Deep Representation Learning for Continuous Treatment Effect Estimation

by

Amirreza Kazemi

B.Sc., Sharif University of Technology, 2021

Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Science

in the
School of Computing Science
Faculty of Applied Sciences

SIMON FRASER UNIVERSITY
Fall 2023

Declaration of Committee

Name:	**Amirreza Kazemi**
Degree:	**Master of Science**
Thesis title:	**Deep Representation Learning for Continuous Treatment Effect Estimation**
Committee:	**Chair:** Lawrence McCandless Professor, Health Sciences
	Martin Ester Supervisor Professor, Computing Science
	Sharan Vaswani Committee Member Assistant Professor, Computing Science
	Oliver Schulte Examiner Professor, Computing Science

Abstract

Estimating the individual treatment effect (ITE) from observational datasets has important applications in domains such as personalized medicine, economics, and recommendation systems. The observational datasets often exhibit treatment-selection bias, resulting in a distribution shift among populations of samples that received different treatments. While deep representation learning has shown great promise in adjusting for covariate shifts when the treatment is a binary variable, the more practical and challenging task of handling continuous treatments (e.g., dosage of a medication) remains relatively underexplored. In this thesis, our aim is to address the associated challenges with continuous treatment. Specifically, we propose a deep model that mitigates the distribution shift through an adversarial procedure and predicts the potential outcomes using an attention mechanism. The model's objective is grounded in a theoretical upper bound on counterfactual prediction error. Our experimental evaluation on semi-synthetic datasets also demonstrates the method's empirical superiority over a range of state-of-the-art.

Keywords: Representation Learning; Causal Inference; Treatment Effect Estimation; Dose Response Estimation

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Causal Inference, the question of how a response would change when its causal effect changes, plays a pivotal role in various domains, such as healthcare [37, 11], economics [45], or recommendation systems [31]. For example, in medicine, it aids in the decision-making process regarding which medication yields better outcomes for patients, and in economics, it helps estimate the impact of minimum wages on the employment rate of the population. Treatment effect estimation is a problem in causal inference with the goal of measuring the effect of a treatment (intervention) on an outcome of interest [25, 36]. Randomized control trials (RCTs) are the most effective way of estimating treatment effects since they randomly assign the population to different treatment groups. However, conducting RCTs can be expensive or impractical [40]. Alternatively, observational datasets have been employed due to their abundance. An observational dataset comprises samples with their covariates (features) denoted as X, their assigned treatment denoted as T, and their outcome under the treatment assignment denoted as Y. The causal graph of the variables in observational datasets and RCTs, along with a toy example of an observational dataset, are shown in Figure 1.1 and Table 1.1, respectively.

The effect estimation problem is commonly studied within the potential outcome framework, which aims to predict potential outcomes given the covariates and treatment, and then compare them to estimate the effect [39]. Potential outcomes are the outcomes that could occur under different treatments. For instance, in the case of binary treatment, where $T = 0$ represents the control group and $T = 1$ represents the treated group, if $Y(1)$ denotes the outcome under treatment 1 and $Y(0)$ denotes the outcome under treatment 0, the treatment effect for a patient with covariates x can be defined as $\tau(x) = E[Y(1) - Y(0)|X = x]$. However, in observational data, the patient only receives one of the treatments, and we observe the outcome under that treatment (known as the factual outcome), while the outcomes under the other treatments are unknown (referred to as counterfactual outcomes). This means that in order to estimate the difference, we need to answer a counterfactual question: 'What would be the outcome if a patient had received a different treatment?'

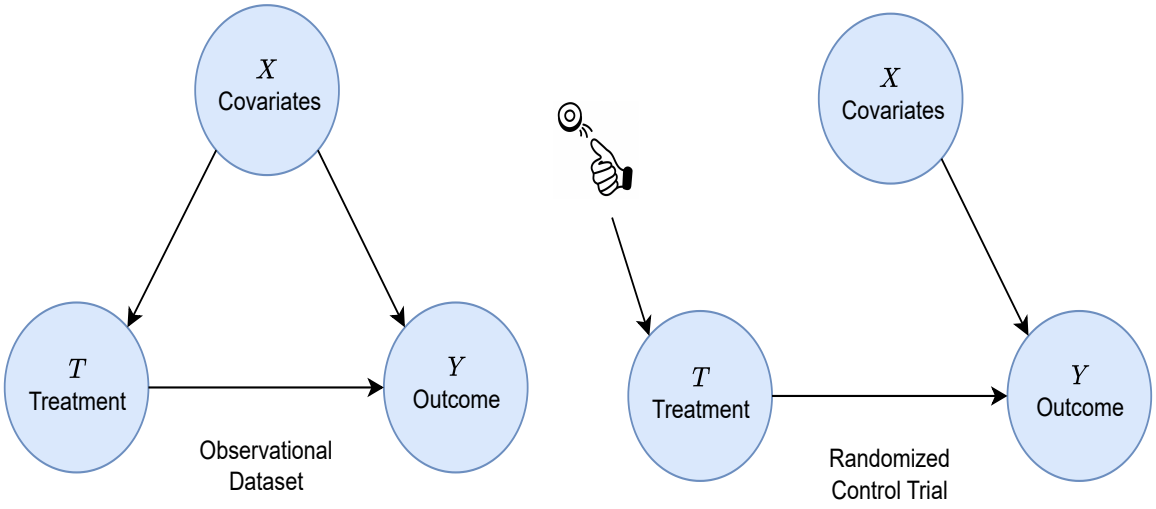

Figure 1.1: The left figure demonstrates the causal graph in an observational dataset. The covariates X have confounding effect on treatment T and outcome Y, and thus we need to adjust for this shift. The ideal scenario is randomized control trial shown in the right figure, where there is no causal edge between X and T and thus no confounding bias affect the causal estimation. The core idea of representation learning is trying to find a lower-dimensional representation similar to the covariates in randomized control trials.

Answering such counterfactual questions requires assumptions about the underlying data generating process. The strong ignorability assumption, which ensures that there is no non-causal association flow between treatment T and outcome Y in the causal graph except through covariates X, is sufficient to make potential outcomes identifiable. In other words, all the confounders are among the covariates X, and by adjusting for them (blocking the association path between T and Y through X), we can estimate the causal effect.

Age	Gender	Salary	T	$Y(1)$	$Y(0)$
24	M	50k	0	-	3
29	M	70k	1	2	-
35	F	100k	0	-	10
39	F	60k	1	3	-

Table 1.1: A toy example of an observational dataset with binary treatment. As shown only one potential outcome is observed for each sample. Age, gender and salary are the covariates.

Numerous statistical [27, 24] and machine learning [48, 43] methods have been proposed to adjust for confounders. While the focus of the literature has mostly been on average-level effects (the effect averaged over the entire population), in recent years, particularly in applications such as precision medicine and recommendation systems, there has been a need to estimate heterogeneous (individual) treatment effects [29, 10]. The balanced representation learning approach has recently shown remarkable success in the estimation of individual

effects [28]. This approach is based on viewing the problem as a covariate shift problem in domain adaptation [4, 33]. Since there exists a causal edge between T and X, the observational dataset exhibits treatment-selection bias, and the covariate distribution of groups receiving different treatments is different. Therefore, we can predict the potential outcomes using the treatment and balanced covariates (where the covariate distribution of groups is the same) and link the difference between outcomes to the treatment effect. Instead of balancing the covariates themselves [14, 26], the authors of seminal work [28, 42] have shown that we can learn a representation of covariates which has a small dependence on treatment and (almost) contains the necessary information to predict the factual outcome. To be more specific, [42] proposed the Treatment-Agnostic Representation Network (TARNet) consisting of an encoder mapping the samples from the covariate space to the latent representation space, and outcome prediction networks predicting the factual outcome from the representation with respect to the given treatment. The architecture is shown in Figure 2. The network has two objectives: predicting factual outcomes accurately with a supervised learning loss function, and minimizing the distribution shift in the representation space between sub-populations using a distributional distance metric called the Integral Probability Metric. Examples of IPM distance class are Wasserstein distance [47, 9], Hilbert Schmidt Criterion Index [21], and Maximum Mean Discrepancy [20]. The objective is inspired by their theoretical analysis demonstrating that factual outcome error plus the IPM term provides a bound on the counterfactual outcome error.

However, the proposed network and its theoretical analysis were limited to binary treatments, with $(T = 0)$ representing control groups and $(T = 1)$ representing treated groups. There have been a few attempts to extend the approach to more practical continuous-valued treatments, such as dosage of a medicine, by modifying the architecture [35, 41] or the objective [3, 51]. The architecture needs to be modified because the prediction network cannot have a separate head for each treatment arm. The objective also needs to be modified because the distance between possible infinite sub-populations should be minimized.

