactual nature of the problem, we are unable to observe the outcomes of the other treatments as they are not performed.

prediction function $f : \mathcal{X} \times \mathcal{T} \rightarrow \mathcal{Y}$.

Figure 4.1 illustrates our problem setting in the context of medical treatments. In the observational data, there are multiple drugs that could be applied to the target individual, where each drug corresponds to a treatment and is associated with a graph representing its molecular structure. Only the outcome $y_i^{t_i}$for the actual prescribed drug t_i (i.e., the factual outcome) is observed, and those for the other drugs (i.e., counterfactual outcomes) are not observed. Because a doctor prescribes a drug based on the condition of the target patient x_i, there is a bias in the choice of t_i in the observational data.

A potential difficulty in our problem is that the number of treatments can be large, say $|\mathcal{T}| > 100$; it is clear that this can cause a data scarcity issue. In our scenario, graphs are available as auxiliary information for the treatments, which potentially help in dealing with such a large number of treatments.

Following the existing work, we make the typical assumptions in the Rubin-Neyman framework [35]: (i) Stable unit treatment value; the outcome of each instance is not affected by the treatments assigned to other instances. (ii) Unconfoundedness; the treatment assignment to an instance is independent of the outcome given the covariates (i.e., the confounder variables). (iii) Overlap; each instance has a positive probability of treatment assignment, i.e., $\forall x, t$, $p(x, t) > 0$.

4.4 GraphITE

Previous studies on individual treatment effect estimation have not considered rich information associated with treatments, which in our case is given as graphs. We expect that the use of such auxiliary information will be effective, especially when the number of treatments is relatively high and the training dataset is biased. We propose GraphITE, which utilizes graph-structured treatments while reducing selection bias effectively. We first introduce the network architecture of GraphITE, and then apply HSIC regularization to estimate outcomes appropriately from a biased

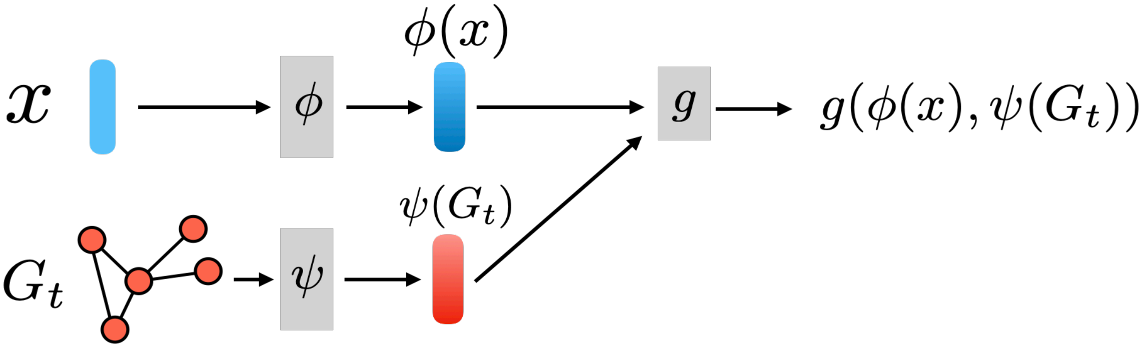


Figure 4.2: The model architecture of GraphITE. A target individual x and graph-structured treatment G_t are the inputs. The ϕ and ψ map them to the low-dimensional representations, where ϕ is a standard feed-forward neural network and ψ is a graph neural network. The two representation vectors $\phi(x)$ and $\psi(G_t)$ are concatenated to be an input to another feed-forward network g, which predicts the outcome.

dataset.

4.4.1 Model

The model of GraphITE consists of three components: two mapping functions $\phi : \mathcal{X} \to \Phi$ and $\psi : \mathcal{T} \to \Psi$ for extracting representations of the input and treatment graph, respectively, and a prediction function $g : \Phi \times \Psi \to \mathcal{Y}$ for predicting the outcome, where Φ and Ψ are the latent representation spaces of the inputs and treatments induced by ϕ and ψ, respectively. Figure 4.2 illustrates the overview of the neural network architecture of GraphITE.

As mentioned earlier, the mapping function ψ extracts representations that capture the features of graph-structured treatments. If we simply take ψ as a one-hot encoding of discrete treatments, it coincides with the standard setting with multiple treatments; however, this approach cannot take advantage of the rich structural information that the graph treatments have, and therefore suffers from a large number of treatments.

As the mapping function ψ of the treatment graphs, we employ a GNN. GNNs have been successfully applied in various domains [6, 66, 5] and are capable of extracting features of graphs owing to the flexible expressive power of neural networks optimized in an end-to-end manner.

The representation vector of graph-structured treatment $G = (\mathcal{V}, \mathcal{E})$ is otained as follows. First, for each node $v_k \in \mathcal{V}$, the representation of v_k is initialized to a low-dimensional vector $\mathbf{v}_k^{(0)} \in \mathbb{R}^{D_\Psi}$ determined by randomized initialization depending on the node label, such as the atom type. At the c-th layer of the GNN, the node representations are updated using

$$\mathbf{v}_k^{(c)} = \sigma^{\mathcal{V}} \left(\mathbf{W}\mathbf{v}_k^{(c-1)} + \sum_{v_m \in \mathcal{N}_k} \mathbf{M}\mathbf{v}_m^{(c-1)} \right), \tag{4.1}$$

where $\sigma^{\mathcal{V}}$ is an activation function, such as the ReLU function, $\mathcal{N}_k$ is the set of nodes adjacent to v_k, and $\mathbf{W}$ and $\mathbf{M}$ are transformation matrices. After the updates through C layers, the representations of all the nodes are aggregated into a graph-level representation $\psi(G)$ as

$$\psi(G) = \sum_{v_k \in \mathcal{V}} \sigma_G \left(\sum_{c=0}^{C} \mathbf{v}_k^{(c)} \right), \tag{4.2}$$

where σ_G is an activation function, such as the softmax function.

Note that, as we will see later, we require the treatment mapping function to be invertible, i.e., one-to-one, which most GNNs are not; however, some recent studies have proposed GNNs with the one-to-one property [88, 86, 89]. In our experiments, we use a non-invertible GNN which exhibits satisfactory performance.

4.4.2 Bias mitigation using HSIC regularization

With an unbiased dataset collected through randomized controlled trials (RCT), it suffices to minimize the objective function

$$\sum_{i=1}^{N} \ell(y_i^{t_i}, g(\phi(x_i), \psi(G_{t_i}))) \tag{4.3}$$

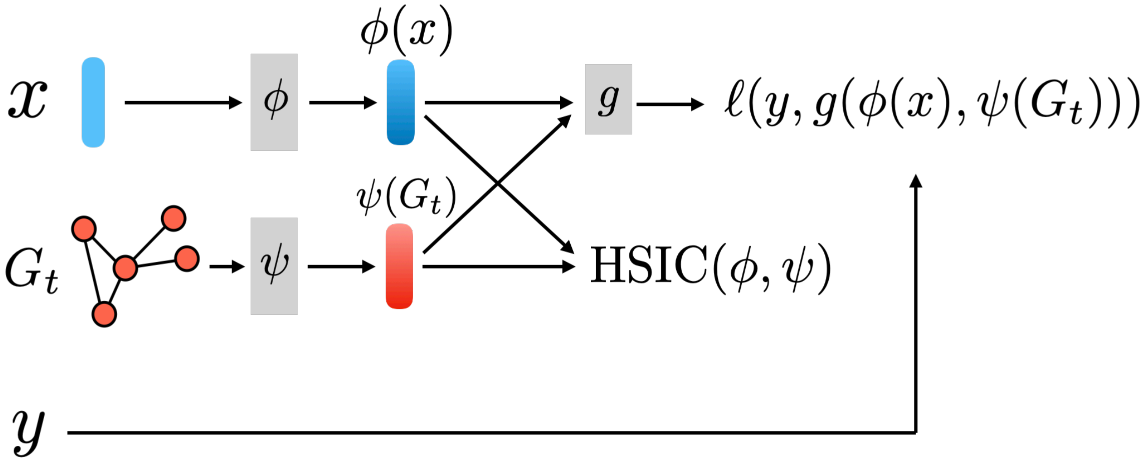


Figure 4.3: Training of GraphITE using the HSIC regularization. In addition to the prediction loss function ℓ between the prediction $g(\phi(x), \psi(G_t))$ and the true outcome y, the HSIC regularization term encourages the two representations $\phi(x)$ and $\psi(G_t)$ to be independent of each other in order to mitigate selection biases. Theorem 1 gives a theoretical guarantee on how the HSIC regularization contributes to the bias mitigation.

to estimate the components of GraphITE, ϕ, ψ, and g, where ℓ is a loss function, such as mean squared error (MSE). However, the objective function is biased when the training dataset is biased, and we must adjust it to mitigate the negative effect.

We first propose our approach from an intuitive viewpoint. The main source of the bias is that, in contrast with RCT, the treatments in the observational data are selected depending on the target individuals (i.e., the covariates). Our idea for mitigating the bias is to reduce the dependency, i.e., to find representations of the target and treatment that are as independent of each other as possible. To implement this idea, we employ HSIC [92] to measure the independence; the HSIC is defined as

$$\mathrm{HSIC}(\phi, \psi) = (N-1)^{-2}\mathrm{tr}(\mathbf{K}^{\Phi}\mathbf{H}\mathbf{K}^{\Psi}\mathbf{H}), \tag{4.4}$$

where $\mathbf{K}^{\Phi}$ and $\mathbf{K}^{\Psi}$ are the kernel matrices of the representations of the targets and treatments, respectively, and $\mathbf{H}$ is the centering matrix $\mathbf{H} = \mathbf{I} - \frac{1}{N}\mathbf{1}$. If the kernel function is characteristic, HSIC becomes 0 in expectation if and only if the two representations are independent; we use the Gaussian kernel as the kernel function. HSIC is somewhat computationally expensive, which requires $\mathcal{O}(N^2)$ time and space complexity, and does not scale to the sample size. Therefore, for the sake of computational convenience, we compute the HSIC loss in a mini-batch manner.

With the HSIC as a regularization term, our objective function is modified to

$$\sum_{i=1}^{N} \ell(y_i^{t_i}, g(\phi(x_i), \psi(G_{t_i})) + \lambda \cdot \mathrm{HSIC}(\phi, \psi), \tag{4.5}$$

where λ is the regularization hyper-parameter. The ϕ is implemented as a standard feed-forward neural network, while ψ is a GNN; the ϕ and ψ are concatenated as an input to g that is another feed-forward network. Figure 4.3 illustrates the training of GraphITE using the HSIC regularization.

The objective function (4.5) is optimized in a mini-batch manner using Adam [49]; more specifically, each epoch divides the entire training dataset $\mathcal{D}$ into mini batches without overlapping, and approximates the loss function and HSIC term with them. Recent theoretical analysis reveals that minimizing the HSIC loss in a mini-batch manner is equivalent to bagging block HSIC method [93, 94], which ensures that it converges to the same value. The training procedure for GraphITE is outlined in Algorithm 2.

Finally, we give some historical remarks explaining why we specifically chose the HSIC as the regularization term. Previous studies have considered a broad class of regularization terms, integral probability metrics (IPM) [17, 18]; however, they basically assume a binary treatment or a relatively small number of treatments, and they cannot be directly applied to a large number of treatments. For example, typical IPM such as the maximum mean discrepancy (MMD) and Wasserstein distance, require many regularization terms for all pairs of treatments; otherwise, an expedient “pivot" control treatment must be set, which is not effective, as demonstrated in our experiments. The use of the HSIC is proposed by Lopez et al. [74] as it naturally allows multiple treatments. However, they did not consider learning representations of treatments.

Algorithm 2: GraphITE training procedure

```
Input: Observational data: $\mathcal{D} = \{(x_i, t_i, y_i^{t_i})\}_{i=1}^{N} \sim p_{\text{train}}$,
a set of graph-structured treatments $\mathcal{G} = \{G_j\}_{j=1}^{|\mathcal{T}|}$,
and a hyperparameter $\lambda \geq 0$.
Output: An outcome prediction model $f = (g, \phi, \psi)$.
while not converged do
    Sample a mini batch $\mathcal{B} = \{(x_{i_o}, t_{i_o}, y_{i_o}^{t_{i_o}})\}_{o=1}^{B} \subset \mathcal{D}$
    # Mini-batch approximation of the supervised loss (Eq. (4.3))
    Compute $L_{\mathcal{B}}(g, \phi, \psi) = \sum_{o=1}^{B} \ell(y_{i_o}, g(\phi(x_{i_o}), \psi(G_{t_{i_o}})))$
    # Mini-batch approximation of the HSIC loss (Eq. (4.4))
    Compute $\mathrm{HSIC}_{\mathcal{B}}(\phi, \psi) = \mathrm{HSIC}(\phi_{x \in \mathcal{B}}, \psi_{t \in \mathcal{B}})$
    # Update the parameters of $f$
    Minimize $L_{\mathcal{B}}(g, \phi, \psi) + \lambda \cdot \mathrm{HSIC}_{\mathcal{B}}(\phi, \psi)$ using SGD
end
```

4.4.3 Theoretical justification of HSIC regularization

Now we consider the theoretical justification for using the HSIC regularization in GraphITE. Our discussion is based on generalizations of the theories [74, 95] to treatments with features. We also discuss the benefits of our formulation and how it makes the prediction model more flexibly deal with complex situations than the existing approaches.

Denote by p_{train} the probability distribution on $\mathcal{X} \times \mathcal{T}$ the training dataset $\mathcal{D}$ follows, and by p_{test} the one for the test dataset. We assume that the test distribution has the form of $p_{\text{test}}(x, t) = p^{\mathcal{X}}(x) p^{\mathcal{T}}(t)$ because we want our model to perform well on the distribution where treatments do not depend on the covariates.

For some (unknown target) function $f^* : \mathcal{X} \times \mathcal{T} \to \mathcal{Y}$ and probability distribution p over $\mathcal{X} \times \mathcal{T}$, let the expected risk of our prediction model $f : \mathcal{X} \times \mathcal{T} \to \mathcal{Y}$ with the mapping functions ϕ, ψ and a predictive function g be

$$\epsilon_p(f_{g,\phi,\psi}) = \mathbb{E}_{(x,t) \sim p}[\ell(f(x, t), f^*(x, t))], \tag{4.6}$$

where $f(x, t) \coloneqq g(\phi(x), \psi(G_t))$.

Then we have the following theoretical upper bound of the expected risk on the test distribution in terms of that for the training distribution and the HSIC regularization term.

Theorem 1. *Let $\epsilon_{p_{train}}(f_g)$ and $\epsilon_{p_{test}}(f_g)$ be the expected risk for the training distribution and the test distribution, respectively. Let the IPM between two distributions p and q with respect to a function family $\mathcal{H}$ be*

$$IPM_{\mathcal{H}}(p, q) = \sup_{h \in \mathcal{H}} |\mathbb{E}_p[h] - \mathbb{E}_q[h]|. \tag{4.7}$$

Let $J_\phi^{-1}(\xi)$, $J_\psi^{-1}(\tau)$ be the Jacobian matrices of ϕ^{-1} and ψ^{-1} at ξ and τ, respectively. Assume that there exist positive constants A and B that satisfy $|J_{\phi^{-1}(\xi)}||J_{\psi^{-1}(\tau)}| \leq A$ and $\frac{\ell_{f_{g,\phi,\psi}}}{B} \in \mathcal{H} \subseteq \{g : \Phi \times \Psi \to \mathcal{Y}\}$. Then the expected risk for the test distribution is upper-bounded by

$$\epsilon_{p_{test}}(f_{g,\phi,\psi}) \leq \epsilon_{p_{train}}(f_{g,\phi,\psi}) + \lambda \cdot HSIC(\phi, \psi). \tag{4.8}$$

Proof.

$$
\begin{aligned}
&\epsilon_{p_{\text{test}}}(f_{\phi,\psi}) - \epsilon_{p_{\text{train}}}(f_{g,\phi,\psi}) \\
=&\int_{\mathcal{X}\times\mathcal{T}} \ell_{f_{g,\phi,\psi}}(x,t)(p(x)p(t))dxdt - \int_{\mathcal{X}\times\mathcal{T}} \ell_{f_{g,\phi,\psi}}(x,t)p(x,t))dxdt \\
=&\int_{\mathcal{X}\times\mathcal{T}} \ell_{f_{g,\phi,\psi}}(x,t)(p(x)p(t) - p(x,t))dxdt \\
=&\int_{\Phi\times\Psi} \ell_{f_{g,\phi,\psi}}(\phi^{-1}(\xi),\psi^{-1}(\tau))(p(\xi)p(\tau) - p(\xi,\tau)) \\
&\qquad \cdot |J_{\phi^{-1}(\xi)}||J_{\psi^{-1}(\tau)}|d\xi d\tau \\
\leq& A\int_{\Phi\times\Psi} \ell_{f_{g,\phi,\psi}}(\phi^{-1}(\xi),\psi^{-1}(\tau))(p(\xi)p(\tau) - p(\xi,\tau))d\xi d\tau \\
\leq& A\cdot B \sup_{g\in\mathcal{H}} \left| \int_{\Phi\times\Psi} g(\phi^{-1}(\xi),\psi^{-1}(\tau))(p(\xi)p(\tau) - p(\xi,\tau))d\xi d\tau \right| \\
\leq& A\cdot B\cdot \mathrm{IPM}_{\mathcal{H}}(p(\xi)p(\tau), p(\xi,\tau)) \\
\leq& \cdot A\cdot B\cdot C\cdot \mathrm{HSIC}(p(\xi),p(\tau)),
\end{aligned}
$$

where $\ell_{f_{g,\phi,\psi}} = \mathbb{E}_y[\ell(y^t, g(\phi(x),\psi(G_t)) \mid x,t]$. C is a constant that denotes the radius of the function space. By setting $\lambda = A\cdot B\cdot C$, we obtain the inequality (4.8). □

The theorem states that minimizing the HSIC between the representations of the targets and graph-structured treatments leads to minimizing the expected risk for the test distribution, making the predictive model to handle even unseen graph-structured treatments unbiasedly. For the inequality (4.8) to hold, we require several conditions: ϕ and ψ must be twice-differentiable one-to-one mapping functions, and the HSIC must be defined using continuous, bounded, positive semi-definite kernels $k^\Phi : \Phi\times\Phi \to \mathbb{R}$ and $k^\Psi : \Psi\times\Psi \to \mathbb{R}$. The λ must also be theoretically determined based on the radius of the function space in which f lies, but empirically, we simply treat it as a hyper-parameter. Our choices of the kernels and the hyper-parameters are detailed in Section 4.5.

