TREATMENT EFFECT ESTIMATION USING INVARIANT RISK MINIMIZATION

Abhin Shah,[†,⋆] *Kartik Ahuja,*[†] *Karthikeyan Shanmugam,*[†] *Dennis Wei,*[†]
Kush R. Varshney,[†] *and Amit Dhurandhar*[†]

[†]IBM Research, [⋆]Massachusetts Institute of Technology

ABSTRACT

Inferring causal individual treatment effect (ITE) from observational data is a challenging problem whose difficulty is exacerbated by the presence of treatment assignment bias. In this work, we propose a new way to estimate the ITE using the domain generalization framework of invariant risk minimization (IRM). IRM uses data from multiple domains, learns predictors that do not exploit spurious domain-dependent factors, and generalizes better to unseen domains. We propose an IRM-based ITE estimator aimed at tackling treatment assignment bias when there is little support overlap between the control group and the treatment group. We accomplish this by creating *diversity*: given a single dataset, we split the data into multiple domains artificially. These diverse domains are then exploited by IRM to more effectively generalize regression-based models to data regions that lack support overlap. We show gains over classical regression approaches to ITE estimation in settings when support mismatch is more pronounced.

Index Terms— Causal inference, individual treatment effect estimation, invariant risk minimization

1. INTRODUCTION

Estimating the individual-level causal effect of a treatment is a fundamental problem in causal inference and applies to many fields. A few examples include understanding how a certain medication affects a patient's health [1, 2], understanding how Yelp ratings influence a potential restaurant customer [3], estimating the influence of individuals in social networks [4], inferring the effect of a policy in recommendation systems [5], assessing the causal impact of the treatment reception in sensor networks [6], estimating the impact of demand response signals [7], and evaluating the effect of a policy on unemployment rates [8]. Traditionally, randomized control trials (RCTs) have been used to evaluate treatment effects, but they can often be expensive and in some cases unethical.

In most scenarios, observational data that contains past actions and their responses is readily available. However, observational data does not provide access to the causal reasoning behind a particular action. See Table 1 for an illustrative example of a hospital record where age and blood pressure are features of patients, and blood sugar (either 'low' or 'high') is the response to a drug (either '0' or '1'). For a binary treatment, one of the options is often referred to as *the control* (say drug '0') and the other one as *the treatment* (say drug '1'). The group of individuals receiving *the control* is collectively referred to as *the control group* (patients 'A', 'B' in Table 1), and the group of individuals receiving *the treatment* is collectively referred to as *the treatment group* (patients 'C', 'D', 'E' in Table 1).

The individual treatment effect (ITE) of a binary treatment is the difference between the outcome under *the treatment* and the outcome under *the control*. Estimating ITE from observational data differs from classical supervised learning because we never observe the ITE in our training data. For example, in Table 1 we do not observe the blood sugar under *the treatment* for patients in the *control group* and the blood sugar under *the control* for patients in the *treatment group*.

Table 1: A typical observational record from a hospital

Patient	Age	Blood Pressure	Drug	Blood sugar
A	22	145/95	0	Low
B	26	135/80	0	Low
C	58	130/70	1	Low
D	50	145/80	1	High
E	24	150/85	1	Low

Unlike RCTs, observational data is often prone to treatment assignment bias [9]. For instance, patients receiving drug '0' may have a higher natural tendency (due to their age) to have low blood sugar than patients receiving drug '1'. In other words, sub-populations receiving different treatments can have very different distributions, and a traditional supervised learning model trained to predict the effect of treatment would fail to generalize well to the entire population. This issue calls for domain generalization methods for ITE estimation; in this paper, we make progress in this direction.

1.1. Related Works

Covariate adjustment. Existing works on covariate adjustment, a popular approach in treatment effect estimation, can be divided into two broad categories (a) balancing/matching, and (b) regression adjustment. Classical techniques for balancing rely on propensity score estimation [10]. Propensity score weighting [11] re-weights the samples to make *the treatment group* and *the control group* more similar. A few approaches [12, 13, 14] directly minimize imbalance metrics like kernel maximum mean discrepancy or discriminative discrepancy. Classical matching techniques match the samples from *the treatment group* and *the control group* using nearest neighbor matching [15, 16] or optimal matching [17]. More recent methods match using the estimated propensity score [11], coarsened versions of the observed covariates [18], or cardinality matching [19].