Recently, the authors of [3] extended the theoretical analysis of [42] and demonstrated that minimizing the IPM between $P(Z, T)$ and $P(Z)P(T)$ results in a similar bound on counterfactual error for continuous treatments. Given that the marginal distribution $P(Z)$ is unknown, they approximated it with the IPM distance between $P(Z, T = t)$ and $P(Z, T = \neg t)$, where t denotes a treatment value assigned to a patient in the observational data, and $\neg t$ denotes all existing treatments in the data except t. Similarly, [51] suggested discretizing the treatment range into intervals and minimizing the maximum IPM distance between the distributions of the two intervals. Both methods involve non-parametric approximations of several IPM distances, which may be inaccurate for high-dimensional representation and small training data [32]. Furthermore, IPMs are by definition worst-case distances, and obtaining a treatment-invariant representation through IPM might be overly restrictive,

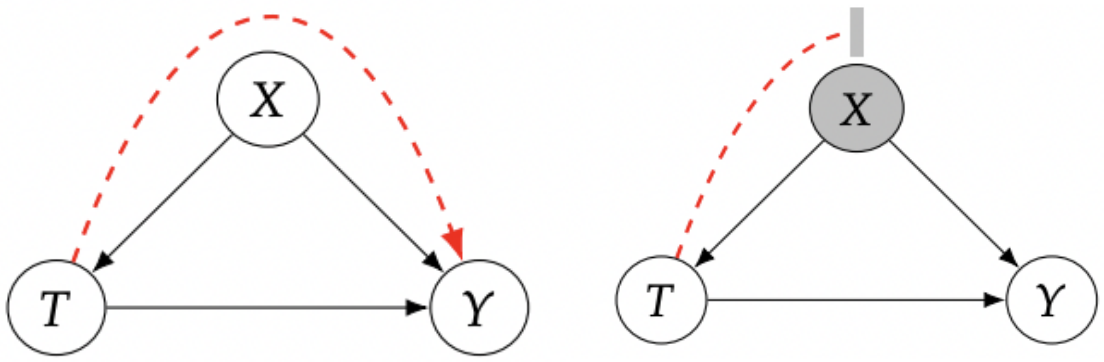

Figure 1.2: Covariate Adjustment (conditioning on X) ensures that the remaining association between two variables T and Y is causal.

potentially excluding important confounding factors for outcome prediction [56]. There are also non-IPM representation learning methods. For instance, [12, 5] proposed an adversarial game between a discriminator and the encoder to minimize the distance of different (non-continuous) treatment group representations, [54] proposed preserving the local similarity in the representation space, [22, 52] learn a disentangled representation to distinguish different latent factors, [56] enforce invertibility of the encoder function to prevent the loss of covariate information, and [1] introduced a regularization scheme to generalize to counterfactual outcomes. However, these methods lack theoretical justification [12, 22, 54] or their guarantees are limited to binary treatments [56]. To address this, we propose a novel generalization bound that uses Kullback-Leibler divergence instead of IPM to minimize the distance between the distribution corresponding to the observational dataset $P(Z, T)$ and the distribution corresponding to the RCT, $P(Z)P(T)$. We use an adversarial procedure to estimate and minimize the KL divergence and demonstrate its superiority compared to IPMs in our experiments.

The other challenge is that the network predicting the potential outcome from the representation must incorporate the treatment value. Having a distinct prediction head for each treatment value (as in the case of binary treatments shown in Figure 2) is not practical. Additionally, considering the treatment variable as an input of the outcome prediction network along with the representation leads to overfitting in the much higher-dimensional representation and significantly limits the impact of the treatment value [42, 41, 35]. Instead, [41] proposed the Dose Response Network (DRNet), which divides the treatment range into intervals and considers a distinct network for the prediction corresponding to treatments in that interval. [35] proposed a varying coefficient network (VCNet) that involves the treatment value in the network parameters through spline functions of treatment. However, both networks may not capture the dependency between representation and treatment effectively, as the choice of spline functions in VCNet or the intervals in DRNet are made

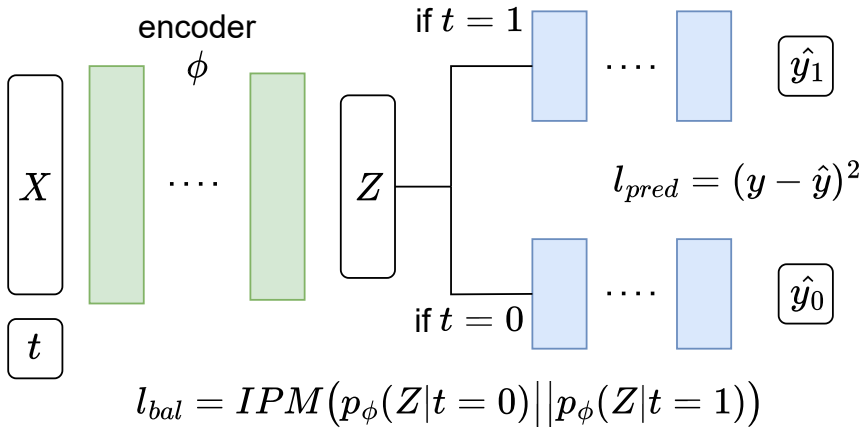

Figure 1.3: The architecture of Treatment-Agnostic Representation network (TARNet) [42]. All the samples are encoded into latent space Z and based on their assigned treatment will be given as input to the corresponding regression head. The objective is to minimize l_{bal} the IPM distance between representation distribution of two sub-populations, and to minimize the factual outcome prediction loss l_{pred}. If a sample received $t = 1$ ($t = 0$), only the top (bottom) prediction head will be used to predict the factual outcome.

irrespective of the representation and much of the flexibility is lost [8, 57]. To address this, we construct a treatment embedding using spline functions of treatment, and then employ a cross-attention mechanism taking representation and treatment as inputs to understand how relevant each treatment spline is to representation dimension for predicting potential outcomes. Constructing an embedding for scalar inputs in natural language processing [46] and tabular datasets [19]. Moreover, using spline functions of treatment has been shown to be effective to approximate a function [35]. Also, unlike [41, 35], the architecture is capable of incorporating many spline functions without introducing additional parameters

In this thesis, we aim to generalize the approach of learning a balanced representation with suitable optimization objective and architecture tailored for continuous treatment. Our proposed objective is based on a theoretical generalization bound, and can be estimated parameterically using an adversarial approach. We leverage attention mechanism in the outcome prediction network which substantially reduces the error compared to other architectures. Our main contributions are as follows:

1. We prove that under certain assumptions the counterfactual error is bounded by the factual error and the KL divergence between $P(Z)P(T)$ and $P(Z,T)$. Unlike the IPM distance, the KL divergence can be estimated parametrically, leading to more reliable bound.

2. We propose Adversarial Balanced Counterfactual Regression (ACFR) network. ACFR minimizes the KL divergence using an adversarial game extracting a balanced representation for continuous treatments. ACFR also minimizes the factual prediction error by a cross-attention network that captures the complex dependency between treatment and the representation.

3. We conduct an experimental comparison of ACFR against state-of-the-art methods on semi-synthetic datasets, News and TCGA, and analyze the robustness to varying-levels of treatment-selection bias for the methods.

The rest of the thesis is organized as follows. We discuss related works in chapter 2, the problem formulation and analysis in chapter 3, our methodology in chapter 4, the experiments in chapter 5 and finally the conclusion in chapter 6.

Chapter 2

Related Works

Treatment effect estimation is a broad field with a literature spanning across various disciplines including statistics and machine learning. We discuss the most related works to ours in the following three categories.

2.1 Average Treatment Effect Estimation

One prominent method is Inverse Propensity Weighting (IPW), which assigns weights $\frac{T}{e(x)} + \frac{1-T}{1-e(x)}$ to units based on propensity score $e(x)$ to reduce treatment-selection bias. The IPW estimator of ATE balances covariates between treatment and control groups, but its accuracy depends on the accuracy of the propensity score estimation. The DR estimator [38] was proposed to provide consistent and unbiased results leveraging an outcome regression model. Based on the sufficiency of propensity score theorem, DragonNet [43] has shown that you can learn a representation predictive of propensity score and outcome. The above methods has been generalized from binary to continuous treatments. [24] proposed the generalized propensity score (GPS) that generalizes the notion of propensity score to continuous treatments. [53, 16] proposed approaches to matching and covariate balancing, respectively, according to the weights learned using GPS. [35] proposed the Varying Coefficient network (VCNet) which extracts a representation sufficient for GPS prediction and predicts the outcome using a network where the treatment value influences the outcome indirectly through parameters instead of being given directly as input. The VCNet architecture is shown in right side of Figure 2.1. [2] proposes an entropy balancing method to learn more stable weights compared to GPS based weights. Our approach is fundamentally different since sufficiency of (generalized) propensity score theorem holds only for the average effect and we aim to estimate the individual effect.

2.2 Individual Treatment Effect Estimation

In the individual effect category, methods are mostly based on learning a balanced representation. [28] offers the capability to learn complex, nonlinear relationships and balancing the representation of covariates. [42] further improves the network by considering separate regression heads for each potential outcome since there is a risk of losing the influence of the treatment variable on the outcome with the concatenation. This allows for shared statistical power in the common representation layers, while retaining the influence of treatment in the separate heads. [22] introduces context-aware weighting scheme based on importance sampling in addition to representation learning, effectively mitigating treatment-selection bias in (ITE) estimation. [54] argues the necessity of imprtance sampleing and propose Local Similarity Preserved Individual Treatment Effect (SITE) estimation. The approach is rooted in semi supervised learning objectives and aims to preserve local similarity while simultaneously balancing data distributions. [22] modeled the underlying factors in observational datasets and they propose to identify the disentangled representations these factors. Leveraging the knowledge of underlying factors, help to effectively reduce the shift by balancing only the confounders. DRNet [41] considers continuous treatment values and discretized the treatment range into intervals, minimized the pair-wise shift between the populations fall within these intervals in the representation, and to handle continuous treatments they proposed a multi head network to predict outcomes for the intervals of treatment. The DRNet architecture is shown in the left side of Figure 2.1. [3] demonstrated that minimizing IPM between $P(Z,T)$ and $P(Z)P(T)$ coupled with factual outcome error leads to an upper bound of the counterfactual error in continuous treatment setup. Nonetheless, in practice they minimized the IPM sample-wise since $P(Z)$ is unknown. Similarly, [50] demonstrated that discretizing the treatment range and minimizing the maximum pair-wise IPM bounds the counterfactual error. Our method is different as we learn the balanced representation by minimizing the KL divergence.

