Supplementary Material: Counterfactual Representation Learning with Balancing Weights

Serge Assaad[1] **Shuxi Zeng**[2] **Chenyang Tao**[1] **Shounak Datta**[1]
Nikhil Mehta[1] **Ricardo Henao**[1] **Fan Li**[2] **Lawrence Carin**[1]
[1]Department of ECE, Duke University [2]Department of Statistical Science, Duke University

Contents

1 Theory

1.1 Proof of Proposition 1

Proposition 1 (Balancing Property)**.** *Given the true propensity score $e(x)$, the reweighted treatment and control arms both equal the target distribution. In other words, $g(x|T=1) = g(x|T=0) = g(x)$*

Proof.

$$g(x|T=1) \overset{\triangle}{\propto} w(x,1)p(x|T=1) = \frac{f(x)}{e(x)}p(x|T=1) = \frac{f(x)\Pr(X=x|T=1)}{\Pr(T=1|X=x)} \propto f(x)p(x) \propto g(x) \tag{1}$$

Similarly, we can also show that $g(x|T=0) = g(x)$. □

1.2 Proof of Proposition 2

Assumption 1. *The odds ratio between the model propensity score and true propensity score is bounded, namely:*

$$\exists\ \Gamma \geq 1 \quad s.t.\ \forall x \in \mathcal{X}, \quad \frac{1}{\Gamma} \leq \frac{e(x)(1-e_\eta(x))}{e_\eta(x)(1-e(x))} \leq \Gamma \tag{2}$$

This assumption is conceptually related to the Marginal Sensitivity Model of Kallus et al. (2019) in that it measures the gap between two propensity functions – we use it here to quantify the gap between true and model propensities rather than the degree of unobserved confounding.

Proposition 2 (Generalized Balancing)**.** *Under Assumption 1, and assuming that all tilting functions f satisfy $f(x) > 0 \ \ \forall x \in \mathcal{X}$, we have:*

$$D_{KL}(g_\eta(x|T=1)||g_\eta(x|T=0)) \leq 2 \cdot \log \Gamma,$$

where D_{KL} is the KL-divergence.

Proof. First, we write the reweighted treatment group distribution as follows:

$$g_\eta(x|T=1) \overset{\triangle}{\propto} w_\eta(x,1)p(x|T=1) = \frac{f_\eta(x)}{e_\eta(x)}p(x|T=1), \tag{3}$$

where we write f_η since the tilting function is (in general) computed from the propensity score model. With $f(x)$ the "true" tilting function (*i.e.*, the tilting function computed from the true propensity $e(x)$), we may write:

$$g_\eta(x|T=1) \propto \frac{f(x)}{e(x)}p(x|T=1)\frac{f_\eta(x)}{f(x)}\frac{e(x)}{e_\eta(x)} \propto g(x)\frac{f_\eta(x)}{f(x)}\frac{e(x)}{e_\eta(x)} \tag{4}$$

where the last equality holds from Proposition 1. Similarly, we can write the reweighted control group distribution as

$$g_\eta(x|T=0) \propto g(x)\frac{f_\eta(x)}{f(x)}\frac{1-e(x)}{1-e_\eta(x)}.$$

Now, computing the KL-divergence between $g_\eta(x|T=1)$ and $g_\eta(x|T=0)$, we get:

$$D_{KL}(g_\eta(x|T=1)||g_\eta(x|T=0)) = \int_{\mathcal{X}} g_\eta(x|T=1)\log\left[\frac{\frac{1}{Z_1}g(x)\frac{f_\eta(x)}{f(x)}\frac{e(x)}{e_\eta(x)}}{\frac{1}{Z_0}g(x)\frac{f_\eta(x)}{f(x)}\frac{1-e(x)}{1-e_\eta(x)}}\right]dx \tag{5}$$

where $Z_1 \triangleq \int_{\mathcal{X}} g(x)\frac{f_\eta(x)}{f(x)}\frac{e(x)}{e_\eta(x)}dx$ and $Z_0 \triangleq \int_{\mathcal{X}} g(x)\frac{f_\eta(x)}{f(x)}\frac{1-e(x)}{1-e_\eta(x)}dx$. Simplifying (5) further, we get:

$$D_{KL}(g_\eta(x|T=1)||g_\eta(x|T=0)) = \int_{\mathcal{X}} g_\eta(x|T=1)\left[\log\frac{Z_0}{Z_1} + \log\frac{e(x)(1-e_\eta(x))}{e_\eta(x)(1-e(x))}\right]dx \tag{6}$$

$$\overset{(*)}{\leq} \int_{\mathcal{X}} g_\eta(x|T=1)\left[\log\frac{Z_0}{Z_1} + \log\Gamma\right]dx = \log\frac{Z_0}{Z_1} + \log\Gamma \tag{7}$$

where $(*)$ holds from Assumption 1. Notice that we may relate Z_1 and Z_0 as follows:

$$Z_0 \triangleq \int_{\mathcal{X}} g(x)\frac{f_\eta(x)}{f(x)}\frac{1-e(x)}{1-e_\eta(x)}dx \overset{(**)}{\leq} \int_{\mathcal{X}} g(x)\frac{f_\eta(x)}{f(x)}\Gamma\frac{e(x)}{e_\eta(x)}dx = \Gamma Z_1 \tag{8}$$

where $(**)$ also holds from Assumption 1. Hence $\log\frac{Z_0}{Z_1} \leq \log\Gamma$ – plugging this into (7) yields:

$$D_{KL}(g_\eta(x|T=1)||g_\eta(x|T=0)) \leq \log\Gamma + \log\Gamma = 2\log\Gamma. \tag{9}$$

□

Corollary 1. *The bound presented in Proposition 2 also holds for the induced distributions $g_{\Phi,\eta}(r|T=1), g_{\Phi,\eta}(r|T=0)$ from $g_\eta(x|T=1), g_\eta(x|T=0)$ (respectively) via any invertible map $\Phi : \mathcal{X} \to \mathcal{R}$ (with inverse Ψ), namely:*

$$D_{KL}(g_{\Phi,\eta}(r|T=1)||g_{\Phi,\eta}(r|T=0)) \leq 2\log\Gamma. \tag{10}$$

Proof. To see this, we can write:

$$g_{\Phi,\eta}(r|T=1) \overset{(*)}{\propto} g_\eta(\Psi(r)|T=1)|\det(\Psi')| \propto \frac{f(\Psi(r))}{e(\Psi(r))}p(\Psi(r)|T=1)\frac{f_\eta(\Psi(r))}{f(\Psi(r))}\frac{e(\Psi(r)}{e_\eta(\Psi(r))}|\det(\Psi')| \tag{11}$$

$$\propto g(\Psi(r))\frac{f_\eta(\Psi(r))}{f(\Psi(r))}\frac{e(\Psi(r))}{e_\eta(\Psi(r))}|\det(\Psi')| \tag{12}$$

where $\det(\Psi')$ is the determinant of the Jacobian of Ψ, and $(*)$ holds from the change-of-variables formula. Similarly, we can write $g_{\Phi,\eta}(r|T=0)$ as:

$$g_{\Phi,\eta}(r|T=0) \propto g(\Psi(r))\frac{f_\eta(\Psi(r))}{f(\Psi(r))}\frac{1-e(\Psi(r))}{1-e_\eta(\Psi(r))}|\det(\Psi')| \tag{13}$$

Computing the KL divergence between (12) and (13) is then similar to the proof of Proposition 2, and the same bound holds. □

1.3 Proof of Proposition 3

Definition 1. *The total variation distance (TVD) between distributions p and q on $\mathcal{R}$ is defined as*

$$\delta(p,q) \triangleq \frac{1}{2}\cdot\sup_{m:||m||_\infty\leq 1}\Big\{\int_{\mathcal{R}} m(r)(p(r)-q(r))dr\Big\} \tag{14}$$

Lemma 1. *Under Assumption 1, the total variation distance between the reweighted representation distribution for the treatment and control groups is upper bounded as:*

$$\delta(g_{\Phi,\eta}(r|T=1), g_{\Phi,\eta}(r|T=0)) \leq \sqrt{\log\Gamma} \tag{15}$$

Proof.