Regression adjustment estimates the potential outcomes with a supervised learning model fit on the features and the treatment. There are two main categories: (a) the T-learner (T for 'two') that uses separate base-learners to estimate the outcome under *control* and under *treatment* and (b) the S-learner (S for 'single') that uses one base-learner to estimate the outcome using the features and the treatment assignment, without giving the treatment assignment any special role. This terminology comes from [20], and we will use it throughout this paper. Ordinary least squares (OLS) regression, which solves the empirical risk minimization (ERM) problem for square loss and linear function class, is one traditional choice as the base-learner for T-learner and S-learner. We denote these by

OLS/LR2 and OLS/LR1, respectively. Advanced machine learning (ML) methods such as tree-based models [21, 22, 23] and deep generative models like generative adversarial networks, variational autoencoders, and multi-task Gaussian processes [2, 24, 25] have also been employed.

Domain Adaptation and Generalization. In domain adaptation for supervised learning, the learner exploits the access to labeled data from the training domain and unlabeled data from test domain and performs well on the test domain. In domain adaptation-driven ITE estimation methods [1], the labeled training data consists of outcomes under *the treatment* of *the treatment group* and unlabeled test data is *the control group* for which *the treatment* outcomes are unknown. Recent works inspired by domain adaptation [1, 26, 27] focus on learning new feature representations using neural architectures to match *the treatment group* and *the control group* in the representation space. This is effective when the learned feature representation is strongly ignorable (no unmeasured confounding [28]). However, the usual strong ignorability assumption might not hold for this learned representation even if it holds for the original features. Domain generalization methods [29, 30, 31] in supervised learning use labeled data from multiple training domains while not requiring any unlabeled test data and learn models that generalize well to unseen domains. Domain generalization based methods seem to offer several advantages over domain adaptation in supervised learning but have not been explored for ITE estimation, which is the objective of this work. In our work, we rely on a recent domain generalization framework called invariant risk minimization (IRM) [31]. The IRM uses the following principle to perform well on unseen domains: rely on features whose predictive power is invariant across domains, and ignore features whose predictive power varies across domains. Note that our usage of IRM for ITE does not rely on any additional ignorability assumptions on intermediate representations learned.

1.2. Contributions

In this work, we explore the idea of domain generalization for ITE estimation from observational data. More specifically, we propose a new way to estimate ITE by bridging the framework of IRM and causal effect estimation. Our estimator is most effective when there is limited overlap in the support between *the control group* and *the treatment group*. Although the data comes from a single domain, we artificially create the diverse domains required for IRM. We provide an intuitive explanation of how IRM uses these diverse domains to tackle treatment assignment bias when there is little support overlap. We support this with experiments and show gains over OLS/LR1 (linear S-learner) and OLS/LR2 (linear T-learner) in various settings when support mismatch is more pronounced.

Comparisons. For a first evaluation of IRM in ITE estimation, we consider experiments in a simpler linear setting with the necessary interaction term for heterogeneity. Our primary approach uses the IRM framework as the base-learner for the T-learner (denoted by IRM$_2$) and is most comparable to OLS/LR2. The base-learner of OLS/LR2 that estimates the outcome under *the control* does not use any information about the feature distribution of *the treatment group*, similar to IRM$_2$. This is in contrast to OLS/LR1 that uses the feature distribution of both *the control group* and *the treatment group* to estimate the outcome under *the control*. For the sake of completeness, we also use the IRM framework as the base-learner for the S-learner (denoted by IRM$_1$) and compare with both OLS/LR2 and OLS/LR1. We defer the comparison of our approach with ITE estimation approaches that use non-linear ML methods for future work.

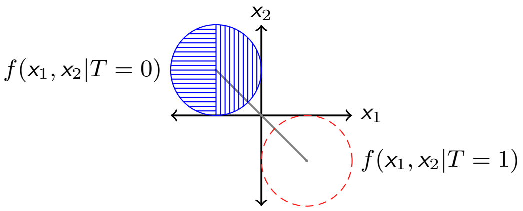

Fig. 1: A toy example of an observational data to illustrate the intuition behind applying the IRM framework to ITE estimation

2. A TOY EXAMPLE

Consider the illustrative example in Figure 1 with a binary treatment T. The feature distribution for *the control group* ($f(x_1, x_2|T = 0)$) is in blue and the feature distribution for *the treatment group* ($f(x_1, x_2|T = 1)$) is in red. As shown, this is a case of a support mismatch between the two groups. We use a full circle i.e., ○ and a dashed circle i.e., ◌ to denote the treatment assignment $T = 0$ and $T = 1$ respectively. Given observational data, we have access to the outcome under $T = 0$ for the *control group* i.e., ○ and the outcome under $T = 1$ for the *treatment group* i.e., ◌. We aim to estimate the outcome under $T = 1$ for the *control group* i.e., ◌ and the outcome under $T = 0$ for the *treatment group* i.e., ○.