Note that MMD and Wasserstein distance are special cases of IPM when the function family includes the set of 1-Lipschitz functions and the set of unit norm

Table 4.1: Summary statistics of the datasets.

Dataset	#Units	#Treatments	#Interactions
CCLE	491	24	11,054
GDSC	925	117	105,694

functions in a universal reproducing norm Hilbert space [18], respectively. Hence, they are obviously bounded by the inequality (4.8).

4.5 Experiments

We experimentally investigate the performance of the proposed GraphITE and its merits of using the GNN and the HSIC regularization compared with various baseline methods on two real-world datasets.

4.5.1 Datasets

We use two real-world datasets on drug responses: the Cancer Cell-Line Encyclopedia (CCLE) dataset [96] and Genomics of Drug Sensitivity in Cancer (GDSC) dataset [97]. Table 4.1 lists their basic statistics. #Units, #Treatments, and #Interactions represent the number of units, the number of treatments, and the number of labeled data in a dataset. CCLE is a relatively small dataset with a moderate number of treatments, while GDSC has more than 100 treatments. Both of the datasets include IC_{50} values for drug–cell pairs, which are known to be closely related to drug sensitivity. Namely, we define the drug sensitivity as $y = -\log \mathrm{IC}_{50}$ following previous studies [98, 99], which is the regression target in our experiments. We use the similarity matrices of each cell line as the input features. Both datasets are publicly available[2] [99].

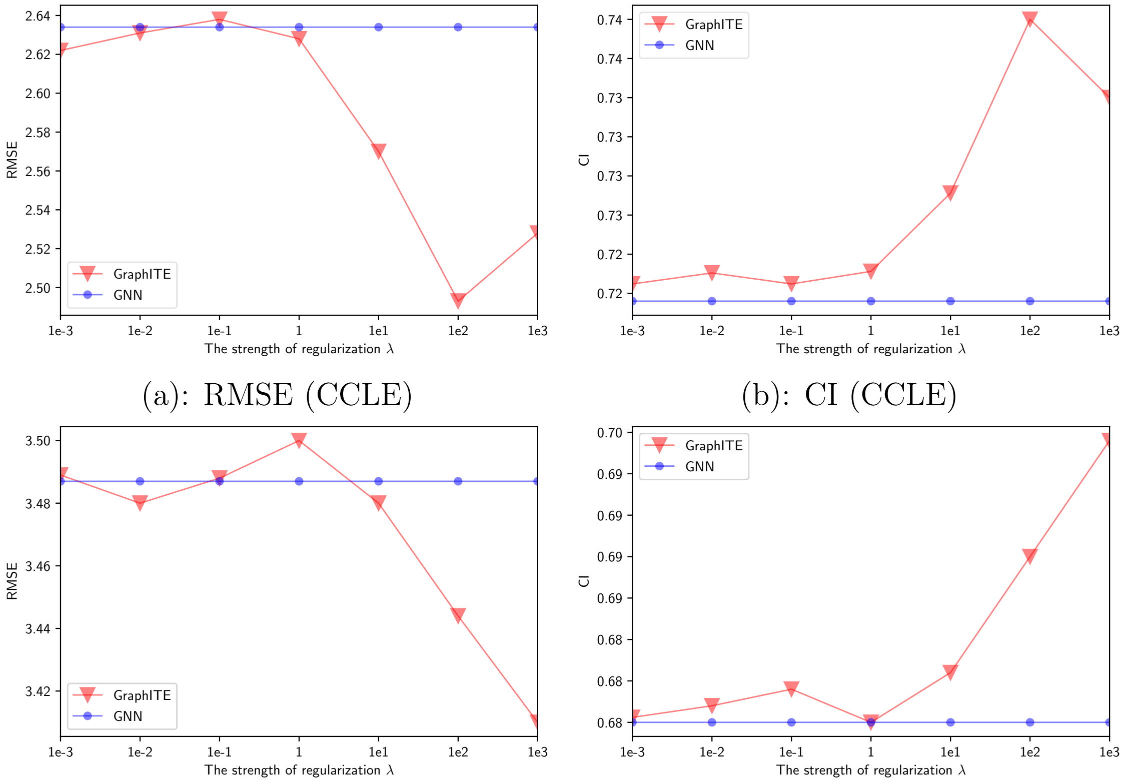
Because the two original datasets are fairly close to complete observations (specifically, their observation rates are about 94% and 98%, respectively), we simply as-

[2] `https://github.com/CSB5/CaDRReS`

Table 4.2: Performance comparison of different methods on the CCLE and GDSC dataset in terms of RMSE and CI. † and $\ddagger$ indicate statistically significantly better performance of the proposed GraphITE than the baseline by the paired t-test with $p < 0.05$ and $p < 0.01$, respectively. The bold results indicate the statistically significant best results. The shaded rows indicate the GNN-based methods. Lower RMSEs are better, and higher CIs are better.

	CCLE		**GDSC**	
Method	RMSE	CI	RMSE	CI
Mean	$^{\ddagger}3.777_{\pm 0.101}$	–	$^{\ddagger}4.030_{\pm 0.102}$	–
OLS	$^{\ddagger}4.861_{\pm 0.755}$	$^{\ddagger}0.642_{\pm 0.021}$	$^{\ddagger}6.463_{\pm 0.493}$	$^{\ddagger}0.602_{\pm 0.018}$
BART	$^{\ddagger}2.993_{\pm 0.203}$	$^{\ddagger}0.711_{\pm 0.016}$	$^{\ddagger}3.965_{\pm 0.102}$	$^{\ddagger}0.632_{\pm 0.015}$
Treatment Embedding	$^{\ddagger}2.662_{\pm 0.161}$	$^{\ddagger}0.724_{\pm 0.013}$	$^{\ddagger}3.642_{\pm 0.131}$	$^{\ddagger}0.670_{\pm 0.015}$
TARNet	$^{\ddagger}2.831_{\pm 0.123}$	$^{\ddagger}0.711_{\pm 0.013}$	$^{\ddagger}3.813_{\pm 0.135}$	$^{\ddagger}0.663_{\pm 0.009}$
CFR	$^{\ddagger}2.822_{\pm 0.121}$	$^{\ddagger}0.712_{\pm 0.013}$	$^{\ddagger}3.792_{\pm 0.134}$	$^{\ddagger}0.664_{\pm 0.009}$
GANITE	$^{\ddagger}3.652_{\pm 0.211}$	$^{\ddagger}0.651_{\pm 0.023}$	$^{\ddagger}7.739_{\pm 1.394}$	$^{\ddagger}0.613_{\pm 0.018}$
GNN	$^{\ddagger}2.652_{\pm 0.123}$	$^{\ddagger}0.720_{\pm 0.010}$	$^{\ddagger}3.553_{\pm 0.126}$	$^{\ddagger}0.681_{\pm 0.010}$
GNN+MMD	$^{\dagger}2.596_{\pm 0.162}$	$^{\dagger}0.726_{\pm 0.014}$	$^{\ddagger}3.531_{\pm 0.136}$	$^{\ddagger}0.683_{\pm 0.013}$
GraphITE (Proposed)	$\mathbf{2.561}_{\pm \mathbf{0.112}}$	$\mathbf{0.732}_{\pm \mathbf{0.009}}$	$\mathbf{3.421}_{\pm \mathbf{0.135}}$	$\mathbf{0.695}_{\pm \mathbf{0.015}}$

sume that they are complete, and introduce synthetic observation biases to extract biased training datasets, and then test the predictors obtained from them on the remainder. We introduce synthetic treatment bias that assigns treatment t using $t \sim \text{Categorical}(\text{softmax}(\rho y))$ following previous studies [79, 73, 78]. The $\rho = \frac{\eta}{100\sigma}$ is a bias coefficient, where η is the magnitude of selection bias and σ is the standard deviation of target values. A larger η indicates a higher selection probability; intuitively, this indicates that scientists are more likely to conduct experiments for drug–cell pairs with higher sensitivity values. Note that although this bias generation procedure does not necessarily satisfy the typical unconfoundedness assumption, it is more practical and reasonable in the sense that the scientists likely to select promising experimental targets based on their knowledge and experience. In other words, we assume scientists are not incompetent.



(a): RMSE (CCLE)

(b): CI (CCLE)

(c): RMSE (GDSC)

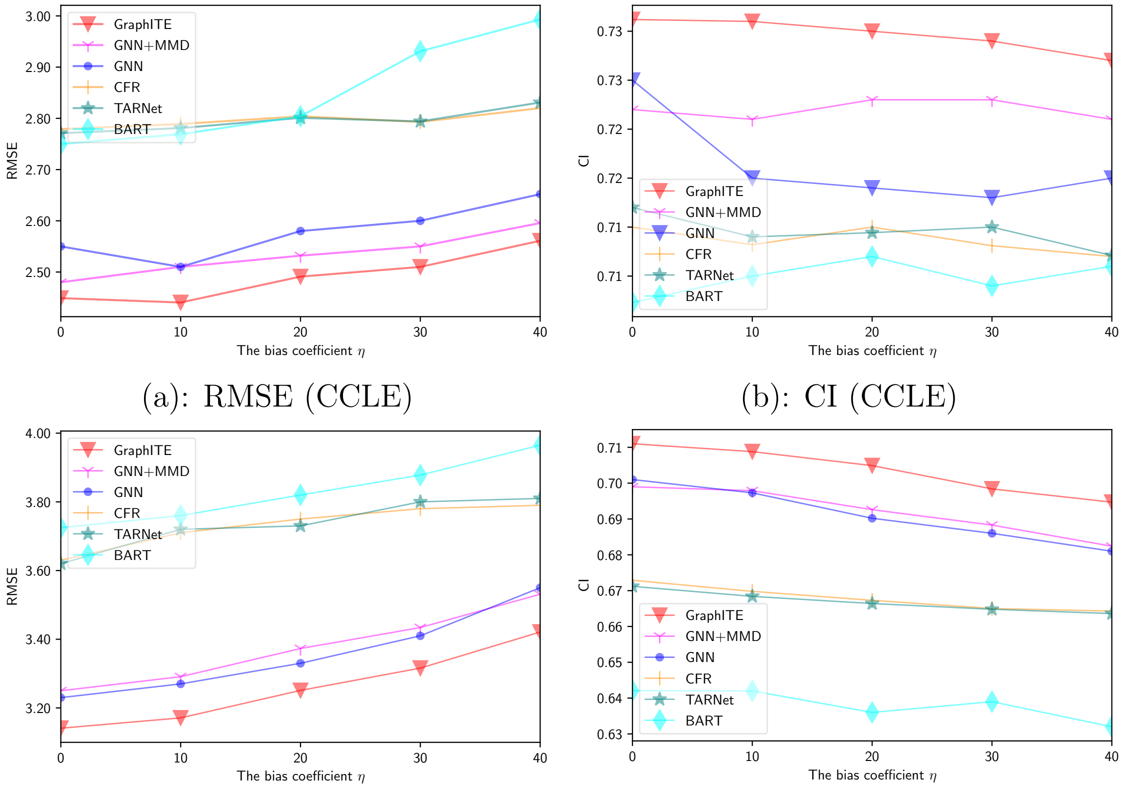
(d): CI (GDSC)

Figure 4.4: Sensitivity of the results to the regularization strength λ. Although the optimal choices significantly improve the performance, at worst the other choices do not harm the performance.

4.5.2 Baseline methods

We compare GraphITE with the following six baselines. (i) Ordinary least squares linear regression (OLS) concatenates two vectors, the covariate vector and treatment vector coded as a one-hot vector, which is used as the input. (ii) Bayesian additive regression trees (BART) [59, 15] predicts the outcomes by an ensemble of multiple regression trees; we used a Python implementation of BART[3]. (iii) Treatment embedding method exploits low-dimensional representations of treatments to deal with a large number of treatments. Each treatment is associated with a low-dimensional vector, which is input to a neural network, as well as a covariate vector. Note that this method does not use the graph structures of the treatments at all.

[3] `https://github.com/JakeColtman/bartpy`

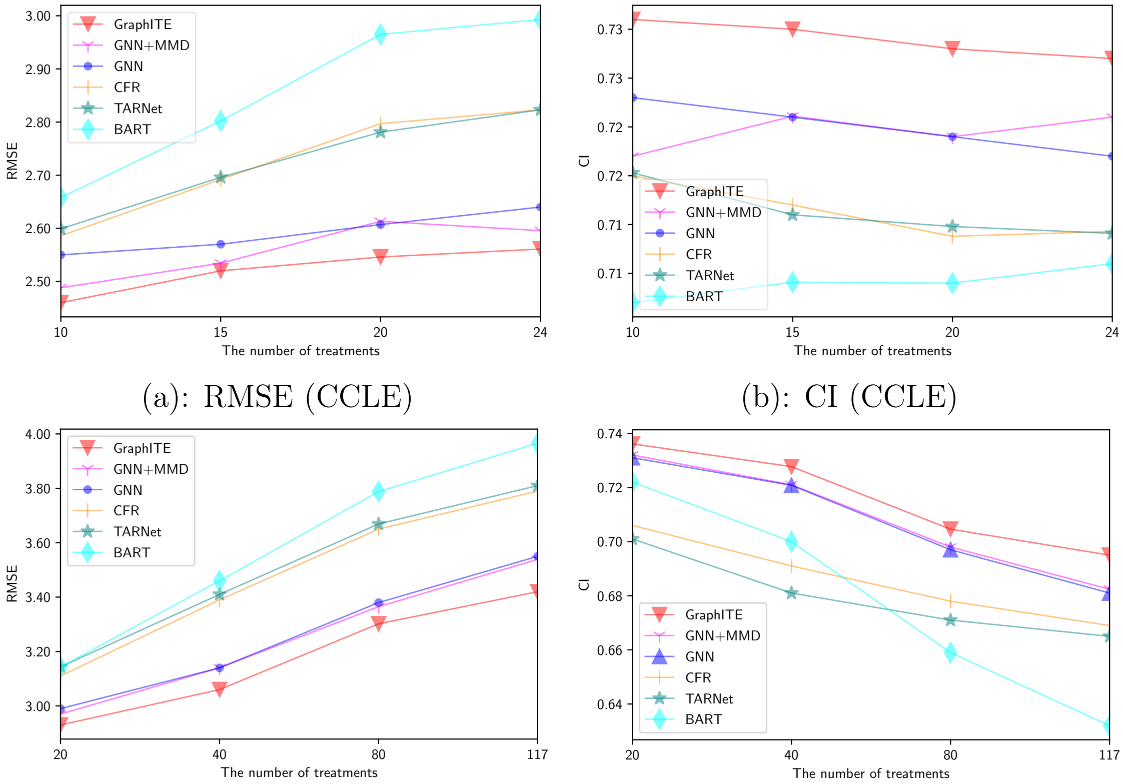


(a): RMSE (CCLE) (b): CI (CCLE)

(c): RMSE (GDSC) (d): CI (GDSC)

Figure 4.5: Predictive performance depending on the bias coefficient η. A large η indicates a larger selection bias. GraphITE shows its strong and stable tolerance to the biases and consistently performs the best in the whole range.