2.3 Adversarial approach in Causal Inference

GANITE [55] and SCIGAN [7] employs a generative adversarial network (GAN) [18] to generate counterfactual outcomes, enabling estimation of individualized treatment effects in binary and continuous treatment setup. Moreover, learning a balanced (invariant) representation using an adversarial discriminator has been studied in the transfer learning literature to align source domain(s) to target domain(s) [17, 44, 49]. Similarly, in causal inference [12, 5] aimed to balance the distributions of two treatment groups adversarially. [6] extended the approach to the multiple time-varying treatment setting. However, the existing methods consider only scenarios with a finite number of treatment options and do not provide theoretical guarantees of their generalization capability.

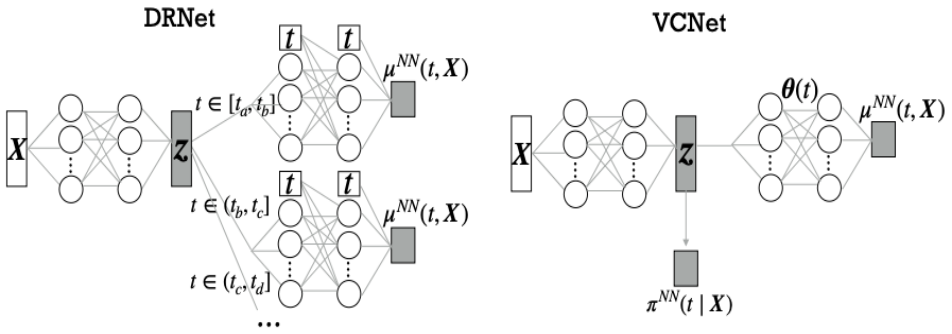

Figure 2.1: The figure is from [35] to compare the architecture of DRNet and VCNet. The left figure shows that DRNet handles continuous treatment by discretizing the treatment range and considering distinct regression head for each interval. The right figure shows that VCNet handles the treatment by incorporating it into the parameters $\theta(t)$.

Chapter 3

Problem Formulation & Theoretical Analysis

We assume a dataset of the form $D = \{x_i, t_i, y_i\}_{i=1}^N$, where $x_i \in \mathcal{X} \subseteq \mathbb{R}^d$ denotes the covariates of the ith unit, $t_i \in [0, 1]$ is the continuous treatment that unit i received, and $y_i \in \mathcal{Y} \subseteq \mathbb{R}$ denotes the outcome of interest for unit i after receiving treatment t_i. N is the total number of units, and d is the dimension size of covariates. We are interested in learning a machine learning model to predict the causal quantity $\mu(x, t) = E_{\mathcal{Y}}[Y(t)|X = x]$, which is the potential expected outcome under treatment t for the individual with covariates x. Note that, unlike binary ITE, the goal is to predict all potential outcomes, not just the difference between them. Similar to previous works, we rely on the following standard assumptions to make treatment effects identifiable from an observational dataset. We note that the validity of strong ignorability cannot be assessed from data, and must be determined by domain knowledge and understanding of the causal relationships between the variables.

[**Assumption 1 - Unconfoundedness**] $\{Y(t)\}_{t\in[0,1]} \perp\!\!\!\perp T \,|\, X$. Given covariates, treatment and the potential outcomes are independent. In other words, there is no hidden confounder, and observing X is sufficient to predict the potential outcomes $Y(t)$.

[**Assumption 2 - Overlap**] $P(T = t|X = x) > 0, \forall t \in [0, 1], \forall x \in X$. In words, every unit receives treatment level t with a probability greater than zero.

With the Assumption 1, we can condition on $T = t$ in the definition $\mu(x, t)$, and then $Y(t)$ would be equivalent to Y, the factual outcome seen in the observational dataset. It is now possible to predict potential outcome $\mu(x, t)$ for unit with covariates x and treatment t using a machine learning model. Assumption 2 is also needed since the machine learning model will produce some predicted outcome for any pairs, and the propensity score for them

should not be zero.

$$\mu(x,t) = E_{\mathcal{Y}}[Y(t)|X=x] = E_{\mathcal{Y}}[Y(t)|X=x,T=t] = E_{\mathcal{Y}}[Y|X=x,T=t]$$

However, the model will be trained with $p(x,t)$ distribution and need to perform well on $p(x)p(t)$ distribution. In other words, the model should be able to generalize to all counterfactuals. We aim to encode the covariates into a balanced latent representation and then use the representation and treatment to predict factual outcomes. We analyze the properties of encoder function $\phi : \mathcal{X} \to \mathcal{Z}$, where $\mathcal{Z}$ is the representation space, outcome prediction function $h : \mathcal{Z} \times [0,1] \to \mathcal{Y}$ and loss function $L : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}^+$.

[**Definition 1**] Define $\ell_{L,h,\phi}(x,t) = L(h(\phi(x),t),y) : \mathcal{X} \times [0,1] \to \mathbb{R}^+$ to be the unit-loss for a unit with covariate x that is intervened with treatment t. Unit-loss $\ell_{L,h,\phi}(x,t)$ measures loss L between the predicted outcome $\hat{y} = h(\phi(x),t)$ and the ground-truth outcome $y = \mu(x,t)$.

Using the definition of unit-loss, we can define the expected prediction error of some treatment t by marginalizing over the covariate distribution. As a result of treatment-treatment-selection bias, covariate distribution of samples having received treatment t (factual) and samples not having received treatment t (counterfactual) are different. We define factual error $\varepsilon_f^\ell(t)$ by marginalizing over $p(x|t)$ and counterfactual error $\varepsilon_{cf}^\ell(t)$ by marginalizing over $p(x)$ as follows.

$$\begin{aligned}
\varepsilon_f^\ell(t) &= \int_{\mathcal{X}} \ell_{L,h,\phi}(x,t)\, p(x|t)\, dx \\
\varepsilon_{cf}^\ell(t) &= \int_{\mathcal{T}'=[0,1]-\{t\}} \int_{\mathcal{X}} \ell_{L,h,\phi}(x,t) p(x|t')\, dx\, dt' \\
&= \int_{\mathcal{X}} \ell_{L,h,\phi}(x,t)\, p(x)\, dx
\end{aligned}$$

We also define the expected error of all treatments by marginalizing over their range $[0,1]$ as follows: $\varepsilon_f^\ell = \int_{[0,1]} \int_{\mathcal{X}} \ell_{L,h,\phi}(x,t)\, p(x,t)\, dx\, dt$ and $\varepsilon_{cf}^\ell = \int_{[0,1]} \int_{\mathcal{X}} \ell_{L,h,\phi}(x,t)\, p(x)\, p(t)\, dx\, dt$. Note that the expected factual error ε_f^ℓ integrates over joint distribution $p(x,t)$, and expected counterfactual error ε_{cf}^ℓ integrates over $p(x)\, p(t)$. We aim to reduce the distributional distance in representation space Z to ensure that minimizing ε_f^ℓ results in minimizing ε_{cf}^ℓ. We need the following two assumptions on encoder ϕ and unit-loss ℓ respectively to ensure that minimizing the distributional distance leads to minimizing the difference between counterfactual and factual errors, and the loss value is not arbitrary large.

[**Assumption 3**] The encoder function ϕ is a one-to-one mapping and representation space $\mathcal{Z}$ is the image of $\mathcal{X}$ under ϕ with the induced distribution $p_\phi(z)$.

[**Assumption 4**] Let G be a class of functions with infinity norm less than 1, $G = \{g : \mathcal{Z} \times [0,1] \to \mathbb{R}^+ \,|\, ||g||_\infty \leq 1\}$. Then, there exist a constant $C > 0$ such that $\frac{\ell_{L,h,\phi}(x,t)}{C} \in G$. This means for any (x,t) we have $\frac{\ell_{L,h,\phi}(x,t)}{C} \leq 1$.

Note that Assumptions 3 is common in balanced representation learning works [42, 3, 22], and instantiating any distributional distance from $IPM_G(p,q) = \sup_{g \in G} \int_S g(s)(p(s) - q(s))ds$ where p and q are two probability distributions, requires specifying G class of functions. Now we present our main theoretical results which demonstrates a bound on the expected counterfactual error ε^ℓ_{cf} consisting of the expected factual error ε^ℓ_f and the KL divergence between distributions $p_\phi(z)\, p(t)$ and $p_\phi(z,t)$. Note that the KL divergence is non-negative and becomes zero if and only if two distributions are the same. Therefore, $D_{KL}\Big(p_\phi(z)p(t)||p_\phi(z,t)\Big) = 0$ implies $p_\phi(z,t) = p_\phi(z)\, p(t)$ and $\varepsilon^\ell_f = \varepsilon^\ell_{cf}$. In other words, in an randomized controlled trial setting, where tx, the KL divergence term is 0, and our bound naturally reduces to a standard learning problem of learning a function to minizie factual outcomes.