Let us first focus on the T-learner. Our first base-learner (say *the control* branch) is supposed to learn the outcome for $T = 0$ using only ○ (i.e., training data) and estimate the outcome for ○ (i.e., test data). Similarly, our second base-learner (say *the treatment* branch) is supposed to learn the outcome for $T = 1$ using only ◌ (i.e., training data) to estimate the outcome for ◌ (i.e., test data). Each of these base-learners is required to do domain generalization to

Let us look at *the control* branch in detail. If we were to use OLS as the base-learner, then with access to finite data, it will pick up the spurious correlations (induced by treatment assignment bias) in ○ and fail to generalize well. In other words, the OLS/LR2 trained on ○ will perform well on individuals from ○ but will fail to do well on individuals from ○. If we were to use the IRM as the base-learner, we first need to split ○ into multiple domains (say ◑ and ◐) so as to have varying levels of spurious correlations in ◑ and ◐. By training on ◑ and ◐, *the control* branch of IRM$_2$ learns how to transport between ◑ and ◐. Being a domain generalization method, we expect the IRM method to generalize well on ○ which is outside the convex hull of the training data (i.e., outside ○). As the support overlap between *the control group* and *the treatment group* increases, we will see in Section 5 that gains of IRM$_2$ over OLS/LR2 and OLS/LR1 become more prominent.

Let us now focus on the S-learner that uses a single base-learner to learn the outcomes for both $T = 0$ and $T = 1$ using ○ and ◌ (i.e., training data) to estimate the outcome for ◌ and ○ (i.e., test data). As before, OLS will pick spurious correlations and fail to generalize well, but we still expect the IRM to generalize well. However, this domain generalization is not as straightforward as the T-learner because the treatment assignment is treated in a similar fashion as the other features of an individual, and there is lesser information for IRM to exploit the invariant factors across the domains.

3. PROBLEM FORMULATION

3.1. Setup

We adopt the Rubin-Neyman potential outcomes framework [32].

Let $\mathcal{X}$ be the d-dimensional feature space, $\mathcal{Y}$ be the outcome space, and $\boldsymbol{\mathsf{x}} \in \mathcal{X}$ be the d-dimensional feature vector. Let $\mathsf{t} = \{0, 1\}$ be the binary treatment variable with $\mathsf{t} = 1$ being *the treatment* and $\mathsf{t} = 0$ being *the control*. For $i \in \{0, 1\}$, let $\mathsf{y}_i \in \mathcal{Y}$ be the potential outcome under $\mathsf{t} = i$. Let $\mathbf{x} \in \mathcal{X}, t \in \{0, 1\}, y_0 \in \mathcal{Y}, y_1 \in \mathcal{Y}$ denote realizations of $\boldsymbol{\mathsf{x}}, \mathsf{t}, \mathsf{y}_0, \mathsf{y}_1$ respectively. Suppose we have an observational dataset of n individuals where for each individual we only observe the potential outcome that corresponds to the assigned treatment (denoted by y_f and referred to as the *factual outcomes*). Let our dataset be $\mathcal{D}^{(n)} = \{\mathbf{x}^{(i)}, t^{(i)}, y_f^{(i)}\}_{i=1}^n$ where $y_f^{(i)} = y_0^{(i)}$ if $t^{(i)} = 0$ and $y_f^{(i)} = y_1^{(i)}$ if $t^{(i)} = 1$. Let $[n] = \{1, \cdots, n\}$.

We assume the standard strong ignorability condition: $0 < p(\mathsf{t} = 1|\boldsymbol{\mathsf{x}}) < 1$ and $(\mathsf{y}_0, \mathsf{y}_1) \perp\!\!\!\perp \mathsf{t}|\boldsymbol{\mathsf{x}}$ for all $\boldsymbol{\mathsf{x}}$. This is a sufficient condition for ITE to be identifiable from observational data [28, 33].