(iv) TARNet [18] is a deep neural network model with shared layers for representation learning and different layers for outcome prediction for treatment and control instances. (v) Counterfactual regression (CFR) [18] is one of the state-of-the-art deep neural network models based on balanced representations between treatment and control instances; we used the MMD as its IPM. Following previous studies [20, 72], we extend the CFR [18] to the multiple-treatment setting; we regard the most frequent treatment as the control treatment. (vi) GANITE [20] is another state-of-the-art deep neural network model based on GAN. It trains a TARNet-like generator that generates counterfactual outcomes, and a discriminator tells whether outcomes come from the generator or the real distribution. In the original GANITE, the discriminator just tries to solve a binary classification; on the other hand in our setting,



(a): RMSE (CCLE)

(b): CI (CCLE)

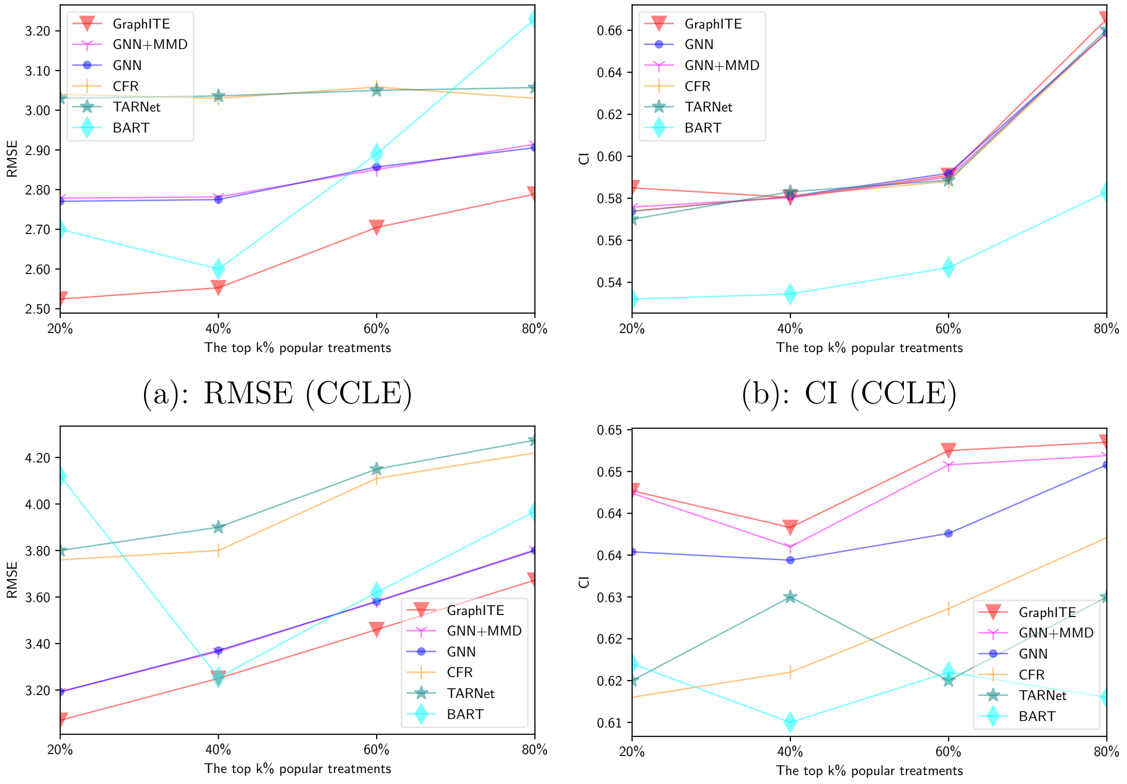
(c): RMSE (GDSC)

(d): CI (GDSC)

Figure 4.6: Predictive performance depending on the the number of treatments. Whereas the baseline methods get degraded as the number of treatments increase, graphite shows relatively robust to its increase and achieves the best performances, especially in the larger dataset (GDSC).

GANITE has to solve multi-class classification to tell which outcome is the genuine one.

In addition to the use of graph structured treatments, one of the key features of GraphITE is the bias mitigation using HSIC regularization; therefore, we use the versions without it our baseline methods for the ablation study. We also tested several variants of GraphITE: (vii) a variant with no bias mitigation that does not have the HSIC regularization term and only uses a GNN, which we refer to as “GNN” hereafter and (viii) another variant using MMD regularization instead of HSIC regularization. We used the same approach as CFR to deal with multiple treatments. We denote it by “GNN+MMD".



(a): RMSE (CCLE)

(b): CI (CCLE)

(c): RMSE (GDSC)

(d): CI (GDSC)

Figure 4.7: Predictive performance depending on treatment popularity. From the RMSE results, the methods that do not rely on treatment graph information (CFR, TARNet, BART) suffer from a lack of data especially for unpopular treatments. From the CI results, the methods that have no bias mitigation mechanism put too much attentions on popular treatments (i.e., difficult in terms of ranking) treatments, and perform suboptimally. GraphITE shows the most stable and best performance on every group.

4.5.3 Experimental setting

As the evaluation metrics, we employ the root mean square error (RMSE) of all target–treatment pairs in the test set defined as:

$$\mathrm{RMSE} = \sqrt{\frac{1}{N^{\mathrm{test}}} \frac{1}{|\mathcal{T}|} \sum_{i=1}^{N^{\mathrm{test}}} \sum_{t=1}^{|\mathcal{T}|} (y_i^t - f(x_i, t))^2}, \tag{4.9}$$

where N^{test} is the number of target individuals included in the test dataset. We also employ the concordance index (CI) [100] to evaluate predictive performance

Table 4.3: Performance comparison of different methods on the CCLE and GDSC dataset in terms of RMSE and CI. † and ‡ indicate statistically significantly better performance of the proposed GraphITE than the baseline by the paired t-test with $p < 0.05$ and $p < 0.01$, respectively. The bold results indicate the statistically significant best results. The shaded rows indicate the GNN-based methods. Lower RMSEs are better, and higher CIs are better.

	CCLE		**GDSC**	
Method	RMSE	CI	RMSE	CI
Mean	$3.458_{\pm 1.301}$	–	$^{\dagger}4.705_{\pm 0.702}$	–
GNN	$3.920_{\pm 0.932}$	$0.551_{\pm 0.130}$	$^{\ddagger}4.646_{\pm 0.631}$	$0.570_{\pm 0.061}$
GNN+MMD	$3.903_{\pm 0.923}$	$0.549_{\pm 0.132}$	$^{\ddagger}4.640_{\pm 0.674}$	$0.574_{\pm 0.061}$
GraphITE	$3.637_{\pm 0.905}$	$0.545_{\pm 0.114}$	$\mathbf{4.482_{\pm 0.595}}$	$0.569_{\pm 0.054}$

in terms of ranking accuracy, which has been widely used in previous studies [101, 102]. The CI is defined as

$$\text{CI} = \frac{1}{N^{\text{test}}} \sum_{i=1}^{N^{\text{test}}} \sum_{t,u \mid y_i^t > y_i^u} \frac{\theta(f(x_i, t) - f(x_i, u))}{|\{t, u \mid y_i^t > y_i^u\}|}, \tag{4.10}$$

where γ is the Heaviside step function defined as

$$\theta(x) = \begin{cases} 1, & x > 0 \\ 0.5, & x = 0 \\ 0 & x < 0 \end{cases} \cdot \tag{4.11}$$

Note that the CI is identical to ROC-AUC when all outcomes are binary.

We split the whole individuals into 80%, 10%, and 10% for training, validation, and testing sets, respectively. We report the average results of 50 different trials. Note that while we sample the factual treatments in the training and validation sets following the biased sampling scenario explained in the Dataset section, all of the treatments are included in the test set because we want our prediction model to perform uniformly well on all treatments.

In GraphITE, to promote effective feature extraction from small data, we pre-train the GNN ψ on regression tasks using three popular molecular datasets: ESOL, FreeSolv, and Lipophilicity, provided by MoleculeNet [103][4]. To the best of our observation, GraphITE performs slightly better with pre-training than the one without pre-training. We believe that this phenomenon may be caused by the small size of the training data, and is the one of limitations of this study to be addressed in future work. For the HSIC regularization, we use the normalized version of the HSIC (nHSIC), defined as

$$\mathrm{nHSIC}(\phi, \psi) = \frac{\mathrm{tr}(\mathbf{K}^{\Phi}\mathbf{H}\mathbf{K}^{\Psi}\mathbf{H})}{\|\mathbf{K}^{\Phi}\mathbf{H}\|\|\mathbf{K}^{\Psi}\mathbf{H}\|}, \tag{4.12}$$

where $\|\cdot\|$ denotes the Frobenius norm. The regularization parameter λ is optimized in $\{10^{-3}, 10^{-2}, \ldots, 10^{3}\}$ based on the RMSE for the validation set. We set the number of representation dimensions for target individuals and graph-structured treatments to 64 because we did not observe the significant differences in $\{16, 32, 64, 128\}$ in the both datasets. Similarly, we set the numbers of layers of ϕ, ψ, and g to 3.

4.5.4 Results

Table 4.2 summarizes the predictive performances of the different methods for $\eta = 40$ (i.e., the strongest bias). In the remainder, we report the experimental results when we set $\eta = 40$ unless otherwise stated.

The deep learning-based methods (Treatment Embedding, TARNet, CFR) outperform the naïve methods, such as Mean and OLS. BART also gives the comparable performance. However, we observe GANITE performs poorly in comparison with the other deep learning-based methods. We believe this is because of the difficulty in learning the GAN models, that is, GANITE has to solve many-class classification with a limited amount of training data. The existing bias mitigation methods that are simply extended to multiple treatments (CFR and GNN+MMD) do not not show

[4] `http://moleculenet.ai/`

significant improvements over the corresponding original ones (TARNet and GNN). By contrast, GraphITE achieves the best performance, which is statistically significant against all the baselines, and it demonstrates its effectiveness, especially on the larger dataset (GDSC). The merit of exploiting the graph structures associated with treatments can also be seen in Table 4.2. The GNN-based methods (shaded rows) perform better than the methods that neglect graph-structured information.

Figure 4.4 shows the sensitivity of the results to the strength of HSIC regularization; although the optimal choices significantly improves the performance, none of the other choices harm the performance. The results also highlight the effectiveness of the HSIC as the choice for the regularization term; MMD shows no remarkable improvement over the plain GNN because it cannot handle many treatments efficiently, whereas GraphITE using HSIC regularization shows distinct improvements.

Now, we investigate the robustness of GraphITE against selection bias. Figure 4.5 shows the performances for different bias strengths, where a larger η represents a larger selection bias. GraphITE shows its strong and stable tolerance to the biases and consistently performs best under all bias strength settings.

The impacts of the number of treatments are shown in Figure 4.6. For small numbers of treatments, both GraphITE and the other baseline methods perform similarly well; however, the baseline methods, especially BART, degrade the performances as the number increase. GNN+MMD does not show improvements from the plain GNN, particularly on the larger dataset (GDSC). GraphITE shows the remarkable robustness to the selection bias, even with large numbers of treatments.

Next, we investigate the predictive performance based on treatment popularity, as shown in Figure 4.7. We focus on the treatment groups that are grouped by their popularity, namely, the top 20%, 20–40%, 40–60%, 60–80% most popular treatment groups. As can be seen from the RMSE results, the methods that do not rely on treatment graph information (CFR, TARNet, and BART) suffer from a lack of training data especially for unpopular treatments; On the other hand, the methods exploiting the auxiliary information mitigate this problem.

Now we turn to the CI results. From its definition (4.10), CI measures ranking performance. It is more difficult to estimate accurate ranking for popular treatments rather than less popular treatments, because in our bias setting, more popular treatments have larger outcomes, and they are more heterogeneous sets than less popular ones. On the other hand, it is rather easy to obtain small RMSEs for popular treatments because their outcomes have small variances. The methods that have no bias mitigation mechanism put too much attentions on such difficult treatments (in terms of ranking), and eventually perform suboptimally, while GraphITE shows the most stable and best performance on every group.

Finally, we investigate the predictive performance on the unobserved "zero-shot" treatments that are not included in training data. We keep 30% of the entire treatments aside in advance as the validation and target unobserved treatments. Table 4.3 shows the prediction accuracy for the unobserved treatments when we set $\eta = 40$. Note that the existing methods such as CFR are incapable of dealing with this setting. In the smaller dataset, CCLE, the selection bias prevents the models from working appropriately, and all of the variants perform worse than the simply baseline taking the mean of training data in terms of RMSE. On the other hand in the larger dataset, GDSC, GraphITE achieves the best RMSE, while the other methods still suffer from the bias. However, we do not observe significant improvements in terms of CI in the both datasets.

4.6 Conclusion

In this study, we proposed GraphITE, which can handle graph-structured treatments in order to achieve better treatment effect estimation even when the number of treatments is large. GraphITE is based on the recent developments of deep neural networks and representation learning, namely, GNNs and HSIC regularization, which contribute to improving estimation accuracy of complex -structured treatments from biased observational data. In addition, GraphITE is applicable to previously unobserved “zero-shot" treatments, which the existing ITE estimation methods are intrinsically not capable of dealing with. In the experiments on two real-world drug response datasets, GraphITE achieved the best performances in terms of RMSE and CI when compared to the various baselines. In particular, we observed a significant improvement when the effect of selection bias and the number of treatments were large. A potential future direction is to consider other types of complex structured data, such as texts, images, and videos. We also plan to apply GraphITE to much larger datasets, in which we expect further improvements.

Chapter 5

InfoCEVAE: Treatment Effect Estimation with Hidden Confounding Variables Matching

5.1 Introduction

Treatment effect estimation plays an essential role in decision making in various domains, such as healthcare, economic policy, and education. The goal of treatment effect estimation is to estimate the effect of an action by a decision maker. The main difficulty of treatment effect estimation based on observational data is that a treatment assignment is not randomized, which is often referred to as observational or selection bias. For example, elderly people might be more likely to receive drug treatment than younger people. In this example, age is a variable that impacts treatment assignment and outcome. This variable is called a confounding variable. We need to find such confounding variables to mitigate bias and give appropriate treatment effect.

In the context of treatment effect estimation, many studies usually assume that observational data include all the confounding variables. However, this assumption seems too strong and not realistic because we cannot always practically obtain sufficient information regarding individuals to guarantee that we observe all the confounding variables. Confounding variables that are not included in observational data are often referred to as hidden confounding variables. For example, private and sensitive individual information like income might be difficult to obtain, but this variable can have an effect on treatment assignment and outcome. Without knowing confounding variables, it is impossible to know the true treatment effect, and treating proxy variables as confounding variables will lead to incorrect estimands [104, 24]. Fig. 5.1 illustrates a graphical model of the data generation process. In this graphical model, it is indispensable to infer $\mathbf{z}$ correctly to know the true treatment effect. Prior studies have used strong assumptions that they have knowledge regarding the nature of hidden confounding variables beforehand, like the number of categories of hidden confounding variables [105]. These assumptions limit the application range of these approaches.

Recently, the Causal Effect Variational Autoencoder (CEVAE, the VAE-based method has been successfully incorporated into treatment effect estimation with the existence of hidden confounding variables [24, 28]. One of the advantages of VAE is that it can recover a large class of hidden confounding variable models thanks to the expressive power of neural networks [106]. Previous researches require that we know the nature of hidden confounding variables, such as the number of categories.

Xu et al. [26] employed a deep learning-based technique but they also assumed that they can distinguish variables that have an effect only on treatment assignment from variables which have an effect only on outcome. This assumption requires prior knowledge and seems unrealistic.

However, recent theoretical analysis revealed that the global optimum of VAE evidence lower bound (ELBO) does not correctly model the data generation process [34] because VAE focus on reconstruction loss too much, which becomes severer when input variables have much higher dimensions than latent variables. To mitigate this problem, InfoVAE [34], which adds a mutual information regularizer to the VAE loss function, was proposed.

This phenomenon obviously arises in VAE-based methods for treatment effect estimation and makes recovering hidden confounding variables by VAE difficult. We first remark there are datasets that the optimal solution of VAE-based methods, such as CEVAE, does not give the correct treatment effect. This is a strict limitation without any guarantee when they achieve optimal solution even though they are capable of recovering them.

To mitigate these problems, we propose hidden confounding variable matching VAE, which combines VAE with information regularization and matching to give appropriate treatment effect. The proposed method obtains the correct treatment effect when it achieves the optimal solution of its loss function, even under the existence of hidden confounding variables. We summarize the contribution of this study as follows:

- To the best of our knowledge, this is the first work that shows the optimal solution of naive VAE-based methods is not a correct average treatment effect (ATE) for types of datasets.
- We propose an effective method based on information regularization and matching algorithm to mitigate hidden confounding variables and bias, with theoretical guarantee.
- In experiments using semi-synthetic and synthetic datasets, the proposed method significantly outperformed existing methods.

5.2 Related work

5.2.1 Treatment effect estimation

Treatment effect estimation plays a essential role in decision making across various domains, such as healthcare [75, 107], economic policy [40], and education [76].