[**Proposition 1 - Counterfactual Generalization Bound**] *Given the one-to-one encoder function $\phi : \mathcal{X} \to \mathcal{Z}$, the outcome prediction function $h : \mathcal{Z} \times [0,1] \to \mathcal{Y}$, and the unit-loss function $\ell_{L,h,\phi}(x,t)$ that satisfies Assumption 4,*

$$\varepsilon^\ell_{cf} \leq \varepsilon^\ell_f + C\sqrt{2\, D_{KL}\Big(p_\phi(z)p_t) \,||\, p_\phi(z,t)\Big)}$$

Proof.

$$\varepsilon^\ell_{cf} - \varepsilon^\ell_f = \int_{[0,1]} \int_{\mathcal{X}} \ell_{L,h,\phi}(x,t)\, [p(x)p(t) - p(x,t)]\, dxdt \tag{3.1}$$

$$= \int_{[0,1]} \int_{\mathcal{Z}} \ell_{L,h,\phi}(\psi(z),t)\, [p(\psi(z))p(t) - p(\psi(z),t)] J_\psi J_\psi^{-1} d\psi(z)dt \tag{3.2}$$

$$= \int_{[0,1]} \int_{\mathcal{Z}} \ell_{L,h,\phi}(\psi(z),t)\, [p_\phi(z)p(t) - p_\phi(z,t)]\, dzdt \tag{3.3}$$

$$\leq \int_{[0,1]} \int_{\mathcal{Z}} C|p_\phi(z)p(t) - p_\phi(z,t)|dzdt \tag{3.4}$$

$$\leq C\sqrt{2 \int_{[0,1]} \int_{\mathcal{Z}} p_\phi(z)p(t) \log\Big(\frac{p_\phi(z)p(t)}{p_\phi(z,t)}\Big)dzdt} \tag{3.5}$$

$$= C\sqrt{2\, D_{KL}\Big(p_\phi(z)p(t)||p_\phi(z,t)\Big)} \tag{3.6}$$

where in equality (3.2) J is the Jacobiant of $\psi(z)$ which cancels with the inverse Jacobian that appears after the change of variables in the differential term, the equality (3.3) holds by the reparameterization $x = \psi(z)$, inequality (3.4) holds by Assumption 4 constraining the function ℓ, and the last two inequalities are based on Pinkser's inequality $\int |p - q| = 2TV_D(p, q) \leq \sqrt{2D_{KL}(p, q)}$. □

For some applications, one might be interested in the treatment effect between two different treatments rather than predicting all counterfactual outcomes. For instance, in binary treatment setting it is standard to report the model performance in terms of precision of estimating heterogeneous effect (PEHE) [23] which measures the squared difference between ground-truth treatment effect $\tau(x) = \mu(x, 1) - \mu(x, 0)$ and predicted treatment effect $\hat{\tau}(x) = h(\phi(x), 1) - h(\phi(x), 0)$. We define the continuous counterpart $\varepsilon_{pehe}(t_1, t_2)$ between two treatment levels t_1 and t_2 and present an upper bound on it in Proposition 2.

[**Definition 2**] Define $\varepsilon_{pehe}(t_1, t_2) = \int_{\mathcal{X}} \left[(\mu(x, t_1) - \mu(x, t_2)) - (h(\phi(x), t_1) - h(\phi(x), t_2))\right]^2 p(x)dx$ to be expected precision of estimating heterogeneous effect between treatments t_1 and t_2.

[**Proposition 2 - Precision of Estimating Heterogeneous Effect Bound**] *Given the one-to-one encoder function ϕ and outcome prediction function h as in Proposition 1, and a unit-loss function $\ell_{L,h,\phi}(x, t)$ that satisfies Assumption 4 and its associated L is squared error $||.||^2$,*

$$\varepsilon_{pehe}(t_1, t_2) \leq \varepsilon_f^{\ell}(t_1) \; + \; \varepsilon_f^{\ell}(t_2) + \\ C\left[\sqrt{2\, D_{KL}\Big(p_\phi(z) \,||\, p_\phi(z|t_1)\Big)} + \sqrt{2\, D_{KL}\Big(p_\phi(z)||p_\phi(z|t_2)\Big)}\right]$$

Proof. In order to prove Proposition 2, we first prove $\varepsilon_{cf}^{\ell}(t) \leq \varepsilon_f^{\ell}(t) + C\sqrt{2\, D_{KL}\Big(p_\phi(z)||p_\phi(z|t)\Big)}$. Let $\psi : \mathcal{Z} \to \mathcal{X}$ be the inverse of ϕ.

$$\varepsilon_{cf}^{\ell}(t) - \varepsilon_f^{\ell}(t) = \int_{\mathcal{X}} \ell_{L,h,\phi}(x, t) \left[p(x) - p(x|t)\right] dx \tag{3.1}$$

$$= \int_{\mathcal{Z}} \ell_{L,h,\phi}(\psi(z), t) \left[p_\phi(z) - p_\phi(z|t)\right] dz \tag{3.2}$$

$$\leq \int_{\mathcal{Z}} C|p_\phi(z) - p_\phi(z|t)| \tag{3.3}$$

$$\leq C\sqrt{2 \int_{\mathcal{Z}} p_\phi(z) \log\left(\frac{p_\phi(z)}{p_\phi(z|t)}\right)} \tag{3.4}$$

$$= C\sqrt{2\, D_{KL}\Big(p_\phi(z)||p_\phi(z|t)\Big)} \tag{3.5}$$

Observe the difference between above bound and Proposition 1. Here the counterfactual error is restricted to a treatment value, and thus it is bounded by the factual error of that specific treatment, and the resulting shift from it. The proofs are similar.

$$\varepsilon_{pehe}(t_1, t_2) = \int_{\mathcal{X}} \left[(\mu(x, t_1) - \mu(x, t_2)) - (h(\phi(x), t_1) - h(\phi(x), t_2)) \right]^2 p(x) dx \tag{3.6}$$

$$\leq \int_X \left(\mu(x, t_1) - h(\phi(x), t_1)\right)^2 p(x) dx + \int_X \left(\mu(x, t_2) - h(\phi(x), t_2)\right)^2 p(x) dx \tag{3.7}$$

$$= \varepsilon^{\ell}_{cf}(t_1) + \varepsilon^{\ell}_{cf}(t_2) \tag{3.8}$$

$$\leq \varepsilon^{\ell}_{f}(t_1) \; + \; \varepsilon^{\ell}_{f}(t_2) \; + \; C \left[\sqrt{2\, D_{KL}\Big(p_\phi(z) || p_\phi(z|t_1) \Big)} + \sqrt{2\, D_{KL}\Big(p_\phi(z) || p_\phi(z|t_2) \Big)} \right] \tag{3.9}$$

where the inequality (3.7) is by triangle inequality, and the last two lines hold by the definition of counterfactual error and the above result respectively. □

Chapter 4

Methodology

Based on Proposition 1, we can reduce the counterfactual error by learning an encoder function ϕ and an outcome prediction function h that jointly minimize distribution shift and factual outcome error. We propose our method that implements functions ϕ and h using neural networks and is trained with an objective function inspired by Proposition 1. We note that although our theoretical analysis is based on encoder invertibility assumption, we achieved better results with a more complex non-linear network compared to linear invertible encoders. We note that this is a gap between theory and practice. Figure 4.1 illustrates the architecture of the ACFR network, consisting of an encoder network ϕ, an outcome prediction network h, and a treatment prediction network π. The key aspects of ACFR, distribution shift minimization and minimization of outcome prediction error, are presented in the following.

4.1 Distribution Shift Minimization

As discussed earlier, in order to minimize distribution shift we aim to minimize the KL divergence term with respect to encoder ϕ. The KL divergence can be rewritten as $H(T) - H(T|Z;\phi)$ where $H(T|Z;\phi)$ is the conditional entropy. Marginal entropy $H(T)$ does not depend on ϕ, thus minimizing KL divergence is equivalent to maximizing the conditional entropy $H(T|Z;\phi) = \mathbb{E}[\log p_\phi(t|z)]$. However, as $p_\phi(t|z)$ is intractable we introduce variational distribution $q_\pi(t|z)$ parameterized with π defined over the same space to approximate it. For any variational distribution $q_\pi(t|z)$ the following holds [15].

$$\max_\phi H(T|Z;\phi) = \max_\phi \inf_\pi \mathbb{E}_{p_\phi(t,z)}[-\log q_\pi(t|z)]$$

We assume the distribution $q_\pi(t|z)$ is a normal distribution with a fixed variance. We can estimate the mean of $q_\pi(t|z)$ by a neural network called treatment-prediction network π. By approximating $p_\phi(z,t)$ with empirical data, we derive the following mean squared adversarial

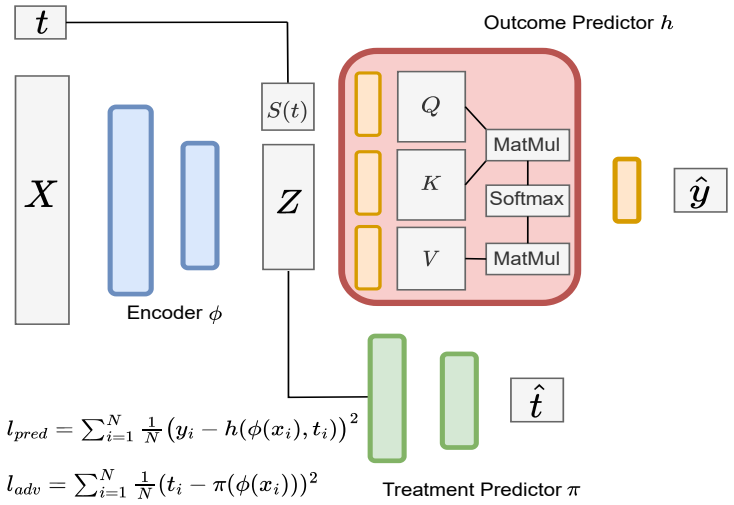

Figure 4.1: The architecture of Adversarial CoutnerFactual Regression Network consisting of three sub-networks encoder ϕ, outcome predictor h, and treatment predictor π. Networks ϕ and h are trained to minimize the outcome prediction loss l_{pred}, and networks ϕ and π are trained to maximize / minimize adversarial loss l_{adv}. The encoder and treatment predictor are implemented using linear layers, and the outcome predictor network consists of a cross-attention module followed by a linear layer.

loss term from the above negative log likelihood.

$$l_{adv} = \max_{\phi} \min_{\pi} \sum_{i=1}^{N} \frac{1}{N}(t_i - \pi(\phi(x_i)))^2$$

Specifically, the treatment-predictor network π (The green network in Figure 4.1) is trained to minimize l_{adv} by predicting treatment t from representation $z = \phi(x)$. The encoder network ϕ (The blue network in Figure 4.1) is trained to maximize l_{adv} by extracting z in such a way that the assigned treatment t is not distinguishable. Therefore KL divergence can be estimated and minimized using two networks and an adversarial loss. Through alternating optimization with respect to ϕ and π, and assuming that the treatment predictor π reaches the optimum in each iteration, the resulting representation z has a desired property: $\mathbb{E}[t|z] = \mathbb{E}[t]$ [49]. This implies that knowing latent representation z does not provide additional information for predicting the expected treatment.