3.2. Inference Tasks

Our interest lies in learning the ITE (denoted by τ) defined as:

$$\tau^{(i)} = y_1^{(i)} - y_0^{(i)} \quad \forall i \in [n] \tag{1}$$

Empirically, the estimate $\hat{\tau}$ is evaluated using the *precision in estimation of heterogeneous effect* (PEHE) which is the mean squared error of the estimated ITE for all the individuals in our data:

$$\epsilon_{PEHE} = \frac{1}{n} \sum_{i=1}^{n} (\tau^{(i)} - \hat{\tau}^{(i)})^2 \tag{2}$$

3.3. Invariant Risk Minimization

[31] consider datasets D_e, consisting of observations of the feature vector ($\boldsymbol{\mathsf{x}} \in \mathcal{X}$) and the response ($\mathsf{y} \in \mathcal{Y}$), collected under multiple training domains $e \in \mathcal{E}_{tr}$. The dataset D_e, from domain e, contains i.i.d. samples according to some probability distribution $\mathcal{P}_e$. The goal is to use these multiple datasets to learn a predictor $f : \mathcal{X} \to \mathcal{Y}$ that minimizes the maximum risk over all the domains $\mathcal{E}$ i.e., $\min_f \max_{e \in \mathcal{E}} R_e(f)$ where $R_e(f) = \mathbb{E}_{(\boldsymbol{\mathsf{x}}, \mathsf{y}) \sim \mathcal{P}_e}[l(f(\boldsymbol{\mathsf{x}}), \mathsf{y})]$ is the risk under domain e for a convex and differentiable loss function l. The practical version of IRM (i.e., IRMv1) is as follows:

$$\min_{\Phi : \mathcal{X} \to \mathcal{Y}} \sum_{e \in \mathcal{E}_{tr}} R_e(\Phi) + \lambda \|\nabla_{w|w=1.0} R_e(w \cdot \Phi)\|^2 \tag{3}$$

where $\Phi : \mathcal{X} \to \mathcal{Y}$ is an invariant predictor, $w = 1.0$ is a fixed "dummy" classifier, the gradient norm penalty measures the optimality of the dummy classifier at each domain e. The first term in (3) is a standard ERM term and $\lambda \in [0, \infty)$ is a regularizer balancing between this, and the invariance of the predictor $1 \cdot \Phi(\mathbf{x})$. [31] solves IRMv1 in (3) using stochastic gradient descent (SGD).

4. OUR APPROACH

We do not assume access to multiple domains as required by IRM. Given access to a dataset from a single domain, we first split the dataset into different components representing diverse domains. The next step is the application of IRM.

4.1. Domain Generation

We split $\mathcal{D}^{(n)}$ into n_e components as if each component is obtained from a different domain. To achieve this, we assign a variable $\mathsf{e} \in [n_e]$ to each individual denoting which domain we place it in. We have $\mathcal{D}^{(n)} = \cup_{j=1}^{n_e} \mathcal{D}_j^{(n)}$ where $\mathcal{D}_j^{(n)} = \{(\mathbf{x}^{(i)}, t^{(i)}, y_f^{(i)}) : i \in [n], e^{(i)} = j\}$. We explored a variety of domain generation schemes. It turns out that, for our relatively simple setup, (uniformly) random domain generation is sufficient[1] i.e., e takes any value in $[n_e]$ with the same probability.

4.2. Procedure

Let there be n_{tr} training samples. Let $\mathcal{D}_j^{(n_{tr})}$ be the component of $\mathcal{D}^{(n_{tr})}$ corresponding to the j^{th} domain. Let $\mathcal{D}^{(n_{te})} = \{\mathbf{x}^{(i)}\}_{i=1}^{n_{te}}$ denote the test dataset consisting of n_{te} samples.