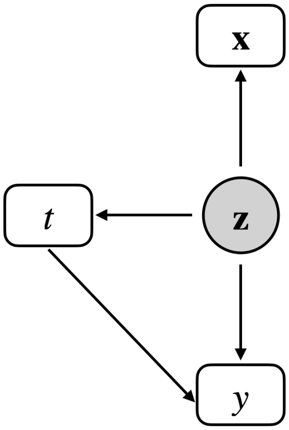


Figure 5.1: A graphical model for the treatment effect estimation methods with hidden confounding variables, which is the same as a graphical model introduced in Figure 2.1(b). Hidden confounding variable $\mathbf{z}$ has an effect on treatment assignment and outcome. Treating proxy variables $\mathbf{x}$ as normal confounding variables gives incorrect treatment effect estimation.

We outline important studies, ranging from established methods to modern deep learning-based methods. The goal of treatment effect estimation is to understand the effect of a specific action, i.e., treatment. One of the classic methods for treatment effect estimation is matching [12, 56, 108]. Matching methods estimate the counterfactual outcomes by the nearest neighbor of each individual in terms of covariates. Because the curse of dimensionality makes finding appropriate nearest neighbors of each individual more difficult, propensity score matching, which defines nearest neighbors in terms of propensity score, was developed [43, 57]. Tree-based methods, such as Random forest and Bayesian additive regression trees (BART), have also been applied [59, 15].

Recently, deep learning-based methods have been successfully applied to the treatment effect estimation problem [18, 17, 51, 20, 24, 28, 60, 90, 109]. Counterfactual regression (CFR) encourages individual representation of each treatment group extracted by neural networks to get closer to each other. Perfect matching combines neural networks and propensity score matching [73], and Counterfactual propagation, which also integrates matching and graph-based semi-supervised learning, aims to estimate treatment effect using a large number of unlabeled individual data [90]. In particular, VAE-based methods [24, 28] have been developed to mitigate the hidden confounding variable problem. They aim to recover hidden confounding variables by the strong expressive power of neural networks. Network structured-data also have been utilized to infer hidden confounding variables effectively [60].

5.2.2 VAE

VAE is one of the most famous deep generative models [29] and has been widely employed in various domains, such as computer vision [110], natural language processing [111], and chemoinformatics [112]. One of the advantages of VAE-based generative models is their strong expressive power based on neural networks. VAE has also been successfully applied in treatment effect estimation [24, 28]. The idea is to recover a joint distribution including hidden confounding variables expressed as latent variables to estimate treatment effect. However, recent theoretical analysis revealed that VAE will ignore the latent variables in the global optimum of the VAE loss function [34]. Hence, due to the nature of the VAE loss function, VAE-based treatment effect estimation methods face the unavoidable issue that they do not provide the correct treatment effect estimation even when their loss function achieves the optimal solution, which we will discuss in this paper.

Our goal is to fill the gap between VAE theoretical analysis and VAE-based treatment effect estimation methods, proposing an efficient method that provides theoretical guarantee of treatment effect even when there are hidden confounding variables.

5.3 Problem statement

In this section, we state the problem setting of treatment effect estimation. Suppose $\mathbf{x}_i \in \mathcal{X} \subset \mathbb{R}^{d_\mathbf{x}}$ is the $d_\mathbf{x}$ dimensional proxy variables of the i-th individual, $t_i \in \mathcal{T} = \{0, 1\}$ is the binary treatment applied to the i-th individual, and $y_i^{t_i} \in \mathcal{Y} \subset R$ is its outcome of the i-th individual. We omit the notation i of a variable when the variable can represent any individual. Given a dataset $\mathcal{D} := (\mathbf{x}_i, t_i, y_i^{t_i})_{i=1}^N$, which includes N individuals, our goal is to estimate the Average Treatment Effect (ATE) and conditional ATE (CATE) defined as:

$$\text{ATE} := \mathbb{E}[y^1 \mid do(t=1)] - \mathbb{E}[y^0 \mid do(t=0)], \tag{5.1}$$

$$\text{CATE}(\mathbf{x}) := \mathbb{E}[y^1 \mid \mathbf{x}, do(t=1)] - \mathbb{E}[y^0 \mid \mathbf{x}, do(t=0)]. \tag{5.2}$$

We make some basic assumptions in this study: (i) stable unit treatment value: the outcome of each instance is not affected by the treatment assigned to other instances; (ii) unconfoundedness: the treatment assignment to an individual is independent of the outcome given hidden confounding variables; (iii) overlap: each individual has a positive probability of treatment assignment; (iv) smoothness: individuals who have similar hidden confounding variables have similar outcomes; (v) noisy proxy variables: hidden confounding variables can be recovered by noisy proxy variables.

5.4 Preliminaries

We briefly introduce some notable deep generative models based on VAE as preliminaries for clarity.

VAE [29] is a widely used deep generative model that sets a prior distribution as the normal distribution. it maximizes the ELBO, consisting of reconstruction loss and the Kullback-Leibler (KL) divergence loss. It usually parameterizes $p_{\theta_\mathbf{x}}$ and q_ϕ by neural networks.

$$p(\mathbf{z}_i) = \prod_{j=1}^{d_\mathbf{z}} \mathcal{N}(z_{ij} \mid 0, 1), p_{\theta_\mathbf{x}}(\mathbf{x}_i \mid \mathbf{z}_i) = \prod_{j=1}^{d_\mathbf{x}} p_{\theta_\mathbf{x}}(x_{ij} \mid \mathbf{z}_i), \tag{5.3}$$

$$\mathcal{L}_{\text{ELBO}} = \sum_{i=1}^{N} \mathbb{E}_{q_\phi(\mathbf{z}_i|\mathbf{x}_i)}[\log p_{\theta_\mathbf{x}}(\mathbf{x}_i \mid \mathbf{z}_i) + \log p(\mathbf{z_i}) - \log q_\phi(\mathbf{z}_i \mid \mathbf{x}_i)] \tag{5.4}$$

$$= \sum_{i=1}^{N} E_{q_\phi(\mathbf{z}_i|\mathbf{x}_i)}[\log p_{\theta_\mathbf{x}}(\mathbf{x}_i \mid \mathbf{z}_i) - \text{KL}(q_\phi(\mathbf{z}_i \mid \mathbf{x}_i), p(\mathbf{z}))]. \tag{5.5}$$

InfoVAE [34] is a VAE with a mutual information regularization term. The mutual information term boils down to the distribution divergence between the prior distribution and marginal distribution of posterior distribution, and the function to be optimized is written as:

$$\mathcal{L}_{\text{InfoVAE}} = \sum_{i=1}^{N} \mathbb{E}_{q_\phi(\mathbf{z}_i|\mathbf{x}_i)}[\log p_{\theta_\mathbf{x}}(\mathbf{x}_i \mid \mathbf{z}_i) - \text{KL}(q_\phi(\mathbf{z}_i \mid \mathbf{x}_i), p(\mathbf{z}))] - D(q_\phi(\mathbf{z}), p(\mathbf{z})), \tag{5.6}$$

where $D(q_\phi(\mathbf{z}), p(\mathbf{z}))$ is a divergence between the two distributions $p(\mathbf{z})$ and $q_\phi(\mathbf{z})$, and any divergence can be used given that $D(q_\phi(\mathbf{z}), p(\mathbf{z})) = 0$ if and only if $q_\phi(\mathbf{z}) = p(\mathbf{z})$ [34].

CEVAE [24] is a recently proposed VAE-based methods for CATE and ATE estimation, which aims to identify treatment effect under the presence of hidden confounding variables. To correctly specify treatment effect, we need to deal with hidden confounding variables. CEVAE assumes that such hidden confounding variables can be recovered from proxy variables as many previous studies. It takes inputs $\mathbf{x}_i, t_i, y_i^{t_i}$ to infer hidden confounding variables, $\mathbf{z}_i$.

$$p_{\theta_t}(t_i \mid \mathbf{z}_i) = \mathrm{Bern}(h(g(\mathbf{z}_i))), \tag{5.7}$$

$$p_{\theta_y}(y_i^{t_i} \mid \mathbf{z}_i, t_i) = \mathcal{N}(\mu = \hat{\mu}_i, \sigma^2 = 1), \hat{\mu}_i = t_i f_1(\mathbf{z}_i) + (1 - t_i) f_0(\mathbf{z}_i), \tag{5.8}$$

$$q_\phi(\mathbf{z}_i \mid \mathbf{x}_i, t_i, y_i^{t_i}) = \prod_{j=1}^{d_\mathbf{z}} \mathcal{N}(\mu_{ij} = \bar{\mu}_{ij}, \sigma_j^2 = \bar{\sigma}_{ij}^2), \tag{5.9}$$

$$\bar{\boldsymbol{\mu}}_i = t_i \bar{\boldsymbol{\mu}}_{t=0,i} + (1 - t_i)\bar{\boldsymbol{\mu}}_{t=1,i}, \bar{\boldsymbol{\sigma}}_i^2 = t_i \boldsymbol{\sigma}_{t=0,i}^2 + (1 - t_i)\boldsymbol{\sigma}_{t=1,i}^2, \tag{5.10}$$

$$\bar{\boldsymbol{\mu}}_{t=0,i}, \boldsymbol{\sigma}_{t=0,i}^2 = f_3 \circ f_2(\mathbf{x}_i, y_i), \bar{\boldsymbol{\mu}}_{t=1,i}, \boldsymbol{\sigma}_{t=1,i}^2 = f_4 \circ f_2(\mathbf{x}_i, y_i), \tag{5.11}$$

where $h(x)$ is a sigmoid function defined as $h(x) := \frac{1}{1+\exp^{-x}}$, $\mathrm{Bern}(p)$ denotes the Bernoulli distribution which returns 1 or 0 with a probability p or $1-p$, respectively, and g, f_0, f_1, f_2, f_3 and f_4 are neural networks. The variational lower bound is given as:

$$\begin{aligned} \mathcal{L}_{\mathrm{ELBO(CEVAE)}} = \sum_{i=i}^{N} \mathbb{E}_{q_\phi(\mathbf{z}_i \mid \mathbf{x}_i, t_i, y_i)}[&\log p_{\theta_{\mathbf{x},t}}(\mathbf{x}_i, t_i \mid \mathbf{z}_i) + \log p_{\theta_y}(y_i^{t_i} \mid t_i, \mathbf{z}_i) \\ &- \mathrm{KL}(q_\phi(\mathbf{z}_i \mid \mathbf{x}_i, t_i, y_i), p(\mathbf{z}))], \end{aligned} \tag{5.12}$$

where $\log p_{\theta_{\mathbf{x},t}}(\mathbf{x}_i, t_i \mid \mathbf{z}_i) = \log p_{\theta_\mathbf{x}}(\mathbf{x}_i \mid \mathbf{z}_i) + \log p_{\theta_t}(t_i \mid \mathbf{z}_i)$. To give outcomes for new individuals, CEAVE is required to have the treatment assignment and outcome beforehand. Therefore, it employs two auxiliary loss functions to deal with new individuals. Finally, the objective function of CEVAE is given as:

$$\mathcal{L}_{\mathrm{CEVAE}} = \mathcal{L}_{\mathrm{ELBO(CEVAE)}} + \sum_{i=i}^{N} \log q(t_i \mid \mathbf{x}_i) + \log q(y_i^{t_i} \mid \mathbf{x}_i, t_i). \tag{5.13}$$

5.5 CEVAE fails to estimate CATE

Treatment effect estimation with hidden confounding variables is an essential problem. CEVAE [24] enabled us to estimate treatment effect with hidden confounding variables without any strong assumption because VAE can recover a larger function class. Prior studies have made strong assumptions, such as on the properties of proxy variables and hidden confounding variables. CEVAE can identify CATE and ATE when it recovers the joint distribution $p(\mathbf{z}, \mathbf{x}, t, y)$.

Theorem 2. *We can recover CATE and ATE when we recover the joint distribution $p(\mathbf{z}, \mathbf{x}, t, y)$ in Fig (5.1)* [24].

Proof. The proof is completed by applying the rules of do-calculus to Fig. (5.1). See CEVAE paper for the details [24].

However, one of the major drawbacks of previous VAE-based methods, including CEVAE, is that they do not guarantee that they can recover the hidden confounding variables, even when when they achieve the optimal solution even though they have a capability to recover them. As a motivating example, we first note that there is a dataset for which the optimal solution of CEVAE does not give the correct CATE and ATE for new individuals. Note that we consider the case that we use only the proxy variables $\mathbf{x}$ because assuming that we have correct outcomes y for new individuals is not realistic.

Theorem 3. *Suppose we have a dataset $\mathcal{D} = \{\mathbf{x}_i, t_i, y_i^{t_i}\}_{i=1}^{N}$, where $\mathbf{z}_i \sim \mathcal{N}(0, 1)$, $\mathbf{x}_i \sim \mathcal{N}(\mathbf{z}_i, 1)$, $t_i \sim \mathrm{Bern}(\rho_{\mathrm{t}})$, $y_i \sim \mathcal{N}(\mathbb{I}(Cz_i > 0)t, 1)$, where ρ_t is a probability of of receiving treatment and C is a constant value. Suppose we only observe $\mathbf{x}_i = 1$ or $\mathbf{x}_i = -1$ and $y_i = 1$ or $y_i = -1$. The optimal solution of CEVAE for this dataset does not give correct CATE and ATE.*

Proof. Appendix.

This result demonstrates the insufficiency of naive VAE-based methods to recover hidden confounding variables and estimate treatment effect. Because there are numerous situations where observational data are limited and over-fitting to observational data may occur, we need to treat this problem carefully. Here we demonstrate a specific dataset, but we leave the proof of a more general form for future work.

5.6 InfoCEVAE with hidden confounding variables matching

The phenomenon described above arises because of the nature that VAE pushes masses away from each other and focuses on reconstruction loss too much. This becomes more crucial when we have higher dimensional proxy variables and a lower number of hidden confounding variables compared to proxy variables (i.e, $d_{\mathbf{x}} \gg d_{\mathbf{z}}$), especially when we have limited data. Some readers might think a larger number of proxy variables makes an unconfoundedness assumption, i.e., non-hidden confounding assumption, more reasonable; however, we usually can not guarantee that there are no hidden confounding variables in practice, and moreover, sometimes we never have access to the hidden confounding variables (e.g., variables including sensitive privacy information) even when we can easily obtain some proxy variables.

The straightforward solution to obtain the correct ATE using VAE-based methods is to employ the theoretical analysis of InfoVAE [34], which adds the mutual information regularization term to the original ELBO of VAE.

The ELBO of InfoCEVAE will be adding the information regularization term to CEVAE given as:

$$\begin{aligned}\mathcal{L} = \sum_{i=i}^{N} \mathbb{E}_{q_\phi(\mathbf{z}_i|\mathbf{x}_i,t_i,y_i)}[\log p_{\theta_{\mathbf{x},t}}(\mathbf{x}_i, t_i \mid \mathbf{z}_i) + \log p(y_i^{t_i} \mid t_i, \mathbf{z}_i) \\ -\mathrm{KL}(q_\phi(\mathbf{z}_i \mid \mathbf{x}_i, t_i, y_i^{t_i}), p(\mathbf{z}))] - D(q_\phi(\mathbf{z}), p(\mathbf{z})).\end{aligned} \tag{5.14}$$

We can employ the several measures of divergence D between two probability distributions, such as 2-Wasserstein distance given that $D(q(\mathbf{z}), p(\mathbf{z})) = 0$ if and only if $q(\mathbf{z}) = p(\mathbf{z})$. We use the 2-Wasserstein distance as D, and the 2-Wasserstein distance for two Gaussian distributions is written as:

$$D(\mathcal{N}(\mu_1, \sigma_1), \mathcal{N}(\mu_2, \sigma_2)) = \|\mu_1 - \mu_2\|^2 + \|\sigma_1 - \sigma_2\|^2. \tag{5.15}$$

We can also get correct CATE and ATE when the model achieves the optimal solution of the objective function $q_\phi(\mathbf{z}) = p(\mathbf{z})$.