4.2 Factual Outcome Error Minimization

In this section, we discuss the minimization of the factual prediction error ϵ_f^ℓ. Recall that:

$$\begin{aligned}\epsilon_f^\ell &= \int_{[0,1]} \int_{\mathcal{X}} \ell_{\phi,h}(x,t)\, p(x,t)\, dx\, dt \\ &= \int_{[0,1]} \int_{\mathcal{X}} L(h(\phi(x),t),y)\, p(x,t)\, dx\, dt\end{aligned}$$

Here, outcome y is a continuous variable, and we consider L to be the squared loss. By approximating $p(x,t)$ with empirical distribution, we derive the following outcome prediction loss that needs to be minimized with respect to ϕ and h:

$$l_{pred} = \min_{\phi,h} \sum_{i=1}^{N} \frac{1}{N} \big(y_i - h(\phi(x_i), t_i)\big)^2$$

The encoder network ϕ is as defined in the previous section. The outcome prediction network h, however, needs to be particularly designed to maintain the treatment impact on the outcome. We aim to obtain an informative embedding for treatment value, and similar to [57] predict the outcome from the embedding and representation using an attention-based network. [57] proposed to learn the embedding by a neural network. While neural networks are universal function approximators, it has been shown that they can not extract an expressive embedding from a scalar value due to optimization difficulties [19]. We construct the treatment embedding applying a set of predefined spline functions to the treatment t shown as $S(t) = [s_1(t), s_2(t), \ldots, s_m(t)]$ in Figure 1. Spline functions have been shown to be able to approximate a function in a piece-wise manner [13]

The treatment embedding and the representation are then passed to the cross attention layer (red module in Figure 1) to learn the dependency between treatment and representation. A cross-attention layer has three matrices query Q, key K, and value V, where Q is learned from treatment embedding using h_q parameter and K and V are learned from the representation using h_k and h_v parameters respectively. The output of the cross-attention layer is $\sigma(\frac{Q^T K}{\sqrt{d_k}})V$ where d_k is the dimension of the Q and K matrices. We then predict the outcome $\hat{y}$ by a linear layer after the cross attention module.

Unlike ad-hoc architectures VCNet [35] and DRNet [41], our outcome prediction network is flexible in terms of the number of spline functions. We can incorporate as many splines as needed without increasing the number of model parameters. This is particularly important in individual effect estimation, because each individual responds differently to a given treatment, and hence different spline functions might be necessary to approximate the treatment-response function for different patients. The proposed architecture can incor-

porate a large number of spline functions, and the attention layer learns how relevant each spline is for estimating each patient treatment-response function. It is also worth mentioning that by setting h_q to the identity, h_k to the unity and parameterizing h_v with a neural network (which are sub-optimal choices) we recover VCNet and DRNet with a cross-attention layer.

4.3 Adversarial Counterfactual Regression

In order to minimize the distribution shift, we derived the adversarial loss l_{adv} and introduced the networks encoder ϕ and treatment-prediction network π to optimize the loss. Similarly, in order to predict factual outcome we derived outcome prediction loss l_{pred} and introduced its associated attention based prediction network h to minimize it. Now, we aim to train the entire network to optimize the following objective function as Proposition 1 suggests:

$$\mathcal{L}_{ACFR} = \max_{\pi} \min_{\phi,h} l_{pred} - \gamma l_{adv}$$

where γ is a tunable parameter. Algorithm 1 presents pseudo-code for the training of the ACFR network. At each iteration a batch of samples is given as input to the network (line 3). At the first stage ACFR predicts the treatment using encoder and treatment-prediction networks, computes l_{adv} and only updates parameters of π for M iterations (line 5-10). At the second stage, ACFR predicts the outcome from the encoded representation of the batch using outcome prediction network h , computes l_{pred} and updates h and ϕ with respect to l_{pred} and $l_{pred} - \gamma\, l_{adv}$ losses (line 11-14). Finally, the parameters of the encoder and the prediction networks are returned for the inference phase (line 15).

Algorithm 1: Adversarial Counterfactual Regression

Input: Factual samples $(x_i, t_i, y_i)_{i=1}^{N}$, encoder network with initial parameter ϕ_0, treatment-predictor network with initial parameters π_0, hypothesis network with initial parameters h_0, batch size b, iteration number T, inner loop size M, trade-off parameter γ, and the step sizes η_1 and η_2.

for $t \leftarrow 0$ **to** $T-1$ **do**

  Sample a mini-batch: $B = \{i_1, i_2, ..., i_b\} \subset \{1, 2, ..., N\}$

  Encode into latent representation: $z_B = \phi_t(x_B)$

  Initialize $\omega_0 = \pi_t$

  for $m \leftarrow 0$ **to** $M-1$ **do**

    Compute l_{adv} and update ω_m

    $\hat{t}_B = \omega_m(z_B) \quad l_{adv} = \frac{1}{b}\sum_{i \in B}(t_i - \hat{t_i})^2$

    $\omega_{m+1} = \omega_m - \eta_2 \nabla_\omega l_{adv}(\omega_m)$

  $\pi_{t+1} = \omega_m$

  Compute l_{pred} and update ϕ_t and h_t:

  $\hat{y} = h_t(z_B) \quad l_{pred} = \frac{1}{b}\sum_{i \in B}(y_i - \hat{y_i})^2$

  $\phi_{t+1} = \phi_t - \eta_1\big(\gamma * \nabla_\phi l_{adv}(\phi_t) - \nabla_\phi l_{pred}(\phi_t)\big)$

  $h_{t+1} = h_t - \eta_1 \nabla_h l_{pred}(h_t)$

Return ϕ_T and h_T

Chapter 5

Experiments

Treatment effect estimation methods have to be evaluated for predicting potential outcomes including counterfactuals which are unavailable in real-world observational datasets. Therefore, synthetic or semi-synthetic datasets are commonly used since their treatment assignment mechanism and outcome function are known and hence counterfactual outcomes can be generated. Note that this does not change the fact that only factual outcomes are accessible during training and the methods need to mitigate the covariate shift. In this section, we first describe the data generation process and the implementation of the baselines, and then present our results.

5.1 Setup

5.1.1 Observational data generation

We used TCGA [34] and News [28] semi-synthetic datasets. TCGA dataset consists of gene expression measurements of the 4000 most variable genes for 9659 cancer patients. The News dataset which was introduced as a benchmark in [28] consists of 3477 word counts for 5000 randomly sampled news items from the NY times corpus. For each dataset, we first normalized each covariate and then scaled every sample to have a norm 1. We then split the datasets with 68/12/20 ratio into training, validation, and test sets. We followed treatment and outcome generating process of [7] to ensure results are not affected by our own data generation. The distribution for treatment assignment and function for outcome generation are summarized in Table 5.1. The parameter α in Beta distribution of treatment assignment determines the treatment-selection bias level (α is set 2 in all experiments unless otherwise stated) and v_1, v_2 and v_3 are vectors whose entries are sampled from the standard normal distribution $\mathcal{N}(0, 1)$, and then are normalized. Based on Table 5.1, we assigned the treatment and factual outcome for all samples in the training and validation sets. All methods are then trained on the training set, and the validation set has been used for hyperparameter selection. Same as [7], counterfactual outcomes for a unit are generated using the outcome

function given the unit's covariates and 65 grids in the range $[0, 1]$ as an approximation of the treatment range since we want our model to generalize uniformly well.

Dataset	#Samples	#Covariates	Outcome function	Treatment assignment
TCGA	9659	4000	$y = 10(v_1^T x + 12 v_2^T x t - 12 v_3^T x t^2)$	$t = \text{Beta}(\alpha, \beta)$
News	5000	3477	$y = 10(v_1^T x + \sin(\frac{v_2^T x}{v_3^T x} \pi t))$	$\beta = \frac{2(\alpha-1) v_2^T x}{v_3^T x} + 2 - \alpha$

Table 5.1: Datasets and data generating functions. The treatment $t \in [0, 1]$ comes from a Beta distribution which depends on the covariates x, and potential outcome y is determined with x and t. The parameter α controls both parameters of treatment assignment distribution and the shape of the distribution, and big α leads to high selection bias.

5.1.2 Baselines

We consider two state-of-the-art architectures for the outcome prediction network proposed in DRNet [41] and VCNet [35], which are able to handle continuous treatments. We also consider two metrics of the IPM class, Hilbert-Schmidt independence criterion (HSIC) and Waaserstein (Wass) [47] distances for minimizing the distribution shift. This results in 4 methods DRNet-HSIC, DRNet-Wass, VCNet-HSIC, and VCNet-Wass. We then consider SCIGAN [7] as a state-of-the-art generative model for continuous treatments. Moreover, we compare against ADMIT [50] network with their own proposed algorithm to estimate the IPM distance, resulting in ADMIT-Wass and ADMIT-HSIC methods. Finally we include a Generalized Proposensity Score (GPS) method and a MLP network as baselines.