1. **T-learner / IRM$_2$**: For $j \in [n_e]$, let $\mathcal{D}_{j,c}^{(n_{tr})} = \{(\mathbf{x}^{(i)}, y_f^{(i)}) : i \in [n_{tr}], t^{(i)} = 0, e^{(i)} = j\}$ be the *control* component of $\mathcal{D}_j^{(n_{tr})}$. Similarly, let $\mathcal{D}_{j,t}^{(n_{tr})} = \{(\mathbf{x}^{(i)}, y_f^{(i)}) : i \in [n_{tr}], t^{(i)} = 1, e^{(i)} = j\}$ be the *treatment* component of $\mathcal{D}_j^{(n_{tr})}$.
    - **Training**. $\mathcal{D}_{j,c}^{(n_{tr})}$ $\forall j \in [n_e]$ is training data for *the control* branch of IRM$_2$. $\mathcal{D}_{j,t}^{(n_{tr})}$ $\forall j \in [n_e]$ is training data for *the treatment* branch of IRM$_2$. Following [31], we use SGD to optimize IRMv1 in (3) for both branches of IRM$_2$.
    - **Testing**. Predict *the control* outcomes on $\mathcal{D}^{(n_{te})}$ using *the control* branch of IRM$_2$ and *the treatment* outcomes on $\mathcal{D}^{(n_{te})}$ using *the treatment* branch of IRM$_2$.
2. **S-learner / IRM$_1$**: For $j \in [n_e]$, let $\hat{\mathcal{D}}_j^{(n_{tr})} = \{(\mathbf{x}^{(i)}, t^{(i)}, \mathbf{x}^{(i)} \times t^{(i)}, y_f^{(i)}) : i \in [n], e^{(i)} = j\}$ where $\mathbf{x} \times t$ is the interaction term.
    - **Training**. $\hat{\mathcal{D}}_j^{(n_{tr})}$ $\forall j \in [n_e]$ is the training data for the IRM$_1$. Following [31], we use SGD to optimize IRMv1 in (3) for IRM$_1$.
    - **Testing**. Using the trained IRM$_1$ framework, predict *the control* outcome on $\mathcal{D}_c^{(n_{te})} = \{(\mathbf{x}^{(i)}, 0, 0)\}_{i=1}^{n_{te}}$ and *the treatment* outcome on $\mathcal{D}_t^{(n_{te})} = \{(\mathbf{x}^{(i)}, 1, \mathbf{x}^{(i)})\}_{i=1}^{n_{te}}$.

OLS/LR1 and OLS/LR2 can be understood as unpenalized cases ($\lambda = 0$) of (3) and with $n_e = 1$ in the above procedure.

5. EXPERIMENTS

5.1. Data Generation

In our data generation mechanism, we first generate the treatment, followed by the features conditional on the treatment, and finally the outcomes conditional on the treatment and the features.

1. **Treatment generation:** Treatment assignments are drawn from a Bernoulli distribution with mean 0.5 i.e., $\mathsf{t} \sim \text{Bernoulli}(0.5)$.
2. **Feature generation:** Given the treatment assignment, we consider two feature generation models.

    • **Model A:** In the first, the features for different groups (*the control* and *the treatment*) are drawn from multivariate Gaussian distribution with different means. For $t \in \{0, 1\}$,

    $$\boldsymbol{\mathsf{x}}|\mathsf{t} = t \sim \mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}) \tag{4}$$

    • **Model B:** In the second, the features for different groups (*the control* and *the treatment*) are drawn from different multivariate Gaussian mixture distributions as follows. For $t \in \{0, 1\}$,

    $$\boldsymbol{\mathsf{x}}|\mathsf{t} = t \sim 0.5 \times \mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_0) + 0.5 \times \mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_1) \tag{5}$$

[1] For our setup, with relatively little data (200 training samples) and in high dimensions, the random scheme is sufficient as the domains appear sufficiently different to the different learners. We do not claim that the random scheme would work all the time.

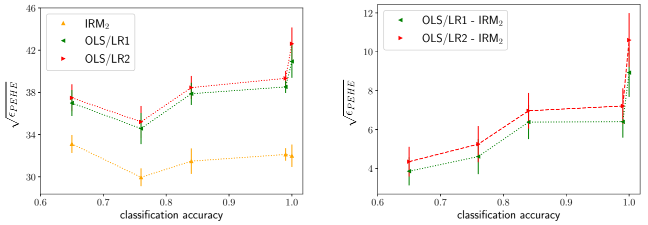

Fig. 2: $\sqrt{\epsilon_{PEHE}}$ (left) and $\sqrt{\epsilon_{PEHE}}$ difference (right) vs treatment group classification accuracy for model A with quadratic outcomes

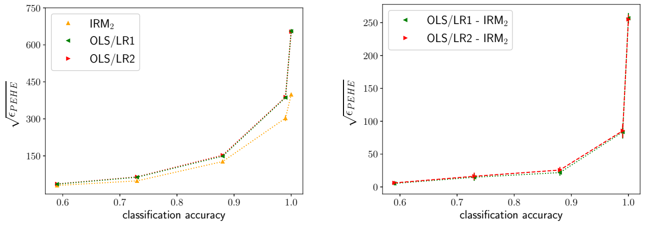

Fig. 3: $\sqrt{\epsilon_{PEHE}}$ (left) and $\sqrt{\epsilon_{PEHE}}$ difference (right) vs treatment group classification accuracy for model B with quadratic outcomes