Theorem 4. *The optimal solution of InfoCEVAE gives the correct CATE and ATE.*

Proof. According to the Proposition of InfoVAE, we obtain the optimal solution when we achieve $q_\phi(y \mid t, \mathbf{z}) = p(y \mid t, \mathbf{z})$ and $q_\phi(\mathbf{z} \mid \mathbf{x}, t, y) = p(\mathbf{z} \mid \mathbf{x}, t, y)$. Therefore,

$$\widehat{CATE}(\mathbf{x}) = p_\theta(y \mid t = 1, \mathbf{x}) - p_\theta(y \mid t = 0, \mathbf{x}) \tag{5.16}$$

$$= \int_{\mathcal{Z}} p_\theta(y = 1 \mid t = 1, \mathbf{z}) q_\phi(\mathbf{z} \mid \mathbf{x}, t = 0, y) \tag{5.17}$$

$$- p_\theta(y = 1 \mid t = 0, \mathbf{z}) q_\phi(\mathbf{z} \mid \mathbf{x}, t = 1, y) dz \tag{5.18}$$

$$= \int_{\mathcal{Z}} p(y \mid t = 1, \mathbf{z}) p(\mathbf{z} \mid \mathbf{x}, t = 0, y) - p(y \mid t = 0, \mathbf{z}) p(\mathbf{z} \mid \mathbf{x}, t = 1, y) dz \tag{5.19}$$

$$= \int_{\mathcal{Z}} p(y \mid do(t = 1), \mathbf{z}) p(\mathbf{z} \mid \mathbf{x}, do(t = 0), y) \tag{5.20}$$

$$- p(y \mid do(t = 0), \mathbf{z}) p(\mathbf{z} \mid \mathbf{x}, do(t = 1), y) dz \tag{5.21}$$

$$= p(y \mid \mathbf{x}, do(t = 1)) - p(y \mid \mathbf{x}, do(t = 0)) \tag{5.22}$$

$$= CATE(\mathbf{x}). \tag{5.23}$$

However, this naive approach requires that we obtain the correct outcome function, i.e., $p(y \mid \mathbf{z}, t) = p_\theta(y \mid \mathbf{z}, t)$ as well as the propensity score function $p(t \mid \mathbf{z})$. Obtaining the correct outcome function is a challenging, especially when we need to consider observational bias. Say we obtain $q_\phi(\mathbf{z}) = p(\mathbf{z})$ once, and then our goal is to recover the joint distribution $\int_z q_\phi(\mathbf{z}, \mathbf{x}, t, y) dz = p(\mathbf{x}, t, y)$. Therefore we need to ensure that we have $q(\mathbf{x}, t, y \mid \mathbf{z}) = p(\mathbf{x}, t, y \mid \mathbf{z})$. Hence, to achieve the optimal solution of InfoCEVAE, we need to learn θ such that $p_\theta(\mathbf{x}, t, y \mid \mathbf{z}) = p(\mathbf{x}, t, y \mid \mathbf{z})$, which means we need to learn the correct outcome function only by skewed observational data. This is almost impossible without modification. The estimator θ_y given observational data is given as:

$$\begin{aligned} \theta_y^{obs} &= \text{argmin}_{\theta_y \in \Theta} - \frac{1}{N} \sum_{i=1}^{N} \mathbb{E}_{q_\phi(\mathbf{z}_i | \mathbf{x}_i, t_i, y_i)} [\log p_{\theta_y}(y_i \mid t_i, \mathbf{z}_i)] \qquad (5.24) \\ &\simeq \text{argmin}_{\theta_y \in \Theta} - \mathbb{E}_{p_{\mathcal{D}_{\text{train}}}(t,y)} [\mathbb{E}_{q_\phi(\mathbf{z}_i | \mathbf{x}, t_i, y_i)} [\log p_{\theta_y}(y_i \mid t_i, \mathbf{z}_i)]]. \qquad (5.25) \end{aligned}$$

However, this estimator is not consistent because of observational bias caused by hidden confounding variables.

$$\begin{aligned} \lim_{N \to \infty} \theta_y^{obs} &= \text{argmin}_{\theta_y \in \Theta} - \mathbb{E}_{p_{\mathcal{D}_{\text{train}}}(t,y)} [\mathbb{E}_{q_\phi(\mathbf{z} | \mathbf{x}, t, y)} [\log p_{\theta_y}(y_i \mid t_i, \mathbf{z}_i)]] \qquad (5.26) \\ &\neq \text{argmin}_{\theta_y \in \Theta} - \mathbb{E}_{p(t)p(y)} [\mathbb{E}_{q_\phi(\mathbf{z} | \mathbf{x}, t, y)} [\log p_{\theta_y}(y_i \mid t_i, \mathbf{z}_i)]]. \qquad (5.27) \\ & \qquad (5.28) \end{aligned}$$

$$\begin{aligned} \because p_{\mathcal{D}_{\text{train}}}(t, y) = \int_{\mathcal{Z}} p(y \mid t, \mathbf{z}) p(t \mid \mathbf{z}) p(\mathbf{z}) d\mathbf{z} &\neq \int_{\mathcal{Z}} p(y \mid t, \mathbf{z}) p(t) p(\mathbf{z}) d\mathbf{z} \qquad (5.29) \\ &= p(t)p(y). \qquad (5.30) \end{aligned}$$

Note that we assume the treatment assignment is randomized when evaluating the model. To resolve this problem, we propose an effective algorithm based on latent variables and a matching algorithm. Note that InfoCEVAE guarantees the correct treatment effect when it achieves the optimal solution, although it is challenging to obtain. However, CEVAE cannot provide the optimal treatment effect, even when it achieves the optimal solution.

5.6.1 Hidden confounding variables matching

To mitigate the above issue, we aim to recover hidden confounding variables by only proxy variables, not using outcomes like CEVAE. This approach sounds reasonable because the assumption that we can recover hidden confounding variables only by proxy variables when we have such high dimensional proxy variables is quite valid [28]. Moreover, the advantage of using only proxy variables is that we do not need to predict outcomes for new individuals. Hence, hidden confounding variables are inferred as:

$$q_\phi(\mathbf{z}_i \mid \mathbf{x}_i) = \prod_{j=1}^{d_\mathbf{z}} \mathcal{N}(\mu = \mu_{ij}, \sigma = \sigma_{ij}); p(\mathbf{z}_i) = \mathcal{N}(0, 1). \tag{5.31}$$

The ELBO is given as:

$$\begin{aligned}\mathcal{L}_{\text{InfoCEVAE}} = &\sum_{i=i}^{N} \mathbb{E}_{q_\phi(\mathbf{z}_i|\mathbf{x}_i)}[\log p_{\theta_{\mathbf{x},t}}(\mathbf{x}_i, t_i \mid \mathbf{z}_i) + \log p_{\theta_y}(y_i \mid t_i, \mathbf{z}_i) \\ &- \mathrm{KL}(\log q_\phi(\mathbf{z}_i \mid \mathbf{x}_i), p(\mathbf{z}))] - \lambda D(q_\phi(\mathbf{z}), p(\mathbf{z})),\end{aligned} \tag{5.32}$$

where λ is a hyper-parameter that controls the strength of regularization.

For bias mitigation, we propose latent variable matching, which makes use of latent variables to match individuals. Thanks to the theoretical advantage of InfoCEVAE, we can find the matching based on the some metric using latent variables. By nearest neighbor matching, we construct the counterfactual outcome for each individual i as:

$$\hat{y}_i^{\bar{t}_i} = \frac{1}{k} \sum_{j \in \mathrm{NN}(\mathbf{z}_i, k)} y_j^{t_j}, \tag{5.33}$$

where $\mathrm{NN}(\mathbf{z}_i, k) = \{i_1, \dots, i_k\}$ is a set of indices ordered by a similarity that defines nearest neighbors of $\mathbf{z}_i$, and $\bar{t}_i \in \mathcal{T}$ represents the other treatment of t_i. Here, we consider two variants of nearest neighbor selection: (i) Euclidean distance of means of the two latent variables: (ii) propensity score matching. The advantage of (i) is that we can use all the information of latent variables and does not need to infer propensity score, while (i) might fail to find good matching in higher dimensions of latent variables. The pros and cons of (ii) are the opposite of those of (i). Note that under the smoothness assumption and when we achieve the optimal solution of InfoCEVAE, both hidden confounding variable matching methods yield consistency estimators. We compute the log-likelihood of counterfactual outcome as:

$$\mathcal{L}_{\text{cf}} = \sum_{i=i}^{N} \mathbb{E}_{q_{\phi}(\mathbf{z}_i|\mathbf{x}_i)}[\log p(\hat{y}_i^{\bar{t}_i} \mid \bar{t}_i, \mathbf{z}_i)]. \tag{5.34}$$

Finally, the objective function to be optimized is given as:

$$\mathcal{L} = \mathcal{L}_{\text{cf}} + \mathcal{L}_{\text{InfoCEVAE}}. \tag{5.35}$$

Theorem 5. *The optimal solution of InfoCEVAE with hidden variables matching gives the consistent treatment effect estimator under the smoothness assumption.*

Proof. According to the theorem of InfoVAE, we can obtain the correct posterior function when we obtain the optimal solution. Using correct hidden confounding variables, we can obtain correct counterfactual outcomes under the smoothness assumption. Using the correct counterfactual outcomes as well as factual outcomes, we can obtain a consistent estimator, which yields the correct ATE.

5.7 Experiments

We validated the performance of the proposed method, especially when there are hidden confounding variables. First, we introduce the datasets used in the experiments, and detail the experimental settings.

5.7.1 Datasets

We rarely have real-world datasets due to the counterfactual nature of treatment effect estimation problem. We employed a widely-used semi-synthetic dataset and a synthetic dataset for this experiment.

News dataset [17].

This is a dataset including opinions of media consumers for news articles [17][1]. It contains 5,000 news articles and outcomes generated from the NY Times corpus[2]. Each article is consumed on desktop ($t = 0$) or mobile ($t = 1$), and it is assumed that media consumers prefer to read some articles on mobile than desktop. We use the News dataset by setting the scale parameter for outcome in previous research [17] as 200. Each article is generated by a topic model and represented in the bag-of-words representation. The size of the vocabulary is 3,477. As preprocessing, we apply principal component analysis (PCA) with $d_{\mathbf{z}} = 30$. To simulate hidden confounding variables situation, we generate proxy variables using these variables after PCA. More concretely, we treat these variables as hidden confounding variables z_{ij} and generate proxy variables as

$$x_{i,j\times 1,\dots,j\times d_{proxy}} \sim \mathcal{N}(z_{ij}, \sigma_{\mathbf{z}}^2), \tag{5.36}$$

$$\mathbf{x}_i = [x_{i,1}, \dots, x_{i,30\times d_{\mathrm{proxy}}}], \tag{5.37}$$

where $\sigma_{\mathbf{z}}$ is a standard deviation of the entire variables after PCA , d_{proxy} stands out for the number of proxy variables per hidden confounding variables and [] represents the concatenation. We set d_{proxy} as 30 for the News dataset.

Synthetic dataset.

The synthetic dataset is a benchmark generated in this study. This dataset includes $5,000$ individuals, binary treatment, and continuous outcomes. We generated the dataset according to the following procedure:

[1] `https://www.fredjo.com/`

[2] `https://archive.ics.uci.edu/ml/datasets/Bag+of+Words`

$$z_{ij} \sim \mathcal{N}(0,1) \quad (j = 1, \dots, 5), \tag{5.38}$$

$$x_{i,j\times 1,\dots,j\times d_{\text{proxy}}} \sim \mathcal{N}(10z_{ij}, 1), \tag{5.39}$$

$$\mathbf{x}_i = [x_{i,1}, \dots, x_{i,5\times d_{\text{proxy}}}], \tag{5.40}$$

$$t_i \sim \text{Bern}(\alpha h(\sum_{j=1}^{5} z_{ij})), y_i \sim \mathcal{N}(3\mathbb{I}(\sum_{j=1}^{5} z_{ij} \geq 0) \times t_i + 5t_i, 1), \tag{5.41}$$

where $\alpha \geq 0$ is a variable that controls the strength of observational bias, and $\mathbb{I}(x)$ is an indicator function that is 1 if x is True and 0 otherwise. Note again that h is a sigmoid function. Larger α means we have severer observational bias, and setting α as 0 represents a randomized controlled trial. We clamped the treatment assignment probability at 0.01 and 0.99. We change d_{proxy} as ranging from 10 to 500 for the Synthetic dataset. Unless otherwise stated, we report the results when $d_{\text{proxy}} = 500$. In the experiments, we investigated the robustness against the bias strength by changing the value of α.

5.7.2 Experimental settings

We split the all individuals into 20%, 40%, and 40% train, validation, and test data, respectively. Note that we especially focus on the case when train data are limited because over-estimation becomes severer. As base neural network models including VAE-based methods, we use two-layer neural networks. We also set the number of neurons (i.e, the number of representations) as 50 for TARNet and CFR. We use the *elu* function [113] as the activation function for all neural networks.

As evaluation metrics, we employ *ATE error* defined as

$$\epsilon_{\text{ATE}} = \frac{1}{N} \sum_{i=1}^{N} |(y_i^1 - y_i^0 - (\hat{y}_i^1 - \hat{y}_i^0))|,$$

and *precision in estimation of heterogeneous effect (PEHE)* used in previous researches [15, 17]. ϵ_{PEHE} is the estimation error of individual treatment effects and is defined as

$$\sqrt{\epsilon_{\text{PEHE}}} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i^1 - y_i^0 - (\hat{y}_i^1 - \hat{y}_i^0))^2}.$$

The hyper-parameters are tuned based on the prediction loss using the observed outcomes on the validation data. We log-uniform randomly choose the hyper-paramters λ ranging from $1e-3$ to $1e3$ ten times, and the final hyper-parameter is selected based on the prediction loss using the outcomes on the validation data. For CEVAE, we compute the ELBO using validation data and use the model at the epoch when the ELBO for validation data achieves the maximum value. We report the average results of 10 trials on the Synthetic dataset and 20 trials on the News dataset.

5.7.3 Baseline methods

We compare the proposed method with the following baseline methods including VAE-based methods. Unless otherwise stated, we use the concatenation of proxy variables and treatment assignment coded as a one-hot vector as the input of predictive models of (i) and (ii). (i) Ridge is the ordinary linear regression methods with L2 regularization. (ii) Random forest (RF) [14] and BART [59, 15] are the predictive models based on the decision tree. (iii) TARNet [18] is a deep neural network model that has shared layers for representation learning and different layers for outcome prediction for treatment and control instances. Counterfactual regression (CFR) [18] is a state-of-the-art deep neural network model based on balanced representations between treatment and control instances. We use the Wasserstein distance. (vi) CEVAE [24] is a VAE-based treatment effect estimation method.

5.7.4 Results

We first assess the full results in comparison with the baseline methods, and then we investigate how the performance changes as we change the size of proxy variables or the strength of observational bias. Table 5.1 gives a performance comparison of the proposed method with the baseline methods. Overall, the proposed method outperforms baseline methods significantly. On the News dataset, the both approaches of proposed method show significant improvement from the baseline methods. On the Synthetic dataset, the proposed method with propensity score matching works better. This result makes sense because the propensity score and outcome have strong correlation in this dataset. However, the proposed method with the Euclidean matching does not work because nearby individuals in terms of the Euclidean distance of hidden confounding variables do not necessarily become the good matching unless we have a large amount of individuals. Meanwhile the predictive performance deteriorates as selection bias becomes stronger, the proposed method shows robustness to selection bias and consistently outperforms the baseline methods. Fig 5.2 and 5.3 demonstrate the change of predictive performances as we change the strength of bias α and the number of proxy variables d_{proxy}. Whereas the baseline methods suffer observational bias, the proposed method show robustness to it. Although, the baseline methods result in limited improvement, the proposed method also can deal with and make use of high dimensional proxy variables and improve its predictive performance.

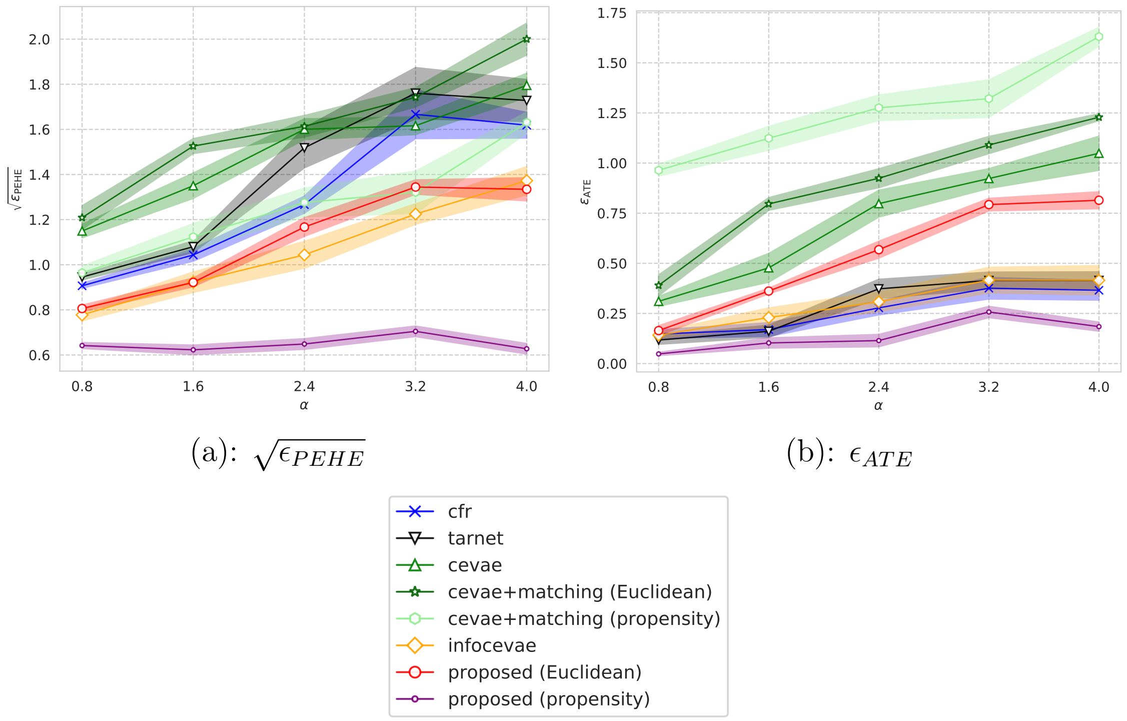
5.8 Conclusion

In this study, we considered treatment effect estimation problem with hidden confounding variables using VAE. VAE has been used to recover hidden confounding variables by making use of its large capability. We first pointed out that the optimal solution of CEVAE is not the correct ATE. We propose an efficient algorithm

Table 5.1: Performance comparison on the News dataset and the Synthetic dataset in terms of PEHE and ATE. Lower is better. † indicates that the proposed method show statistically significantly better result by the paired t-test with p $<$ 0.05. Bold results show the best results in term of average. We also show standard errors for 20 and 10 times repeated experiments for the News dataset and the Synthetic dataset, respectively.