IPM Minimization

We discuss two IPM minimization techniques proposed in [3] and [50] respectively.
1) We approximate the IPM distance between $P(Z, T)$ and $P(Z, T)$ as follows. For each sample in a batch, we minimize the distance of the joint distribution of that sample $p(z_i, t_i)$ and the joint distribution of all other samples $p(z_j, t_j)$ in that batch. This involves computing the IPM term once for each sample as follows:

$$\frac{1}{N} \sum_{i=1}^{N} IPM\Big(p\{z_i, t_i\}, p\{z_i, t_j\}_{j:j \neq i}\Big)$$

2) We first divide the treatment range into l equal intervals. For the distribution of each interval $p\Delta_i$, we compute the IPM distance of that distribution and the distribution of all

other intervals. We then minimize the maximum distance among all as follows

$$\frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{l}\max\Big(IPM(p\Delta_k, p\Delta_{j:j\neq k})\Big)$$

where $p\Delta_k$ is the distribution of kth interval, and $p\Delta_j$ is the distribution of the interval with maximum distance from Δ_k. We use two classes of IPM families called Hilbert Schmidt Independence Criterion (HSIC) and Wasserstein distances.

5.1.3 Implementation

We discuss the implementation of ACFR and baselines as well as the set of hyperparameters used for each method.

Multi-Layer Perceptron

We construct an MLP network using feedforward layers. We vary the number of nodes in each layer $\in \{50, 100, 200\}$ and the number of hidden layers $\in \{1, 2, 3\}$. The network takes the concatenation of covariates and treatment as input and predicts the factual outcome without any attempt to reduce the covariate shift. The objective is minimizing the mean squared error between the ground truth and the predicted outcome.

Dose-Response Network + IPM

The original implementation of DRNet considered the treatment variable as a pair of a medication and a dosage, where the medication is categorical and the dosage is a continuous variable. We adjusted the architecture and algorithm for continuous treatment. To minimize the distribution distance in the representation space, we minimize the IPM distance using the first procedure described above. We also use 5 distinct regression heads for the samples in 5 equal intervals: $[0, 0.2], [0.2, 0.4], \ldots, [0.8, 1]$. Each regression head and the representation network consist of feedforward layers, and similar to the MLP architecture, the treatment value is given to each regression head as input. For the representation network, we vary the number of nodes $\in \{50, 100, 200\}$ and layers $\in \{1, 2, 3\}$, and for regression heads we vary nodes $\{50, 100\}$ and layers $\{0, 1\}$. The weight of the IPM loss term is $\{10^{-3}, 10^{-2}, 10^{-1}, 1\}$ for News and $\{10^{-2}, 10^{-1}, 1, 10\}$ for the TCGA dataset.

Varying Coefficient Network + IPM

We adjusted the implementation of VCNet, which was originally proposed for average treatment effect and learned the representation based on propensity score. The adjusted VCNet has two sub-networks. The representation network consists of feedforward layers, and the outcome prediction network is constructed by involving the treatment value into the network parameters. Specifically, we consider a set of spline functions with degree 2 and knots

$[1/3, 2/3]$ and use 5 heads, each associated with one spline function. The output of the dynamic network is the linear combination of spline functions where the weights are the output of the regression heads. To minimize the distribution distance in the representation space, we minimize the IPM distance using the first technique explained above. The hyper-parameters used for representation network and regression heads in dynamic network is same as DRNet method. The weight of the IPM loss term is $\{10^{-2}, 10^{-1}, 1\}$ for News and $\{10^{-2}, 10^{-1}, 1, 10\}$ for the TCGA dataset.

ADMIT

The IPM minimization procedure in ADMIT is based on the second technique. ADMIT has three sub-networks: representation network, re-weighting network, and hypothesis network. For the representation network, we use feedforward layers, and for the last two, we use the dynamic network proposed by [35] as described above, in order to maintain treatment impact as they suggested. The hyper-parameter range for layers, nodes and spline functions are same as VCNet. The weight of the IPM loss term is $\{10^{-2}, 10^{-1}, 1\}$ for both datasets. Also, we vary the number of intervals: $\{3, 4, 5\}$.

Adversarial Counterfactual Regression

For the representation and treatment-predictor networks, we use feedforward layers. For both networks we vary the number of nodes $\in \{50, 100, 200\}$ and the number of hidden layers $\in \{0, 1, 2\}$. We also vary the knots $\in \{\{1/3, 2/3\}, \{1/4, 2/4, 3/4\}, \{1/5, 2/5, 3/5, 4/5\}\}$ and the degree $\in \{2, 3, 4\}$ in order to obtain spline functions of treatment and construct the treatment embedding. In principle any parameterization can be used to construct the matrices of attention block, but we used the common linear parameterization with dimension $\{32, 64\}$. We vary the dimension of the last layer $\{16, 32\}$. The trade-off parameter in ACFR objective function, $\gamma \in \{10^{-2}, 10^{-1}, 1, 10\}$, and we vary the number of inner loops, $M \in \{1, 10, 100\}$, for optimizing the treatment predictor.

The step size $\in \{10^{-5}, 10^{-4}, 10^{-3}\}$ and the batch size $\in \{32, 64\}$ are shared between all methods. For the Generalized Propensity Score (GPS), we employ the implementation in [30] for continuous outcomes, and adjust the implementation of SCIGAN [7] to support single continuous treatment. Also, in order to select the best set of hyperparameters, we use a baysian approach, tree-structured parzen estimator (TPE) from the Optuna package in Python.

5.1.4 Metrics

Having $\mu(x, t)$ as the ground-truth outcome of the unit with covariate x under treatment t and $f(x, t)$ as the predicted outcome, we report the performance of methods in terms of

the two following metrics defined in [41]. The Mean Integrated Squared Error (MISE) is the squared error of the predicted outcome averaged over all treatment values and all units. MISE is useful when we have no preference for some specific treatment (or some patient) and performance on all potential outcomes are equally important for us. We assumed the marginal distribution of the treatment to be uniform, $p(t) \sim U(0, 1)$.

$$\text{MISE} = \frac{1}{N}\Sigma_{i=1}^{N}\int_0^1 [\mu(x_i, t) - f(x_i, t)]^2 dt$$

In some applications such as precision medicine, one might only be interested in having a predictive model that has a policy close to optimal since we want to prescribe the best treatment. In order to compare different predictive models in terms of their predicted policy, Policy Error (PE) metric can be used. The Policy Error (PE) measures the average squared error of estimated optimal treatment, where t_i^* and $\hat{t}_i^*$ denote ground-truth and predicted best treatments respectively.

$$\text{PE} = \frac{1}{N}\Sigma_{i=1}^{N}[\mu(x_i, t_i^*) - \mu(x_i, \hat{t}_i^*)]^2$$

5.2 Results

We performed two sets of experiments for potential outcome prediction, called out-of-sample prediction and within-sample prediction. The out-of-sample experiment shows the ability of models in predicting the potential outcomes for units in the held-out test set, and the within-sample experiment shows the ability for units in the training set.

5.2.1 Prediction error

For all methods, we reported the mean and the standard deviation of MISE and PE in the format of mean±std over 20 realizations of each dataset. Table 1 shows that on TCGA dataset, ACFR outperformed the baselines in both metrics on average, and on News dataset, ACFR acheived the best and second best average result in terms of MISE and PE metrics respectively. We achieved similar results in with-in-sample experiments. We can also see the substantial gain of cross-attention layer in the performance of ACFR by comparing it with ACFR w/o attention, demonstrating the effectiveness of proposed outcome prediction network. Comparably, DRNet and VCNet methods have more parameters while ACFR and ADMIT methods are more time-consuming because of their corresponding inner loop to minimize the distribution shift.

5.2.2 treatment-selection bias robustness

We also investigate the robustness of 4 methods (ACFR, VCNet-HSIC, DRNet-HSIC, and ADMIT-HSIC) against varying level of treatment-selection bias. As mentioned earlier, the

Method	News		TCGA	
	MISE	PE	MISE	PE
GPS	3.21 ± 0.34	0.39 ± 0.03	6.50 ± 1.21	2.30 ± 0.27
MLP	2.91 ± 0.33	0.31 ± 0.02	4.81 ± 0.54	1.15 ± 0.23
DRNet-HSIC	1.59 ± 0.20	0.21 ± 0.01	2.03 ± 0.27	1.24 ± 0.23
DRNet-Wass	1.64 ± 0.21	0.21 ± 0.01	2.01 ± 0.20	1.29 ± 0.21
VCNet-HISC	1.28 ± 0.10	$\mathbf{0.16 \pm 0.01}$	1.99 ± 0.11	0.94 ± 0.14
VCNet-Wass	1.43 ± 0.11	0.17 ± 0.01	1.76 ± 0.12	0.92 ± 0.14
ADMIT-HSIC	1.25 ± 0.12	0.18 ± 0.01	1.81 ± 0.23	0.86 ± 0.15
ADMIT-Wass	1.35 ± 0.20	0.18 ± 0.01	1.67 ± 0.23	0.81 ± 0.14
SCIGAN	1.21 ± 0.15	0.20 ± 0.01	1.85 ± 0.14	0.97 ± 0.14
CFR-Wass	1.43 ± 0.15	0.20 ± 0.01	1.85 ± 0.17	0.96 ± 0.17
CFR-HSIC	1.49 ± 0.15	0.19 ± 0.01	1.76 ± 0.20	0.85 ± 0.16
ACFR w/o attention	1.58 ± 0.15	0.19 ± 0.01	1.86 ± 0.21	1.01 ± 0.15
ACFR	$\mathbf{1.12 \pm 0.12}$	0.18 ± 0.01	$\mathbf{1.60 \pm 0.20}$	$\mathbf{0.76 \pm 0.12}$

Table 5.2: Results on News and TCGA datasets for the out-of-sample setting.

α parameter of Beta distribution in the treatment generating function controls the amount of treatment-selection bias. As α increases the treatment-selection bias and covariate shift of the observational dataset increase and consequently, we expect the error of methods to increase as well. As shown in Figure 5.1, ACFR performs consistently and has a notable gap with the baselines at the strong treatment-selection bias level ($\alpha = 6$) in terms of MISE for the out-of-sample setting.