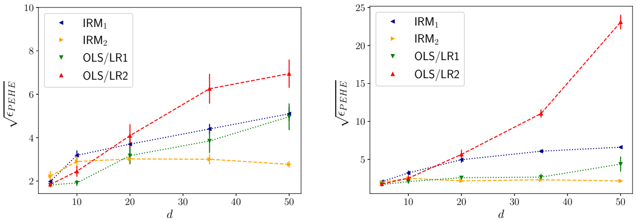

Fig. 4: $\sqrt{\epsilon_{PEHE}}$ vs d for models A and B with linear outcome

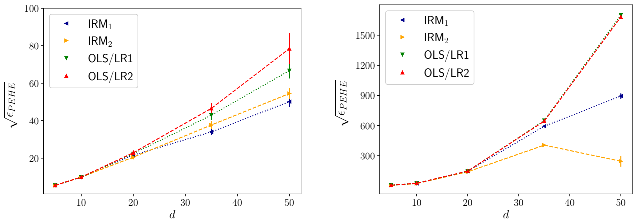

Fig. 5: $\sqrt{\epsilon_{PEHE}}$ vs d for models A and B with quadratic outcome

3. **Outcome generation**: Given the features and the treatment assignment, we consider two outcome generation methods.
   • **Linear:** In the first, outcomes for different groups are drawn from Gaussian distributions with means given by group-dependent linear functions of the features. For $t \in \{0, 1\}$,
   $$y_t|\mathbf{x} = \mathbf{x}, t = t \sim \mathcal{N}(\mathbf{x}^T b_t + c_t, \sigma^2) \tag{6}$$
   • **Quadratic:** In the second, outcomes for different groups are drawn from Gaussian distributions with means given by group-dependent quadratic functions of the features. For $t \in \{0, 1\}$,
   $$y_t|\mathbf{x} = \mathbf{x}, t = t \sim \mathcal{N}(\mathbf{x}^T A_t \mathbf{x} + \mathbf{x}^T b_t + c_t, \sigma^2) \tag{7}$$
   Given the treatment assignment and the potential outcomes, the factual outcomes are: $y_f = t \times y_1 + (1 - t) \times y_0$. We know the true potential outcomes and therefore the ITE using (1).

5.2. ITE estimation

We consider 4 data generation schemes: (a) model A with linear outcome, (b) model B with linear outcome, (c) model A with quadratic outcome, (d) model B with quadratic outcome. We consider $n_{tr} = 200$ train samples, $n_{te} = 100$ test samples, and $n_e = 3$ domains. We average our results over 10 repetitions.

To generate the covariance matrices in (4), (5), we first draw eigenvalues uniformly from $[0, 1]$, place them as the diagonal entries of diagonal matrices $\boldsymbol{\lambda}$, $\boldsymbol{\lambda}_0$, and $\boldsymbol{\lambda}_1$, and re-scale them to sum to 1. The entries of $\boldsymbol{\lambda}$ and $\boldsymbol{\lambda}_0$ are placed in increasing order and the entries of $\boldsymbol{\lambda}_1$ are in decreasing order. We then let $\boldsymbol{\Sigma}$ = $\mathbf{Q}_A \boldsymbol{\lambda} \mathbf{Q}_A$ in (4), $\boldsymbol{\Sigma}_0$ = $\mathbf{Q}_B \boldsymbol{\lambda}_0 \mathbf{Q}_B$ in (5) and $\boldsymbol{\Sigma}_1$ = $\mathbf{Q}_B \boldsymbol{\lambda}_1 \mathbf{Q}_B$ in (5) for two orthonormal eigenvector matrices $\mathbf{Q}_A$ and $\mathbf{Q}_B$ with entries drawn from $\mathcal{N}(0, 1)$. We choose the coefficients c_0, c_1 in (6), (7), the entries of the vectors b_0, b_1 in (6), (7), and the entries of the matrices A_0, A_1 in (7) from the uniform distribution over $[0, 1]$. We let σ in (6), (7) be 1.