	News		**Synthetic**	
Method	$\sqrt{\epsilon_{\text{PEHE}}}$	ϵ_{ATE}	$\sqrt{\epsilon_{\text{PEHE}}}$	ϵ_{ATE}
Mean	$^\dagger 14.325_{\pm 0.128}$	$^\dagger 3.921_{\pm 0.551}$	$^\dagger 1.980_{\pm 0.010}$	$^\dagger 1.292_{\pm 0.015}$
Ridge	$^\dagger 13.764_{\pm 0.959}$	$0.911_{\pm 0.190}$	$^\dagger 1.570_{\pm 0.019}$	$^\dagger 0.438_{\pm 0.061}$
RF	$^\dagger 10.246_{\pm 0.959}$	$^\dagger 2.211_{\pm 0.385}$	$^\dagger 1.465_{\pm 0.021}$	$^\dagger 0.854_{\pm 0.024}$
BART	$^\dagger 13.618_{\pm 0.921}$	$^\dagger 1.310_{\pm 0.221}$	$^\dagger 2.758_{\pm 0.332}$	$^\dagger 1.829_{\pm 0.332}$
TARNet	$^\dagger 8.988_{\pm 0.488}$	$^\dagger 1.135_{\pm 0.200}$	$^\dagger 1.729_{\pm 0.093}$	$^\dagger 0.415_{\pm 0.043}$
CFR	$^\dagger 9.125_{\pm 0.488}$	$^\dagger 1.643_{\pm 0.268}$	$^\dagger 1.619_{\pm 0.057}$	$^\dagger 0.366_{\pm 0.049}$
CEVAE	$^\dagger 9.389_{\pm 0.600}$	$^\dagger 2.319_{\pm 0.381}$	$^\dagger 1.795_{\pm 0.053}$	$^\dagger 1.048_{\pm 0.085}$
CEVAE w/ Euclidean	$^\dagger 8.659_{\pm 0.524}$	$^\dagger 1.196_{\pm 0.250}$	$^\dagger 2.000_{\pm 0.053}$	$^\dagger 1.229_{\pm 0.017}$
CEVAE w/ propensity	$^\dagger 8.642_{\pm 0.523}$	$^\dagger 1.136_{\pm 0.254}$	$^\dagger 1.630_{\pm 0.046}$	$^\dagger 0.683_{\pm 0.013}$
InfoCEVAE	$^\dagger 8.453_{\pm 0.510}$	$^\dagger 1.742_{\pm 0.242}$	$^\dagger 1.373_{\pm 0.062}$	$^\dagger 0.415_{\pm 0.073}$
InfoCEVAE w/ Euclidean	$\mathbf{7.934_{\pm 0.478}}$	$0.928_{\pm 0.172}$	$^\dagger 1.334_{\pm 0.032}$	$^\dagger 0.815_{\pm 0.042}$
InfoCEVAE w/ propensity	$\mathbf{7.930_{\pm 0.476}}$	$\mathbf{0.835_{\pm 0.147}}$	$\mathbf{0.626_{\pm 0.023}}$	$\mathbf{0.184_{\pm 0.022}}$

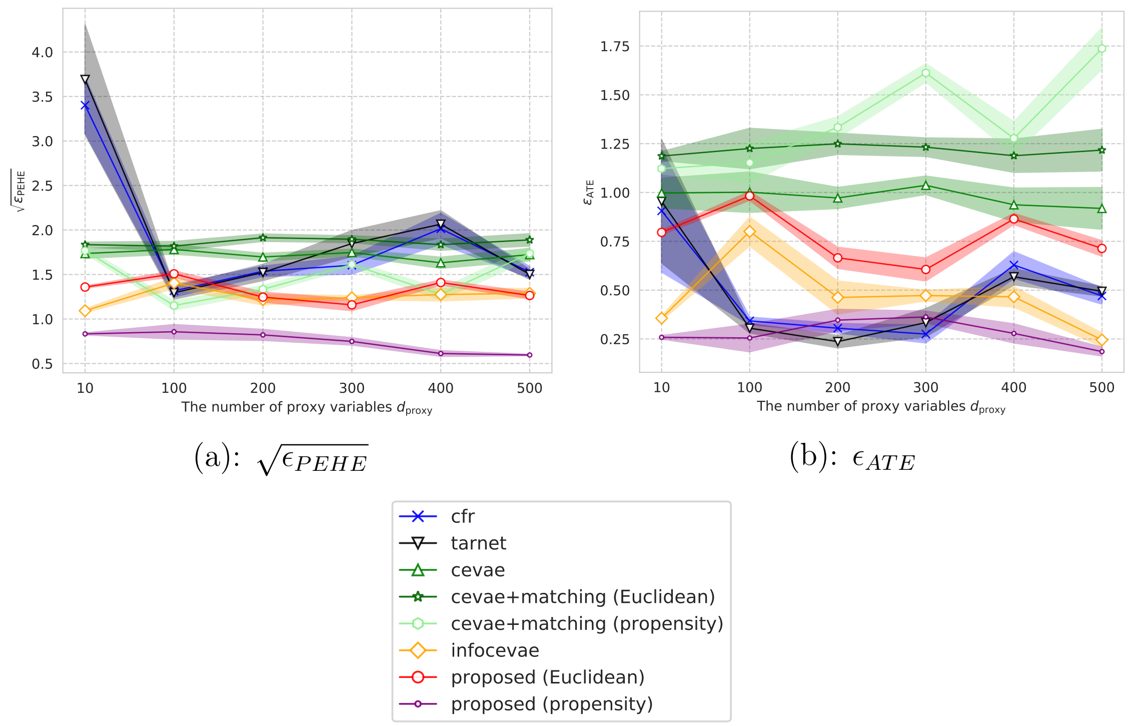
to recover hidden confounding variables and estimate treatment effect making use of mutual information and matching techniques. Experiments on semi-synthetic and synthetic datasets demonstrate the effectiveness of the proposed method, especially when we have higher dimensional proxy variables but still hidden confounding variables.



(a): $\sqrt{\epsilon_{PEHE}}$

(b): ϵ_{ATE}

Figure 5.2: Performance comparison as the change of observational bias α. Lower is better. Whereas baseline methods suffered a observational bias and get degrade its performance, the proposed method demonstrates its robustness to the observational bias and almost entirely surpass the baseline methods in the both metrics. Especially, the proposed method consistently shows the affordable performance in ATE.



(a): $\sqrt{\epsilon_{PEHE}}$

(b): ϵ_{ATE}

Figure 5.3: Performance comparison as the change of the number of proxy variables. Lower is better. While the baseline methods do not improve their predictive performances as the number of proxy variables increase, the proposed method with propensity score matching achieves almost entirely the best results, especially significant in $\sqrt{\epsilon_{\mathrm{PEHE}}}$.

Appendix

Proof of Theorem 3.

Theorem 3. *Suppose we have a dataset* $\mathcal{D} = \{\mathbf{x}_i, t_i, y_i^{t_i}\}_{i=1}^N$*, where* $\mathbf{z}_i \sim \mathcal{N}(0, 1)$*,* $\mathbf{x}_i \sim \mathcal{N}(\mathbf{z}_i, 1)$*,* $t_i \sim \mathrm{Bern}(\rho_\mathrm{t})$*,* $y_i \sim \mathcal{N}(\mathbb{I}(Cz_i > 0)t, 1)$*, where* ρ_t *is a probability of of receiving treatment and* C *is a constant value. Suppose we only observe* $\mathbf{x}_i = 1$ *or* $\mathbf{x}_i = -1$ *and* $y_i = 1$ *or* $y_i = -1$*. The optimal solution of CEVAE for this dataset does not give correct CATE and ATE.*

Proof. Note that $\mathbf{x}$ and $\mathbf{z}$ represent vectors in the main paper but they are also scalar values in this proof. The ATE of this dataset is

$$\mathbb{E}[y^1] - \mathbb{E}[y^0] = p(\mathbf{z}_i \geq 0)\mathrm{C} - 0 \tag{5.42}$$

$$= p(\mathbf{z}_i \geq 0)\mathrm{C}. \tag{5.43}$$

We first show naive CEVAE loss has unbounded reward if the proxy variables come from gaussian distribution family. This step mainly follows the same procedure as InfoVAE [34]. We consider the following restricted a Gaussian models and if we achieve the infinite ELBO in this model, we can achieve the infinite ELBO in any model with more expresiveness than this model.

$$p(\mathbf{x} \mid \mathbf{z}) = \begin{cases} \mathcal{N}(1, \sigma^2) & (\mathbf{z} \geq 0) \\ \mathcal{N}(-1, \sigma^2) & (\mathbf{z} < 0) \end{cases},$$

$$q(\mathbf{z} \mid \mathbf{x}) = \begin{cases} \mathcal{N}(a, \sigma_q^2) & (\mathbf{x} \geq 0) \\ \mathcal{N}(-a, \sigma_q^2) & (\mathbf{x} < 0) \end{cases},$$

$$p(t \mid \mathbf{z}) = \begin{cases} p_1 & (\mathbf{z} \geq 0) \\ p_0 & (\mathbf{z} < 0) \end{cases}, \; p(y \mid \mathbf{z}) = \begin{cases} \mathcal{N}(\mathrm{C}, 1) & (\mathbf{z} \geq 0, t = 1) \\ \mathcal{N}(0, 1) & (\mathbf{z} < 0, t = 1) \\ \mathcal{N}(0, 1) & (t = 0). \end{cases}$$

The ELBO for $\mathbf{x} = 1$ is

$$\begin{aligned} \mathcal{L}_{AE}(\mathbf{x} = 1) \equiv & \mathbb{E}_{q(\mathbf{z}|\mathbf{x}=1)}[\log p(\mathbf{x} = 1 \mid \mathbf{z})] + \mathbb{E}_{q(\mathbf{z}|\mathbf{x}=1)}[\log p(\mathbf{x} = 1 \mid \mathbf{z})] & (5.44) \\ & + \mathbb{E}_{q(\mathbf{z}|\mathbf{x}=1)}[\log p(t \mid \mathbf{z})] + \mathbb{E}_{q(\mathbf{z}|\mathbf{x}=1)}[\log p(y \mid \mathbf{z}, t)]. & (5.45) \end{aligned}$$

Taking the gradient of $\mathcal{L}_{AE}(\mathbf{x} = 1)$,

$$\frac{\partial \mathcal{L}_{AE}(\mathbf{x} = 1)}{\sigma} = -\frac{1}{\sigma} + \frac{4}{\sigma^3} q(\mathbf{z} \leq 0 \mid \mathbf{x} = 1) = 0, \tag{5.46}$$

and the optimal solution for $\mathcal{L}_{AE}(\mathbf{x} = 1)$ is achieved when $\sigma = 2\sqrt{q(\mathbf{z} \leq 0 \mid \mathbf{x} = 1)}$. Therefore,

$$\mathcal{L}^*_{AE}(x = 1) = -\frac{1}{2} \log q(\mathbf{z} \leq 0 \mid \mathbf{x} = 1) + \text{Constant}. \tag{5.47}$$

$q(\mathbf{z} \leq 0 \mid \mathbf{x} = 1)$ is the sum of Gaussian tail probabilities. Hence in the limit $\sigma_q \to 0$, $a \to \infty$,

$$\mathcal{L}^*_{AE}(\mathbf{x} = 1) = \Theta(\frac{a^2}{\sigma_q^2}), \tag{5.48}$$

$$\begin{aligned} \mathcal{L}_{REG} &= -\mathrm{KL}(q_\phi(\mathbf{z} \mid \mathbf{x} = 1) || p(\mathbf{z})) & (5.49) \\ &= \log \sigma_q - \frac{\sigma_q^2}{2} - \frac{a^2}{2} + \frac{1}{2}. & (5.50) \end{aligned}$$

Therefore, we can achieve unbounded ELBO.

$$\lim_{\sigma_q \to 0, a \to \infty} \mathcal{L}_{\text{ELBO}}(\mathbf{x}=1) = \lim_{\sigma_q \to 0, a \to \infty} \mathcal{L}^*_{\text{AE}}(\mathbf{x}=1) + \mathcal{L}_{\text{REG}}(\mathbf{x}=1) \tag{5.51}$$

$$\to \infty. \tag{5.52}$$

Next, we show that treating them as normal confounding variables will not give the correct treatment effect.

$$\mathbb{E}[y^1 \mid t=1] = \int_{\mathcal{X}} p(y \mid t=1, \mathbf{x}) p(\mathbf{x}) d\mathbf{x} \tag{5.53}$$

$$= \int_{\mathcal{X}} \frac{p(t=1, \mathbf{x} \mid y) p(y)}{p(t=1, \mathbf{x})} p(\mathbf{x}) d\mathbf{x} \tag{5.54}$$

$$= \int_{\mathcal{X}} \frac{\int_{\mathcal{Z}} p(y, t=1, \mathbf{x} \mid \mathbf{z}) p(\mathbf{z}) d\mathbf{z}}{\int_{\mathcal{Z}} p(t=1, \mathbf{x} \mid \mathbf{z}) p(\mathbf{z}) d\mathbf{z}} p(\mathbf{x}) d\mathbf{x} \tag{5.55}$$

$$= \int_{\mathcal{X}} \frac{\int_{\mathcal{Z}} p(y, t=1, \mathbf{x} \mid \mathbf{z} \geq 0) p(\mathbf{z} \geq 0) d\mathbf{z} + \int_{\mathcal{Z}} p(y, t=1, \mathbf{x} \mid \mathbf{z} < 0) p(\mathbf{z} < 0) d\mathbf{z}}{\int_{\mathcal{Z}} p(t=1, \mathbf{x} \mid \mathbf{z} \geq 0) p(\mathbf{z} \geq 0) d\mathbf{z} + \int_{\mathcal{Z}} p(t=1, \mathbf{x} \mid \mathbf{z} < 0) p(\mathbf{z} < 0) d\mathbf{z}} p(\mathbf{x}) d\mathbf{x} \tag{5.56}$$

$$= \int_{\mathcal{X}} \frac{\int_{\mathcal{Z}} p(y, t=1, \mathbf{x} \mid \mathbf{z} \geq 0) p(\mathbf{z} \geq 0) d\mathbf{z}}{\int_{\mathcal{Z}} p(t=1, \mathbf{x} \mid \mathbf{z} \geq 0) p(\mathbf{z} \geq 0) d\mathbf{z} + \int_{\mathcal{Z}} p(t=1, \mathbf{x} \mid \mathbf{z} < 0) p(\mathbf{z} < 0) d\mathbf{z}} p(\mathbf{x}) d\mathbf{x} \tag{5.57}$$

$$= \int_{\mathcal{Z}} \frac{\rho_t \mathrm{C} p(\mathbf{x} \mid \mathbf{z} \geq 0) + \rho'_t p(\mathbf{x} \mid \mathbf{z} < 0) 0}{\rho_t p(\mathbf{x} \mid \mathbf{z} \geq 0) + \rho'_t p(\mathbf{x} \mid z < 0)} p(\mathbf{x}) d\mathbf{x} \tag{5.58}$$

$$= \int_{\mathcal{X}} \frac{\rho_t \mathrm{C} p(\mathbf{x} \mid \mathbf{z} \geq 0)}{\rho_t p(\mathbf{x} \mid \mathbf{z} \geq 0) + \rho'_t p(\mathbf{x} \mid \mathbf{z} < 0)} p(\mathbf{x}) d\mathbf{x}. \tag{5.59}$$

One case where this procedure gives the correct estimand is the case when the treatment assignment is randomized, i.e., $\rho_t = \rho'$.