5.2.3 Adversarial loss effect

Figure 5.2 demonstrates that the representation learned through KL divergence minimization is less predictive of the treatment value compared to representations obtained from two classes of IPM, HSIC [21] and Wasserstein [47], thereby showing more effective reduction of the shift. Also ss mentioned before, we tuned the hyper-parameter controlling the adversarial loss. Figure 5.3 shows the effect of adversarial loss where ($\gamma = 8$) performs better than no adversarial loss $\gamma = 0$.

Method	News		TCGA	
	MISE	PE	MISE	PE
GPS	3.08 ± 0.33	0.36 ± 0.02	6.25 ± 0.97	1.95 ± 0.29
MLP	2.79 ± 0.32	0.31 ± 0.02	4.72 ± 0.65	1.27 ± 0.23
DRNet-HSIC	1.32 ± 0.20	0.20 ± 0.01	1.91 ± 0.25	1.04 ± 0.19
DRNet-Wass	1.34 ± 0.21	0.19 ± 0.01	1.88 ± 0.20	1.04 ± 0.21
VCNet-HISC	1.18 ± 0.11	0.16 ± 0.01	1.59 ± 0.12	0.87 ± 0.14
VCNet-Wass	1.23 ± 0.10	0.13 ± 0.01	$\mathbf{1.39 \pm 0.11}$	0.82 ± 0.14
ADMIT-HSIC	1.12 ± 0.12	$\mathbf{0.12 \pm 0.01}$	1.71 ± 0.23	0.76 ± 0.15
ADMIT-Wass	1.20 ± 0.10	0.13 ± 0.01	1.46 ± 0.21	0.79 ± 0.13
SCIGAN	1.15 ± 0.11	0.16 ± 0.01	1.58 ± 0.21	0.86 ± 0.10
CFR-Wass	1.20 ± 0.11	0.18 ± 0.01	1.57 ± 0.19	0.89 ± 0.10
CFR-HSIC	1.25 ± 0.14	0.18 ± 0.01	1.61 ± 0.18	0.71 ± 0.07
ACFR w/o attention	1.34 ± 0.13	0.17 ± 0.01	1.66 ± 0.20	0.92 ± 0.13
ACFR	$\mathbf{0.95 \pm 0.12}$	0.15 ± 0.01	1.42 ± 0.22	$\mathbf{0.62 \pm 0.11}$

Table 5.3: Results on News and TCGA datasets for the within-sample setting.

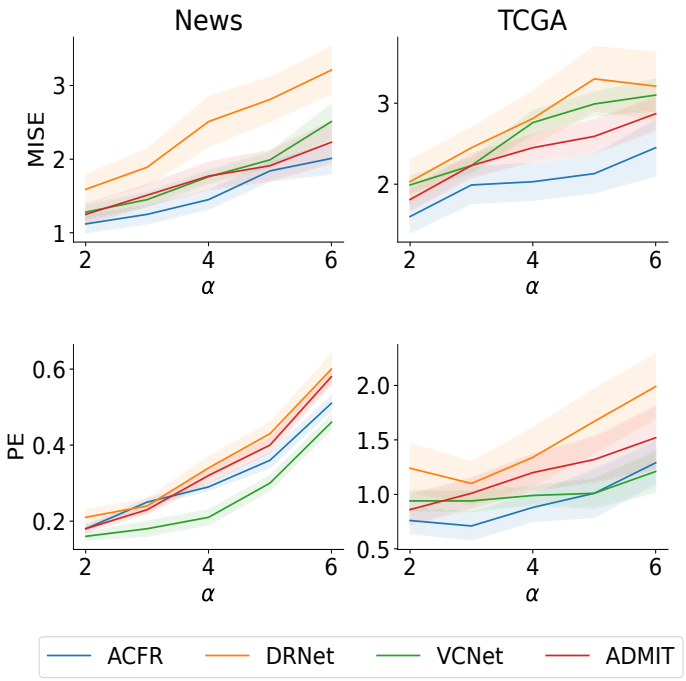

Figure 5.1: Robustness of ACFR against varying level of treatment-selection bias determined by α parameter of treatment assignment distribution. ACFR demonstrates a robust performance in terms of MISE and PE compared to baselines.

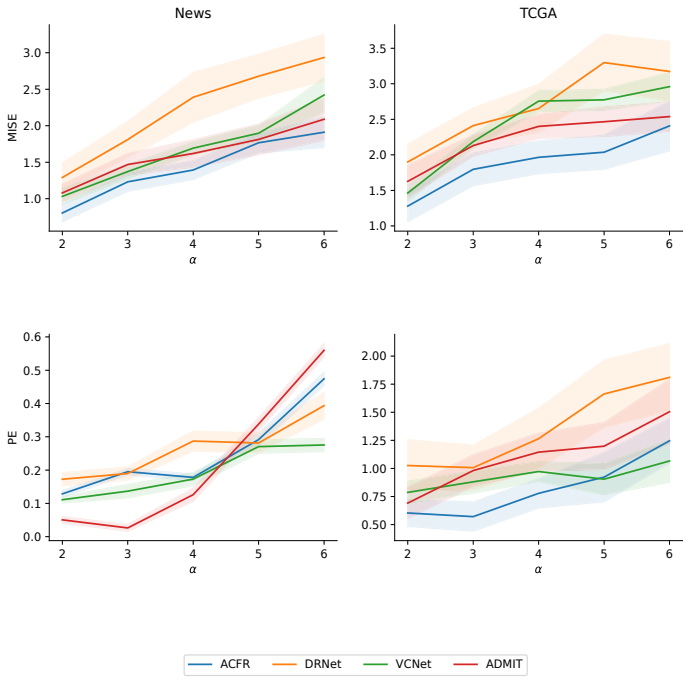

Figure 5.1: Robustness of ACFR against varying level of treatment-selection bias in within-sample setting determined by α parameter of treatment assignment distribution. ACFR demonstrates a robust performance in terms of MISE and PE compared to baselines.

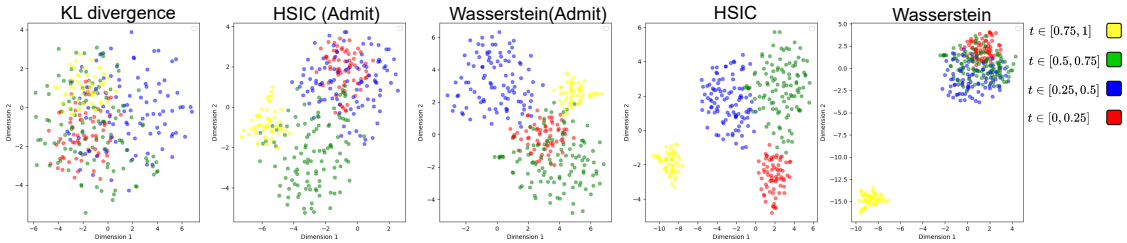

Figure 5.2: Tsne plot of latent representation Z learned using different distributional distances. After training each method on News dataset, we mapped validation samples into latent representation and plotted them using 2d tsne. We categorized the samples into 4 intervals with respect to their assigned treatment value and each interval corresponds to a color. We consider two important classes of IPM metrics, HSIC and Wasserstein. The treatment value is less distinguishable in the KL divergence representation followed by IPM-ADMIT (minimization with the algorithm proposed in [50]), and IPM (minimization with the procedure proposed in [3])

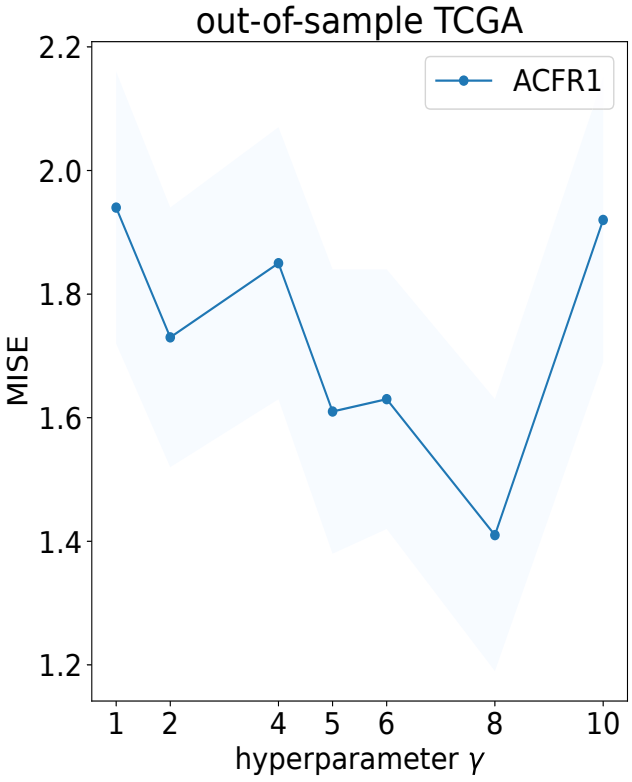

Figure 5.3: The performance of Adversarial Counterfactual Regression with varying level of γ, the hyper-parameter controlling the adversarial loss.

Chapter 6

Conclusion

Deep learning has shown great promise for the treatment effect estimation problem because it can extract a balanced representation of covariates. In this thesis, we investigated the problem of treatment effect estimation for a continuous treatment setting. We discussed the challenges of using existing methods and introduced Adversarial Counter-Factual Regression (ACFR). We proved a new bound on the counterfactual prediction error using the KL divergence instead of an IPM distance, which has the benefit that the KL divergence can be estimated parametrically and results in a more reliable bound. Based on the error bound, ACFR uses an adversarial neural network architecture to minimize the KL divergence of the representations and a cross-attention network to minimize the factual prediction error. Our experimental evaluation on semi-synthetic datasets has demonstrated the superiority of ACFR over several state-of-the-art methods.