For the first set of experiments, we quantify the mismatch between *the control group* and *the treatment group* using the *classification accuracy* in distinguishing between the groups, i.e., predicting treatment assignment with features $\mathbf{x}$ as input to the classifier, $p(t = 1|\mathbf{x})$. With $d = 35$, we vary the $\boldsymbol{\mu}_0$ and $\boldsymbol{\mu}_1$ in (4) and (5) (i.e., the length of ⟍ in Fig. 1) to vary the separation between *the control group* and *the treatment group* and in-turn vary the classification accuracy. Fig. 2 shows that as the distributions of *the control group* and *the treatment group*, for model A with quadratic outcomes, become more mismatched i.e., as the classification accuracy increases, the gains of IRM$_2$ over OLS/LR1, and OLS/LR2 start increasing. Fig. 3 shows the same for model B with quadratic outcomes. We do not show similar plots for the linear outcome generation method because, in the relatively simpler linear setting, the gains of IRM$_2$ are visible only when classification accuracy is very close to one.

For the second set of experiments, for the linear outcome models, we let $\boldsymbol{\mu}_0$ in (4) to be all -1's and $\boldsymbol{\mu}_1$ in (5) to be all +1's. Similarly, for the quadratic outcome models, we let $\boldsymbol{\mu}_0$ in (4) to be all -0.1's and $\boldsymbol{\mu}_1$ in (5) to be all +0.1's. We vary the dimension d as 5,10,20,35,50 and plot the PEHE for IRM$_2$, IRM$_1$, OLS/LR2, and OLS/LR1 for the linear outcome models in Fig. 4 and for the quadratic outcome models in Fig. 5. For linear models, IRM$_2$ outperforms the other methods in high dimensions and we need a greater mismatch between *the control* and *the treatment* groups i.e., $\boldsymbol{\mu}$'s to be -1's and +1's to achieve this. For quadratic models, both IRM$_2$ and IRM$_1$ outperform OLS/LR2 and OLS/LR1 even in the regimes with lower mismatch between *the control group* and *the treatment group*, i.e., $\boldsymbol{\mu}$'s to be -0.1's and +0.1's.

The source code of our implementation is available at `https://github.com/IBM/OoD/tree/master/IRM_ITE`

6. CONCLUSION

We have developed an approach for making ITE estimation robust to treatment assignment bias using the domain generalization framework of IRM. We use IRM base-learners inside the S-learner and T-learner frameworks for ITE estimation. In contrast to the typical setting for IRM, we do not require datasets coming from different domains, but create diverse partitions as part of the inference method. We see from our experiments that in scenarios with treatment assignment bias, IRM captures fewer biases compared to OLS. As the treatment assignment bias increases, the reduction in the bias of IRM becomes more prominent.

7. REFERENCES

[1] Uri Shalit, Fredrik D Johansson, and David Sontag, "Estimating individual treatment effect: generalization bounds and algorithms," in *International Conference on Machine Learning*. PMLR, 2017, pp. 3076–3085.

[2] Ahmed M Alaa and Mihaela van der Schaar, "Bayesian inference of individualized treatment effects using multi-task gaussian processes," in *Advances in Neural Information Processing Systems*, 2017, pp. 3424–3432.

[3] Michael Anderson and Jeremy Magruder, "Learning from the crowd: Regression discontinuity estimates of the effects of an online review database," *The Economic Journal*, vol. 122, no. 563, pp. 957–989, 2012.

[4] S. T. Smith, E. K. Kao, D. C. Shah, O. Simek, and D. B. Rubin, "Influence estimation on social media networks using causal inference," in *2018 IEEE Statistical Signal Processing Workshop (SSP)*, 2018, pp. 328–332.

[5] Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims, "Recommendations as treatments: Debiasing learning and evaluation," in *international conference on machine learning*. PMLR, 2016, pp. 1670–1679.

[6] M. Coates and I. Psaromiligkos, "Evaluating average causal effect using wireless sensor networks," in *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2004, vol. 3, pp. iii–905.

[7] P. Li and B. Zhang, "An optimal treatment assignment strategy to evaluate demand response effect," in *2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 2016, pp. 703–710.

[8] Robert J LaLonde, "Evaluating the econometric evaluations of training programs with experimental data," *The American economic review*, pp. 604–620, 1986.

[9] Paul R Rosenbaum, "Overt bias in observational studies," in *Observational studies*, pp. 71–104. Springer, 2002.