$$\mathbb{E}[\hat{y}^1 \mid t=1] = \int_{\mathcal{X}} \frac{\rho_t \mathrm{C} p(\mathbf{x} \mid \mathbf{z} \geq 0)}{\rho_t p(\mathbf{x} \mid \mathbf{z} \geq 0) + \rho_t' p(\mathbf{x} \mid \mathbf{z} < 0)} p(\mathbf{x}) d\mathbf{x} \tag{5.60}$$

$$= \int_{\mathcal{X}} \frac{\mathrm{C} p(\mathbf{x} \mid \mathbf{z} \geq 0)}{p(\mathbf{x} \mid \mathbf{z} \geq 0) + p(\mathbf{x} \mid \mathbf{z} < 0)} p(\mathbf{x}) d\mathbf{x} \tag{5.61}$$

$$= \mathrm{C} p(\mathbf{x} \mid \mathbf{z} \geq 0). \tag{5.62}$$

Next, we try to estimate treatment effect using CEVAE and prove the estimand is wrong even if we obtain the correct outcome function.

$$\mathbb{E}[\hat{y}^1 \mid t=1] = \int_{\mathcal{X}} \int_{\mathcal{Z}} p(y \mid t=1, \mathbf{z}) p(\mathbf{z} \mid \mathbf{x}) p(\mathbf{x}) d\mathbf{z} d\mathbf{x} \tag{5.63}$$

$$= \frac{1}{2} \int_{\mathcal{Z}} p(y \mid t=1, \mathbf{z}) p(\mathbf{z} \mid \mathbf{x}=1) d\mathbf{z} \tag{5.64}$$

$$+ \frac{1}{2} \int_{\mathcal{Z}} p(y \mid t=1, \mathbf{z}) p(\mathbf{z} \mid \mathbf{x}=-1) d\mathbf{z} \tag{5.65}$$

$$= \frac{1}{2} \mathrm{C} p(\mathbf{z}_i \geq 0 \mid \mathbf{x}=1) + \frac{1}{2} \mathrm{C} p(\mathbf{z}_i \geq 0 \mid \mathbf{x}=-1) \tag{5.66}$$

$$\simeq \frac{1}{2} \mathrm{C}. \tag{5.67}$$

$$\mathbb{E}[\hat{y}^0 \mid t=0] = \int_{\mathcal{X}} \int_{\mathcal{Z}} p(y \mid t=0, \mathbf{z}) p(\mathbf{z} \mid \mathbf{x}) p(\mathbf{x}) d\mathbf{z} d\mathbf{x} \tag{5.68}$$

$$= \frac{1}{2} \int_{\mathcal{Z}} p(y \mid t=0, \mathbf{z}) p(\mathbf{z} \mid \mathbf{x}=1) d\mathbf{z} \tag{5.69}$$

$$+ \frac{1}{2} \int_{\mathcal{Z}} p(y \mid t=0, \mathbf{z}) p(\mathbf{z} \mid \mathbf{x}=-1) d\mathbf{z} \tag{5.70}$$

$$= \frac{1}{2} \mathrm{C} p(\mathbf{z}_i \geq 0 \mid \mathbf{x}=1) + \frac{1}{2} \mathrm{C} p(\mathbf{z}_i \geq 0 \mid \mathbf{x}=-1) \tag{5.71}$$

$$\simeq \frac{1}{2} \mathrm{C} \neq 0. \tag{5.72}$$

$$\widehat{\mathrm{ATE}} = \mathbb{E}[\hat{y} \mid t = 1] - \mathbb{E}[\hat{y} \mid t = 0] = 0 \tag{5.73}$$

$$\neq \mathrm{C}p(\mathbf{z}_i \geq 0) = \mathrm{ATE}. \tag{5.74}$$

Chapter 6

Conclusion and Future Directions

6.1 Conclusion

In this thesis, we addressed treatment effect estimation problems from small observational data. In Chapter 1, we introduced treatment effect estimation and its social impact and applications in the real world. In this thesis, we primarily focused on the following challenges:

- **Data scarcity**
- **Observational bias**
- **Hidden confounding variables**

We provided preliminaries in Chapter 2 including the two main frameworks: (i) the potential outcome framework [11, 35, 30] and (ii) the SCMs framework [21, 22]. In the remaining chapters, we discussed several approaches to overcome these difficulties. In Chapter 3, we introduced CP leverages both labeled and unlabeled data. This is particularly relevant in scenarios where obtaining labeled data is prohibitively expensive, whereas unlabeled data are relatively more accessible. Our method combines classical matching techniques for treatment effect estimation with the popular machine learning technique of label propagation. We presented a salient example to illustrate the benefits of incorporating unlabeled data in estimation. Whilst the existing methods failed to predict treatment effects crucially for some individuals, the proposed method successfully predicted treatment effects for almost the entire individuals. In addition, experiments using synthetic and semi-synthetic datasets demonstrated the effectiveness of the proposed method, in particular, when labeled data are limited. One of the drawbacks of CP is that it might fail when there are noisy unlabeled individuals. As some previous studies regarding semi-supervised learning demonstrated, semi-supervised learning methods can be vulnerable to noisy unlabeled data. In Chapter 4, we presented Graphite, a framework for addressing graph-structured treatments and mitigating bias through HSIC regularization. In certain contexts, such as the evaluation of drug efficacy, the number of treatments under consideration may be substantial. Thus, mitigating bias between covariates and graph-structured treatments is a technical challenge. Hence, by leveraging HSIC regularization, we provided theoretical guarantees for mitigating bias through the utilization of individual targets and treatment representations. Furthermore, based on the experiments on the two real-world datasets, Graphite demonstrated superior performance when compared with strong baseline methods. In Chapter 5, we discussed treatment effect estimation in the presence of latent confounding variables. Unlike the typical assumption in most studies that the confounding variables influencing both treatment assignment and the outcome are available in observational data, it is nearly impracticable to ensure the complete observation of such necessary variables. Furthermore, obtaining covariates containing sensitive or confidential information is challenging, due to privacy concern, for example. Owing to the remarkable expressive power of VAE, VAE-based methods have proven to be effective in treatment effect estimation problems in the presence of hidden confounding variables. However, the recent theoretical analysis revealed that a naive application of VAE may not yield accurate models even upon reaching its optimal solution. Our analysis further revealed that a specific class of dataset that current VAE-based methods failed to give accurate treatment effect even when they achieve an optimal solution. Hence, we proposed a classic matching method using hidden confounding variables that guarantees correct treatment effects when it achieves the optimal solution. Furthermore, based on the experiments using synthetic and semi-synthetic datasets, we demonstrated the effectiveness of InfoCEVAE.

In conclusion, this thesis analyzed three practical challenges in the estimation of treatment effects from observational data. Subsequently, three innovative methods, leveraging machine learning techniques, were proposed to address these difficulties. Despite the progress made, there is still substantial room for further improvement in treatment effect estimation, with the potential of having a decisive impact on a range of fields where treatment effects play a critical role.

6.2 Future directions

Finally, we discuss future directions. In Chapter 3 and 4, we utilized auxiliary information such as unlabeled individuals or graph-structured information. However, the acquisition of such data may incur significant costs. Additionally, as we discussed in Chapter 3, while some individuals may contribute to improving predictive accuracy, others may prove useless or even significantly hurt performance. Consequently, in the process of data acquisition, the implementation of methods that select individuals possessing informative features, such as active learning-based methods [114, 115], may be a promising solution in terms of data accumulation cost and predictive accuracy. In Chapter 5, it is assumed that hidden confounding variables are continuous variables drawn from a Gaussian distribution. However, this assumption may not always be true and there exist scenarios in which hidden confounding variables are discrete variables such as binary or categorical. Currently, there is no affirmation that the same results would be obtained in such cases. Further exploring the theoretical analysis with regard to discrete variables appears to be an interesting direction. Furthermore, we can also consider complex models such as Normalizing Flow [116, 117] which can handle more extensive generative models.

List of Publications

1. (**Chapter 3**): Shonosuke Harada and Hisashi Kashima. Counterfactual Propagation for Semi-Supervised Individual Treatment Effect Estimation. In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 542–558, 2021.
   https://doi.org/10.1007/978-3-030-67658-2_31

2. (**Chapter 4**): Shonosuke Harada and Hisashi Kashima. GraphITE: Estimating Individual Effects of Graph-Structured Treatments. In: *Proceedings of the 30th ACM International Conference on Information and Knowledge Management*, pp. 659–668, 2021.
   https://doi.org/10.1145/3459637.3482349

3. (**Chapter 5**): Shonosuke Harada and Hisashi Kashima. InfoCEVAE: Treatment Effect Estimation with Hidden Confounding Variables Matching. In: *Machine Learning*, pp. 1–19, 2022.
   https://doi.org/10.1007/s10994-022-06246-0

4. (**Chapter 3**): 原田 将之介, 鹿島 久嗣. 反事実伝播: 介入効果推定のための半教師付き学習. 人工知能学会論文誌 37(3), B-LA3_1-14, 2022.
   https://doi.org/10.1527/tjsai.37-3_B-LA3

5. (**Chapter 4**): 原田 将之介, 鹿島 久嗣. GraphITE: グラフ介入に対する介入効果推定. 人工知能学会論文誌 37(6), D-M73_1-11, 2022.
   https://doi.org/10.1527/tjsai.37-2_D-M73

Bibliography

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet Classification with Deep Convolutional Neural Networks. In: *Communications of the ACM* 60.6, pp. 84–90, 2017.

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

[3] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. Distributed Representations of Words and Phrases and their Compositionality. In: *Advances in Neural Information Processing Systems*, pp. 3111–3119, 2013.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, 2019.

[5] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural Message Passing for Quantum Chemistry. In: *Proceedings of the 34th International Conference on Machine Learning*, pp. 1263–1272, 2017.

[6] David K. Duvenaud, Dougal Maclaurin, Jorge Iparraguirre, Rafael Bombarell, Timothy Hirzel, Alán Aspuru-Guzik, and Ryan P. Adams. Convolutional Networks on Graphs for Learning Molecular Fingerprints. In: *Advances in Neural Information Processing Systems*, pp. 2224–2232, 2015.

[7] Shonosuke Harada, Hirotaka Akita, Masashi Tsubaki, Yukino Baba, Ichigaku Takigawa, Yoshihiro Yamanishi, and Hisashi Kashima. Dual Graph Convolutional Neural Network for Predicting Chemical Networks. In: *BMC Bioinformatics* 21, pp. 1–13, 2020.

[8] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. BPR: Bayesian Personalized Ranking from Implicit Feedback. In: *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence*, pp. 452–461, 2009.

[9] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L. Hamilton, and Jure Leskovec. Graph Convolutional Neural Networks for Web-Scale Recommender Systems. In: *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 974–983, 2018.

[10] Jerzy Splawa-Neyman, Dorota M. Dabrowska, and T.P. Speed. On the Application of Probability Theory to Agricultural Experiments. Essay on Principles. Section 9. In: *Statistical Science*, pp. 465–472, 1990.

[11] Donald B. Rubin. Estimating Causal Effects of Treatments in Randomized and Nonrandomized Studies. In: *Journal of Educational Psychology* 66.5, p. 688, 1974.

[12] Donald B. Rubin. Matching to Remove Bias in Observational Studies. In: *Biometrics* 29.1, pp. 159–183, 1973.

[13] Sheng Li, Nikos Vlassis, Jaya Kawale, and Yun Fu. Matching via Dimensionality Reduction for Estimation of Treatment Effects in Digital Marketing

Campaigns. In: *Proceedings of the 25th International Joint Conference on Artificial Intelligence*, pp. 3768–3774, 2016.

[14] Leo Breiman. Random Forests. In: *Machine Learning* 45.1, pp. 5–32, 2001.

[15] Jennifer L. Hill. Bayesian Nonparametric Modeling for Causal Inference. In: *Journal of Computational and Graphical Statistics* 20.1, pp. 217–240, 2011.

[16] Stefan Wager and Susan Athey. Estimation and Inference of Heterogeneous Treatment Effects using Random Forests. In: *Journal of the American Statistical Association* 113.523, pp. 1228–1242, 2018.

[17] Fredrik D. Johansson, Uri Shalit, and David Sontag. Learning Representations for Counterfactual Inference. In: *Proceedings of the 33rd International Conference on Machine Learning*, pp. 3020–3029, 2016.

[18] Uri Shalit, Fredrik D. Johansson, and David Sontag. Estimating Individual Treatment Effect: Generalization Bounds and Algorithms. In: *Proceedings of the 34th International Conference on Machine Learning*, pp. 3076–3085, 2017.

[19] Claudia Shi, David M. Blei, and Victor Veitch. Adapting Neural Networks for the Estimation of Treatment Effects. In: *Advances in Neural Information Processing Systems*, pp. 2503–2513, 2019.

[20] Jinsung Yoon, James Jordon, and Mihaela van der Schaar. GANITE: Estimation of Individualized Treatment Effects Using Generative Adversarial Nets. In: *Proceedigns of the 6th International Conference on Learning Representations*, 2018.

[21] Judea Pearl. Causality. 2009.

[22] Judea Pearl. Causal Inference. In: *Causality: Objectives and Assessment*, pp. 39–58, 2010.

[23] Manabu Kuroki and Judea Pearl. Measurement Bias and Effect Restoration in Causal Inference. In: *Biometrika* 101.2, pp. 423–437, 2014.

[24] Christos Louizos, Uri Shalit, Joris Mooij, David Sontag, Richard Zemel, and Max Welling. Causal Effect Inference with Deep Latent-Variable Models. In: *Advances in Neural Information Processing Systems*, pp. 6446–6456, 2017.

[25] Wang Miao, Zhi Geng, and Eric J. Tchetgen Tchetgen. Identifying Causal Effects with Proxy Variables of an Unmeasured Confounder. In: *Biometrika* 105.4, pp. 987–993, 2018.

[26] Liyuan Xu, Heishiro Kanagawa, and Arthur Gretton. Deep Proxy Causal Learning and Its Application to Confounded Bandit Policy Evaluation. In: *Advances in Neural Information Processing Systems*, pp. 26264–26275, 2021.

[27] Daniel E. Ho, Kosuke Imai, Gary King, and Elizabeth A. Stuart. Matching as nonparametric preprocessing for reducing model dependence in parametric causal inference. In: *Political Analysis* 15.3, pp. 199–236, 2007.

[28] Weijia Zhang, Lin Liu, and Jiuyong Li. Treatment Effect Estimation with Disentangled Latent Factors. In: *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, pp. 10923–10930, 2021.

[29] Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In: *Proceedings of the 2nd International Conference on Learning Representations*, 2014.

[30] Guido W. Imbens and Donald B. Rubin. Causal Inference for Statistics, Social, and Biomedical Sciences. 2015.

[31] Xiaojin Zhu and Andrew B. Goldberg. Introduction to Semi-Supervised Learning. In: *Synthesis Lectures on Artificial Intelligence and Machine Learning* 3.1, pp. 1–130, 2009.

[32] Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien, eds. Semi-Supervised Learning. 2006.

[33] Xiaojin Zhu, Zoubin Ghahramani, and John D. Lafferty. Semi-Supervised Learning Using Gaussian Fields and Harmonic Functions. In: *Proceedings of the 20th International conference on Machine learning*, pp. 912–919, 2003.

[34] Shengjia Zhao, Jiaming Song, and Stefano Ermon. InfoVAE: Balancing Learning and Inference in Variational Autoencoders. In: *Proceedings of the 33rd AAAI Conference on Artificial Intelligence*, pp. 5885–5892, 2019.

[35] Donald B. Rubin. Causal Inference Using Potential Outcomes: Design, Modeling, Decisions. In: *Journal of the American Statistical Association* 100.469, pp. 322–331, 2005.

[36] Jasjeet S. Sekhon. The Neyman—Rubin Model of Causal Inference and Estimation Via Matching Methods. In: *The Oxford Handbook of Political Methodology*, pp. 271–299, 2008.

[37] Guido W. Imbens. The Role of the Propensity Score in Estimating Dose-Response Functions. In: *Biometrika* 87.3, pp. 706–710, 2000.

[38] Judea Pearl. Causal Diagrams for Empirical Research. In: *Biometrika* 82.4, pp. 669–688, 1995.

[39] David Chan, Rong Ge, Ori Gershony, Tim Hesterberg, and Diane Lambert. Evaluating Online Ad Campaigns in a Pipeline: Causal Models At Scale. In: *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 7–16, 2010.

[40] Robert J. LaLonde. Evaluating the Econometric Evaluations of Training Programs with Experimental Data. In: *The American Economic Review*, pp. 604–620, 1986.

[41] Thomas A. Glass, Steven N. Goodman, Miguel A. Hernán, and Jonathan M. Samet. Causal Inference in Public Health. In: *Annual Review of Public Health* 34, pp. 61–75, 2013.

[42] David Lewis. Causation. In: *The Journal of Philosophy* 70.17, pp. 556–567, 1974.

[43] Paul R. Rosenbaum and Donald B. Rubin. The Central Role of the Propensity Score in Observational Studies for Causal Effects. In: *Biometrika* 70.1, pp. 41–55, 1983.

[44] Michael Baiocchi, Jing Cheng, and Dylan S. Small. Instrumental Variable Methods for Causal Inference. In: *Statistics in Medicine* 33.13, pp. 2297–2340, 2014.

[45] Filip Radlinski and Thorsten Joachims. Query Chains: Learning to Rank from Implicit Feedback. In: *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, pp. 239–248, 2005.