It is worth mentioning that the theoretical analysis is not restricted to continuous treatments, and in future work, we plan to extend and evaluate the ACFR framework for structured and time series treatments. Moreover, we can enhance the invertibility property of the encoder, using techniques like normalizing flows or a decoder to encourage better reconstruction. Additionally, we can provide a more accurate estimation of conditional entropy and minimize it more effectively by estimating other moments of the conditional distribution, such as variance, in future works when the number of samples is sufficient. Our work also relies on the unconfoundedness assumption which might be untenable in practice. In future works, we aim to use generative models such as variational autoencoders (VAEs) to infer the non-linear relationships between the observed confounders, latent confounders, treatment assignment, and outcomes.

Bibliography

[1] Ahmed M. Alaa and Mihaela van der Schaar. Bayesian inference of individualized treatment effects using multi-task gaussian processes, 2017.

[2] Mohammad Taha Bahadori, Eric Tchetgen Tchetgen, and David E. Heckerman. End-to-end balancing for causal continuous treatment-effect estimation, 2022.

[3] Alexis Bellot, Anish Dhir, and Giulia Prando. Generalization bounds and algorithms for estimating conditional average treatment effect of dosage, 2022.

[4] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79:151–175, 2010.

[5] Jeroen Berrevoets, James Jordon, Ioana Bica, alexander gimson, and Mihaela van der Schaar. Organite: Optimal transplant donor organ offering using an individual treatment effect. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 20037–20050. Curran Associates, Inc., 2020.

[6] Ioana Bica, Ahmed M. Alaa, James Jordon, and Mihaela van der Schaar. Estimating counterfactual treatment outcomes over time through adversarially balanced representations. 2020.

[7] Ioana Bica, James Jordon, and Mihaela van der Schaar. Estimating the effects of continuous-valued interventions using generative adversarial networks, 2020.

[8] Zhixuan Chu, Jianmin Huang, Ruopeng Li, Wei Chu, and Sheng Li. Causal effect estimation: Recent advances, challenges, and opportunities, 2023.

[9] Marco Cuturi and Arnaud Doucet. Fast computation of wasserstein barycenters. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 685–693, Bejing, China, 22–24 Jun 2014. PMLR.

[10] Issa J Dahabreh, Rodney Hayward, and David M Kent. Using group data to treat individuals: understanding heterogeneous treatment effects in the age of precision medicine and patient-centred evidence. *International journal of epidemiology*, 45(6):2184–2193, 2016.

[11] Ana V Diez Roux. Estimating neighborhood health effects: the challenges of causal inference in a complex world. *Social Science Medicine*, 58(10):1953–1960, 2004.

[12] Xin Du, Lei Sun, Wouter Duivesteijn, Alexander Nikolaev, and Mykola Pechenizkiy. Adversarial balancing-based representation learning for causal effect inference with observational data. *Data Mining and Knowledge Discovery*, 35(4):1713–1738, may 2021.

[13] Paul HC Eilers and Brian D Marx. Flexible smoothing with b-splines and penalties. *Statistical science*, 11(2):89–121, 1996.

[14] Jianqing Fan, Kosuke Imai, Han Liu, Yang Ning, Xiaolin Yang, et al. Improving covariate balancing propensity score: A doubly robust and efficient approach. *URL: https://imai. fas. harvard. edu/research/CBPStheory. html*, 2016.

[15] Farzan Farnia and David Tse. A minimax approach to supervised learning, 2016.

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

[17] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks, 2016.

[18] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[19] Yury Gorishniy, Ivan Rubachev, and Artem Babenko. On embeddings for numerical features in tabular deep learning, 2023.

[20] Arthur Gretton, Karsten Borgwardt, Malte J. Rasch, Bernhard Scholkopf, and Alexander J. Smola. A kernel method for the two-sample problem, 2008.

[21] Arthur Gretton, Kenji Fukumizu, Choon Hui Teo, Le Song, Bernhard Schölkopf, and Alex Smola. A kernel statistical test of independence. In *NIPS*, 2007.

[22] Negar Hassanpour and Russell Greiner. Learning disentangled representations for counterfactual regression. In *International Conference on Learning Representations*, 2019.

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

[24] Keisuke Hirano, Guido W Imbens, and Geert Ridder. Efficient estimation of average treatment effects using the estimated propensity score. *Econometrica*, 71(4):1161–1189, 2003.

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

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

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

[28] Fredrik Johansson, Uri Shalit, and David Sontag. Learning representations for counterfactual inference. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 3020–3029, New York, New York, USA, 20–22 Jun 2016. PMLR.

[29] David M Kent, Ewout Steyerberg, and David van Klaveren. Personalized evidence based medicine: predictive approaches to heterogeneous treatment effects. *Bmj*, 363, 2018.

[30] Roni W. Kobrosly. causal-curve: A python causal inference package to estimate causal dose-response curves. *Journal of Open Source Software*, 5(52):2523, 2020.

[31] Dawen Liang, Laurent Charlin, and David M Blei. Causal inference for recommendation. In *Causation: Foundation to Application, Workshop at UAI. AUAI*, 2016.

[32] Tengyuan Liang. Estimating certain integral probability metric (ipm) is as hard as estimating under the ipm, 2019.

[33] Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation: Learning bounds and algorithms, 2009.

[34] Cancer Genome Atlas Research Network, John N Weinstein, Eric A Collisson, Gordon B Mills, Kenna R Mills Shaw, Brad A Ozenberger, Kyle Ellrott, Ilya Shmulevich, Chris Sander, Joshua M Stuart, and Fabio Vandin. The cancer genome atlas pan-cancer analysis project. *Nature Genetics*, 45(10):1113–1120, October 2013.

[35] Lizhen Nie, Mao Ye, Qiang Liu, and Dan Nicolae. Vcnet and functional targeted regularization for learning causal effects of continuous treatments. *arXiv preprint arXiv:2103.07861*, 2021.

[36] Judea Pearl. Causal inference in statistics: An overview. 2009.

[37] Mattia Prosperi, Yi Guo, Matt Sperrin, James S Koopman, Jae S Min, Xing He, Shannan Rich, Mo Wang, Iain E Buchan, and Jiang Bian. Causal inference and counterfactual prediction in machine learning for actionable healthcare. *Nature Machine Intelligence*, 2(7):369–375, 2020.

[38] James M. Robins, Andrea Rotnitzky, and Lue Ping Zhao. Estimation of regression coefficients when some regressors are not always observed. *Journal of the American Statistical Association*, 89(427):846–866, 1994.

[39] Donald B Rubin. Causal inference using potential outcomes. *Journal of the American Statistical Association*, 100(469):322–331, 2005.

[40] Arthur Schafer. The ethics of the randomized clinical trial. *The New England journal of medicine*, 307:719–24, 10 1982.

[41] Patrick Schwab, Lorenz Linhardt, Stefan Bauer, Joachim M. Buhmann, and Walter Karlen. Learning counterfactual representations for estimating individual dose-response curves. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(04):5612–5619, apr 2020.

[42] Uri Shalit, Fredrik D Johansson, and David Sontag. Estimating individual treatment effect: generalization bounds and algorithms. In *International conference on machine learning*, pages 3076–3085. PMLR, 2017.

[43] Claudia Shi, David Blei, and Victor Veitch. Adapting neural networks for the estimation of treatment effects. *Advances in neural information processing systems*, 32, 2019.

[44] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation, 2017.

[45] Hal R Varian. Causal inference in economics and marketing. *Proceedings of the National Academy of Sciences*, 113(27):7310–7315, 2016.

[46] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.

[47] Cédric Villani. *Optimal transport – Old and new*, volume 338, pages xxii+973. 01 2008.

[48] Stefan Wager and Susan Athey. Estimation and inference of heterogeneous treatment effects using random forests. *Journal of the American Statistical Association*, 113(523):1228–1242, 2018.

[49] Hao Wang, Hao He, and Dina Katabi. Continuously indexed domain adaptation. *arXiv preprint arXiv:2007.01807*, 2020.

[50] Xin Wang, Shengfei Lyu, Xingyu Wu, Tianhao Wu, and Huanhuan Chen. Generalization bounds for estimating causal effects of continuous treatments. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

[51] Xin Wang, Shengfei Lyu, Xingyu Wu, Tianhao Wu, and Huanhuan Chen. Generalization bounds for estimating causal effects of continuous treatments. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 8605–8617. Curran Associates, Inc., 2022.

[52] Anpeng Wu, Junkun Yuan, Kun Kuang, Bo Li, Runze Wu, Qiang Zhu, Yueting Zhuang, and Fei Wu. Learning decomposed representations for treatment effect estimation. *IEEE Transactions on Knowledge and Data Engineering*, 35(5):4989–5001, 2022.

[53] Xiao Wu, Fabrizia Mealli, Marianthi-Anna Kioumourtzoglou, Francesca Dominici, and Danielle Braun. Matching on generalized propensity scores with continuous exposures, 2021.

[54] Liuyi Yao, Sheng Li, Yaliang Li, Mengdi Huai, Jing Gao, and Aidong Zhang. Representation learning for treatment effect estimation from observational data. In *Neural Information Processing Systems*, 2018.

[55] Liuyi Yao, Sheng Li, Yaliang Li, Mengdi Huai, Jing Gao, and Aidong Zhang. Representation learning for treatment effect estimation from observational data. *Advances in neural information processing systems*, 31, 2018.

[56] Yao Zhang, Alexis Bellot, and Mihaela van der Schaar. Learning overlapping representations for the estimation of individualized treatment effects, 2020.

[57] Yi-Fan Zhang, Hanlin Zhang, Zachary C. Lipton, Li Erran Li, and Eric P. Xing. Exploring transformer backbones for heterogeneous treatment effect estimation, 2022.