[10] Paul R Rosenbaum and Donald B Rubin, "The central role of the propensity score in observational studies for causal effects," *Biometrika*, vol. 70, no. 1, pp. 41–55, 1983.

[11] Peter C Austin, "An introduction to propensity score methods for reducing the effects of confounding in observational studies," *Multivariate behavioral research*, vol. 46, no. 3, pp. 399–424, 2011.

[12] Nathan Kallus, "A framework for optimal matching for causal inference," in *Artificial Intelligence and Statistics*. PMLR, 2017, pp. 372–381.

[13] Arthur Gretton, Alex Smola, Jiayuan Huang, Marcel Schmittfull, Karsten Borgwardt, and Bernhard Schölkopf, "Covariate shift by kernel mean matching," .

[14] Nathan Kallus, "Deepmatch: Balancing deep covariate representations for causal inference using adversarial training," *arXiv preprint arXiv:1802.05664*, 2018.

[15] Donald B Rubin, "Matching to remove bias in observational studies," *Biometrics*, pp. 159–183, 1973.

[16] Alberto Abadie, David Drukker, Jane Leber Herr, and Guido W Imbens, "Implementing matching estimators for average treatment effects in stata," *The stata journal*, vol. 4, no. 3, pp. 290–311, 2004.

[17] Paul R Rosenbaum, "Optimal matching for observational studies," *Journal of the American Statistical Association*, vol. 84, no. 408, pp. 1024–1032, 1989.

[18] Stefano M Iacus, Gary King, and Giuseppe Porro, "Causal inference without balance checking: Coarsened exact matching," *Political analysis*, pp. 1–24, 2012.

[19] Giancarlo Visconti and José R Zubizarreta, "Handling limited overlap in observational studies with cardinality matching," 2018.

[20] Sören R Künzel, Jasjeet S Sekhon, Peter J Bickel, and Bin Yu, "Metalearners for estimating heterogeneous treatment effects using machine learning," *Proceedings of the national academy of sciences*, vol. 116, no. 10, pp. 4156–4165, 2019.

[21] Stefan Wager and Susan Athey, "Estimation and inference of heterogeneous treatment effects using random forests," *Journal of the American Statistical Association*, vol. 113, no. 523, pp. 1228–1242, 2018.

[22] Susan Athey and Guido Imbens, "Recursive partitioning for heterogeneous causal effects," *Proceedings of the National Academy of Sciences*, vol. 113, no. 27, pp. 7353–7360, 2016.

[23] Jennifer L Hill, "Bayesian nonparametric modeling for causal inference," *Journal of Computational and Graphical Statistics*, vol. 20, no. 1, pp. 217–240, 2011.

[24] Jinsung Yoon, James Jordon, and Mihaela van der Schaar, "GANITE: Estimation of individualized treatment effects using generative adversarial nets," in *International Conference on Learning Representations*, 2018.

[25] Christos Louizos, Uri Shalit, Joris M Mooij, David Sontag, Richard Zemel, and Max Welling, "Causal effect inference with deep latent-variable models," in *Advances in Neural Information Processing Systems*, 2017, pp. 6446–6456.

[26] Fredrik Johansson, Uri Shalit, and David Sontag, "Learning representations for counterfactual inference," in *International conference on machine learning*, 2016, pp. 3020–3029.

[27] Claudia Shi, David Blei, and Victor Veitch, "Adapting neural networks for the estimation of treatment effects," in *Advances in Neural Information Processing Systems*, 2019, pp. 2507–2517.

[28] Guido W Imbens and Jeffrey M Wooldridge, "Recent developments in the econometrics of program evaluation," *Journal of economic literature*, vol. 47, no. 1, pp. 5–86, 2009.

[29] Toshihiko Matsuura and Tatsuya Harada, "Domain generalization using a mixture of multiple latent domains.," in *AAAI*, 2020, pp. 11749–11756.

[30] Ishaan Gulrajani and David Lopez-Paz, "In search of lost domain generalization," *arXiv preprint arXiv:2007.01434*, 2020.

[31] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz, "Invariant risk minimization," *arXiv preprint arXiv:1907.02893*, 2019.

[32] Donald B Rubin, "Estimating causal effects of treatments in randomized and nonrandomized studies.," *Journal of educational Psychology*, vol. 66, no. 5, pp. 688, 1974.

[33] Judea Pearl, "Detecting latent heterogeneity," *Sociological Methods & Research*, vol. 46, no. 3, pp. 370–389, 2017.