[46] Nuno Pombo, Nuno Garcia, Kouamana Bousson, and Virginie Felizardo. "Machine Learning Approaches to Automated Medical Decision Support Systems". In: *Handbook of Research on Artificial Intelligence Techniques and Algorithms*. IGI Global, 2015, pp. 183–203.

[47] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold Regularization: A Geometric Framework for Learning from Labeled and Unlabeled Examples. In: *Journal of Machine Learning Research* 7, pp. 2399–2434, 2006.

[48] Jason Weston, Frédéric Ratle, Hossein Mobahi, and Ronan Collobert. Deep Learning via Semi-Supervised Embedding. In: *Neural Networks: Tricks of the Trade*, pp. 639–655, 2012.

[49] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In: *Proceedings of the 3rd International Conference on Learning Representations*, 2015.

[50] Vincent Dorie. NPCI: Non-Parametrics for Causal Inference. In: *URL https://github.com/vdorie/npci*, 2016.

[51] Liuyi Yao, Sheng Li, Yaliang Li, Mengdi Huai, Jing Gao, and Aidong Zhang. Representation Learning for Treatment Effect Estimation from Observational Data. In: *Advances in Neural Information Processing Systems*, pp. 2633–2643, 2018.

[52] Arash Vahdat. Toward Robustness Against Label Noise in Training Deep Discriminative Neural Networks. In: *Advances in Neural Information Processing Systems*, pp. 5596–5605, 2017.

[53] Thang D. Bui, Sujith Ravi, and Vivek Ramavajjala. Neural Graph Learning: Training Neural Networks Using Graphs. In: *Proceedings of the 11th ACM International Conference on Web Search and Data Mining*, pp. 64–71, 2018.

[54] Bo Du, Tang Xinyao, Zengmao Wang, Lefei Zhang, and Dacheng Tao. Robust Graph-Based Semisupervised Learning for Noisy Labeled Data via Maximum Correntropy Criterion. In: *IEEE Transactions on Cybernetics* 49.4, pp. 1440–1453, 2018.

[55] Wei Liu, Jun Wang, and Shih-Fu Chang. Robust and Scalable Graph-Based Semisupervised Learning. In: *Proceedings of the IEEE* 100.9, pp. 2624–2638, 2012.

[56] Alberto Abadie and Guido W. Imbens. Large Sample Properties of Matching Estimators for Average Treatment Effects. In: *Econometrica* 74.1, pp. 235–267, 2006.

[57] Paul R. Rosenbaum and Donald B. Rubin. Constructing a Control Group Using Multivariate Matched Sampling Methods That Incorporate the Propensity Score. In: *The American Statistician* 39.1, pp. 33–38, 1985.

[58] Jared K. Lunceford and Marie Davidian. Stratification and Weighting via the Propensity Score in Estimation of Causal Treatment Effects: A Comparative Study. In: *Statistics in Medicine* 23.19, pp. 2937–2960, 2004.

[59] Hugh A. Chipman, Edward I. George, and Robert E. McCulloch. BART: Bayesian Additive Regression Trees. In: *The Annals of Applied Statistics* 4.1, pp. 266–298, 2010.

[60] Ruocheng Guo, Jundong Li, and Huan Liu. Learning Individual Causal Effects from Networked Observational Data. In: *Proceedings of the 13th International Conference on Web Search and Data Mining*, pp. 232–240, 2020.

[61] Hamidreza Alvari, Elham Shaabani, Soumajyoti Sarkar, Ghazaleh Beigi, and Paulo Shakarian. Less is More: Semi-Supervised Causal Inference for Detecting Pathogenic Users in Social Media. In: *Proceedings of the ACM Web Conference*, pp. 154–161, 2019.

[62] Victor Veitch, Yixin Wang, and David M. Blei. Using Embeddings to Correct for Unobserved Confounding in Networks. In: *Advances in Neural Information Processing Systems*, pp. 13769–13779, 2019.

[63] Yoshua Bengio, Pascal Lamblin, Dan Popovici, and Hugo Larochelle. Greedy Layer-Wise Training of Deep Networks. In: *Advances in Neural Information Processing Systems 20*, pp. 153–160, 2007.

[64] Geoffrey E. Hinton, Simon Osindero, and Yee-Whye Teh. A Fast Learning Algorithm for Deep Belief Nets. In: *Neural Computation* 18.7, pp. 1527–1554, 2006.

[65] Zhilin Yang, William Cohen, and Ruslan Salakhudinov. Revisiting Semi-Supervised Learning with Graph Embeddings. In: *Proceedings of the 33rd International Conference on Machine Learning*, pp. 40–48, 2016.

[66] Thomas N. Kipf and Max Welling. Semi-Supervised Classification with Graph Convolutional Networks. In: *Proceedings of the 5th International Conference on Learning Representations*, 2017.

[67] Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondrej Chum. Label Propagation for Deep Semi-supervised Learning. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5070–5079, 2019.

[68] Aditya Pal and Deepayan Chakrabarti. Label Propagation with Neural Networks. In: *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pp. 1671–1674, 2018.

[69] Fan Zhou, Tengfei Li, Haibo Zhou, Hongtu Zhu, and Ye Jieping. Graph-Based Semi-Supervised Learning with Nonignorable Nonresponses. In: *Advances in Neural Information Processing Systems*, pp. 7013–7023, 2019.

[70] Brigitte Ganter, Stuart Tugendreich, Cecelia I. Pearson, Eser Ayanoglu, Susanne Baumhueter, Keith A. Bostian, Lindsay Brady, Leslie J. Browne, John Calvin, Gwo-Jen Day, et al. Development of a Large-Scale Chemogenomics Database to Improve Drug Candidate Selection. In: *Journal of Biotechnology* 119.3, pp. 219–244, 2005.

[71] Swen Hoelder, Paul A. Clarke, and Paul Workman. Discovery of Small Molecule Cancer Drugs: Successes, Challenges and Opportunities. In: *Molecular Oncology* 6.2, pp. 155–176, 2012.

[72] Shiv Kumar Saini, Sunny Dhamnani, Akil Arif Ibrahim, and Prithviraj Chavan. Multiple Treatment Effect Estimation using Deep Generative Model with Task Embedding. In: *Proceedings of the ACM Web Conference*, pp. 1601–1611, 2019.

[73] Patrick Schwab, Lorenz Linhardt, and Walter Karlen. Perfect Match: A Simple Method for Learning Representations For Counterfactual Inference With Neural Networks. In: *arXiv preprint arXiv:1810.00656*, 2018.

[74] Romain Lopez, Chenchen Li, Xiang Yan, Junwu Xiong, Michael I Jordan, Yuan Qi, and Le Song. Cost-Effective Incentive Allocation via Structured

Counterfactual Inference. In: *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, pp. 4997–5004, 2020.

[75] H-G Eichler, Brigitte Bloechl-Daum, Peter Bauer, Frank Bretz, Jeffrey Brown, Lisa Victoria Hampson, Peter Honig, Michael Krams, Hubert Leufkens, Robyn Lim, et al. "Threshold-Crossing": A Useful Way to Establish the Counterfactual in Clinical Trials? In: *Clinical Pharmacology & Therapeutics* 100.6, pp. 699–712, 2016.

[76] Siyuan Zhao and Neil Heffernan. Estimating Individual Treatment Effect from Educational Studies with Residual Counterfactual Networks. In: *Proceedings of the 10th International Conference on Educational Data Mining*, 2017.

[77] Ian J. 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*, pp. 2672–2680, 2014.

[78] Ioana Bica, Ahmed M. Alaa, Craig Lambert, and Mihaela van der Schaar. From Real-World Patient Data to Individualized Treatment Effects Using Machine Learning: Current and Future Methods to Address Underlying Challenges. In: *Clinical Pharmacology & Therapeutics* 109.1, pp. 87–100, 2021.

[79] Patrick Schwab, Lorenz Linhardt, Stefan Bauer, Joachim M. Buhmann, and Walter Karlen. Learning Counterfactual Representations for Estimating Individual Dose-Response Curves. In: *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, pp. 5612–5619, 2020.

[80] Hao Wang, Hao He, and Dina Katabi. Continuously Indexed Domain Adaptation. In: *Proceedings of the 37th International Conference on Machine Learning*, pp. 9898–9907, 2020.

[81] Muhan Zhang and Yixin Chen. Link Prediction Based on Graph Neural Networks. In: *Advances in Neural Information Processing Systems*, pp. 5171–5181, 2018.

[82] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive Representation Learning on Large Graphs. In: *Advances in Neural Information Processing Systems*, pp. 1024–1034, 2017.

[83] Mikael Henaff, Joan Bruna, and Yann LeCun. Deep Convolutional Networks on Graph-Structured Data. In: *arXiv preprint arXiv:1506.05163*, 2015.

[84] Kristof T. Schütt, Huziel E. Sauceda, P-J Kindermans, Alexandre Tkatchenko, and K-R Müller. SchNet–A Deep Learning Architecture for Molecules and Materials. In: *The Journal of Chemical Physics* 148.24, p. 241722, 2018.

[85] Jiaxuan You, Bowen Liu, Zhitao Ying, Vijay Pande, and Jure Leskovec. Graph Convolutional Policy Network for Goal-Directed Molecular Graph Generation. In: *Advances in Neural Information Processing Systems*, pp. 6410–6421, 2018.

[86] Chengxi Zang and Fei Wang. MoFlow: An Invertible Flow Model for Generating Molecular Graphs. In: *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 617–626, 2020.

[87] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How Powerful are Graph Neural Networks? In: *Proceedings of the 6th International Conference on Learning Representations*, 2018.

[88] Ryan Murphy, Balasubramaniam Srinivasan, Vinayak Rao, and Bruno Ribeiro. Relational Pooling for Graph Representations. In: *Proceedings of the 36th International Conference on Machine Learning*, pp. 4663–4673, 2019.

[89] Jenny Liu, Aviral Kumar, Jimmy Ba, Jamie Kiros, and Kevin Swersky. Graph Normalizing Flows. In: *Advances in Neural Information Processing Systems*, pp. 13578–13588, 2019.

[90] Shonosuke Harada and Hisashi Kashima. Counterfactual Propagation for Semi-Supervised Individual Treatment Effect Estimation. In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 542–558, 2021.

[91] Yunpu Ma and Volker Tresp. Causal Inference under Networked Interference and Intervention Policy Enhancement. In: *International Conference on Artificial Intelligence and Statistics*, pp. 3700–3708, 2021.

[92] Arthur Gretton, Kenji Fukumizu, Choon Teo, Le Song, Bernhard Schölkopf, and Alex Smola. A Kernel Statistical Test of Independence. In: *Advances in Neural Information Processing systems*, pp. 585–592, 2007.

[93] Thao Nguyen, Maithra Raghu, and Simon Kornblith. Do Wide and Deep Networks Learn the Same Things? Uncovering How Neural Network Representations Vary with Width and Depth. In: *Proceedings of the 9th International Conference on Learning Representations*, 2021.

[94] Makoto Yamada, Yuta Umezu, Kenji Fukumizu, and Ichiro Takeuchi. Post Selection Inference with Kernels. In: *Proceedings of the 21st International Conference on Artificial Intelligence and Statistics*, pp. 152–160, 2018.

[95] Fredrik D. Johansson, Uri Shalit, Nathan Kallus, and David Sontag. Generalization Bounds and Representation Learning for Estimation of Potential Outcomes and Causal Effects. In: *Journal of Machine Learning Research* 23.166, pp. 1–50, 2022.

[96] Jordi Barretina, Giordano Caponigro, Nicolas Stransky, Kavitha Venkatesan, Adam A. Margolin, Sungjoon Kim, Christopher J. Wilson, Joseph Lehár, Gregory V. Kryukov, Dmitriy Sonkin, et al. The Cancer Cell Line Encyclo-

pedia enables predictive modelling of anticancer drug sensitivity. In: *Nature* 483.7391, pp. 603–607, 2012.

[97] Wanjuan Yang, Jorge Soares, Patricia Greninger, Elena J. Edelman, Howard Lightfoot, Simon Forbes, Nidhi Bindal, Dave Beare, James A. Smith, I. Richard Thompson, et al. Genomics of Drug Sensitivity in Cancer (GDSC): A Resource for Therapeutic Biomarker Discovery in Cancer Cells. In: *Nucleic Acids Research* 41.D1, pp. D955–D961, 2012.

[98] Alex P. Lind and Peter C. Anderson. Predicting Drug activity against Cancer Cells by Random Forest Models Based on Minimal Genomic Information and chemical properties. In: *PLoS ONE* 14.7, e0219774, 2019.

[99] Chayaporn Suphavilai, Denis Bertrand, and Niranjan Nagarajan. Predicting Cancer Drug Response Using a Recommender System. In: *Bioinformatics* 34.22, pp. 3907–3914, 2018.

[100] Frank E. Harrell Jr., Kerry L. Lee, and Daniel B. Mark. Multivariable Prognostic Models: Issues in Developing Models, Evaluating Assumptions and Adequacy, and Measuring and Reducing Errors. In: *Statistics in Medicine* 15.4, pp. 361–387, 1996.

[101] Roman Kurilov, Benjamin Haibe-Kains, and Benedikt Brors. Assessment of Modelling Strategies for Drug Response Prediction in Cell Lines and Xenografts. In: *Scientific Reports* 10.1, pp. 1–11, 2020.

[102] Zhaleh Safikhani, Petr Smirnov, Kelsie L. Thu, Jennifer Silvester, Nehme El-Hachem, Rene Quevedo, Mathieu Lupien, Tak W. Mak, David Cescon, and Benjamin Haibe-Kains. Gene Isoforms as Expression-Based Biomarkers Predictive of Drug Response in Vitro. In: *Nature Communications* 8.1, pp. 1–11, 2017.

[103] Zhenqin Wu, Bharath Ramsundar, Evan N. Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S. Pappu, Karl Leswing, and Vijay Pande. MoleculeNet:

A Benchmark for Molecular Machine Learning. In: *Chemical Science* 9.2, pp. 513–530, 2018.

[104] Kenneth J. Rothman, Sander Greenland, and Timothy L. Lash. Modern Epidemiology. 2008.

[105] Zhihong Cai and Manabu Kuroki. On Identifying Total Effects in the Presence of Latent Variables and Selection Bias. In: *Proceedings of the 24th Conference on Uncertainty in Artificial Intelligence*, pp. 62–69, 2008.

[106] Dustin Tran, Rajesh Ranganath, and David M. Blei. The Variational Gaussian Process. In: *Proceedings of the 4th International Conference on Learning Representations*, 2016.

[107] Jasjeet S. Sekhon. Opiates for the Matches: Matching Methods for Causal Inference. In: *Annual Review of Political Science* 12, pp. 487–508, 2009.

[108] Gary King and Richard Nielsen. Why Propensity Scores Should Not Be Used for Matching. In: *Political Analysis* 27.4, pp. 435–454, 2019.

[109] Shonosuke Harada and Hisashi Kashima. GraphITE: Estimating Individual Effects of Graph-Structured Treatments. In: *Proceedings of the 30th ACM International Conference on Information and Knowledge Management*, pp. 659–668, 2021.

[110] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. Unsupervised Image-to-Image Translation Networks. In: *Advances in Neural Information Processing Systems*, pp. 700–708, 2017.

[111] Yishu Miao, Lei Yu, and Phil Blunsom. Neural Variational Inference for Text Processing. In: *Proceedings of the 33th International Conference on Machine Learning*, pp. 1727–1736, 2016.

[112] Qi Liu, Miltiadis Allamanis, Marc Brockschmidt, and Alexander Gaunt. Constrained Graph Variational Autoencoders for Molecule Design. In: *Advances in Neural Information Processing Systems*, pp. 7806–7815, 2018.

[113] Djork-Arné Clevert, Thomas Unterthiner, and Sepp Hochreiter. Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). In: *Proceedings of the 3rd International Conference on Learning Representations*, 2016.

[114] Burr Settles. Active Learning. In: *Synthesis Lectures on Artificial Intelligence and Machine Learning* 6.1, pp. 1–114, 2012.

[115] Iiris Sundin, Peter Schulam, Eero Siivola, Aki Vehtari, Suchi Saria, and Samuel Kaski. Active Learning for Decision-Making from Imbalanced Observational Data. In: *Proceedings of the 36th International Conference on Machine Learning*, pp. 6046–6055, 2019.

[116] Danilo Rezende and Shakir Mohamed. Variational Inference with Normalizing Flows. In: *Proceedings of the 32nd International Conference on Machine Learning*, pp. 1530–1538, 2015.

[117] Ilyes Khemakhem, Ricardo Monti, Robert Leech, and Aapo Hyvarinen. Causal Autoregressive Flows. In: *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics*, pp. 3520–3528, 2021.

[118] Shonosuke Harada and Hisashi Kashima. InfoCEVAE: Treatment Effect Estimation with Hidden Confounding Variables Matching. In: *Machine Learning*, pp. 1–19, 2022.