$$\delta(g_{\Phi,\eta}(r|T=1), g_{\Phi,\eta}(r|T=0)) \overset{(*)}{\leq} \sqrt{\frac{1}{2}D_{KL}(g_{\Phi,\eta}(r|T=1)||g_{\Phi,\eta}(r|T=0))} \overset{(**)}{\leq} \sqrt{\log\Gamma} \tag{16}$$

where $(*)$ follows from Pinsker's inequality, and $(**)$ follows from Corollary 1. □

Proposition 3. *Under Assumption 1, assuming the representation space $\mathcal{R}$ is bounded, and assuming the tilting functions satisfy $f(x) > 0\ \forall x \in \mathcal{X}$, the following bounds hold:*

$$\begin{aligned}&\mathcal{W}(g_{\Phi,\eta}(r|T=1), g_{\Phi,\eta}(r|T=0)) \leq \mathrm{diam}(\mathcal{R})\sqrt{\log\Gamma}\\ &\mathrm{MMD}_k(g_{\Phi,\eta}(r|T=1), g_{\Phi,\eta}(r|T=0)) \leq 2\sqrt{C_k\log\Gamma},\end{aligned} \tag{17}$$

where $\mathcal{W}$ is the Wasserstein distance, $\mathrm{diam}(\mathcal{R}) \triangleq \sup_{r,r'\in\mathcal{R}}||r-r'||_2$, MMD_k is the MMD with kernel k, and $C_k \triangleq \sup_{r\in\mathcal{R}}k(r,r)$.

Proof.

$$\mathcal{W}(g_{\Phi,\eta}(r|T=1), g_{\Phi,\eta}(r|T=0)) \overset{(*)}{\leq} \text{diam}(\mathcal{R})\delta(g_{\Phi,\eta}(r|T=1), g_{\Phi,\eta}(r|T=0)) \overset{(**)}{\leq} diam(\mathcal{R})\sqrt{\log\Gamma} \tag{18}$$

where $(*)$ holds from Theorem 4 of Gibbs & Su (2002), and $(**)$ holds from Lemma 1.

$$\text{MMD}_k(g_{\Phi,\eta}(r|T=1), g_{\Phi,\eta}(r|T=0)) \overset{(*)}{\leq} 2\sqrt{C_k}\delta(g_{\Phi,\eta}(r|T=1), g_{\Phi,\eta}(r|T=0)) \overset{(**)}{\leq} 2\sqrt{C_k\log\Gamma} \tag{19}$$

where $(*)$ holds from Theorem 14-ii of Sriperumbudur et al. (2009), and $(**)$ holds from Lemma 1. □

1.4 Relationship between $\epsilon_{\text{PEHE},p}$ and $\epsilon_{\text{PEHE},g}$

In this section, we establish a relationship between $\epsilon_{\text{PEHE},p}$ and $\epsilon_{\text{PEHE},g}$ which explains why targeting the population $g(x) \overset{\triangle}{\propto} f(x)p(x)$ for ITE prediction may also aid ITE prediction on the original covariate distribution $p(x)$. As a reminder, Table S1 and Figure S1 shows the different tilting functions of interest and their corresponding weighting schemes. $e(x) \triangleq \Pr(T=1|X=x)$ is the propensity score. The weight schemes we use here have been carefully examined in classical causal inference literature (Crump et al., 2009; Li et al., 2018; Li & Greene, 2013). Specifically, the Matching Weights (Li & Greene, 2013) were designed as a weighting analogue to matching, the Truncated IPW weights (Crump et al., 2009) were used to estimate a low-variance average treatment effect for a subpopulation, and the Overlap Weights (Li et al., 2018) were proven to minimize (out of all the possible balancing weights) the asymptotic variance of the estimated weighted average treatment effect. Figure S1 shows how TruncIPW, MW, and OW place a specific emphasis on regions of good overlap in covariate space.

Table S1: Choices of tilting function $f(x)$ and associated weight schemes $w(x,t)$ (see equation (6) in the main text). Note $\mathbb{1}(\cdot)$ is the indicator function. We set $\xi = 0.1$ as in Crump et al. (2009).

Tilting function $f(x)$	**Associated weight scheme** $w(x,t)$
1	Inverse Probability Weights (IPW)
$\mathbb{1}(\xi < e(x) < 1-\xi)$	Truncated IPW (TruncIPW)
$\min(e(x), 1-e(x))$	Matching Weights (MW)
$e(x)(1-e(x))$	Overlap Weights (OW)

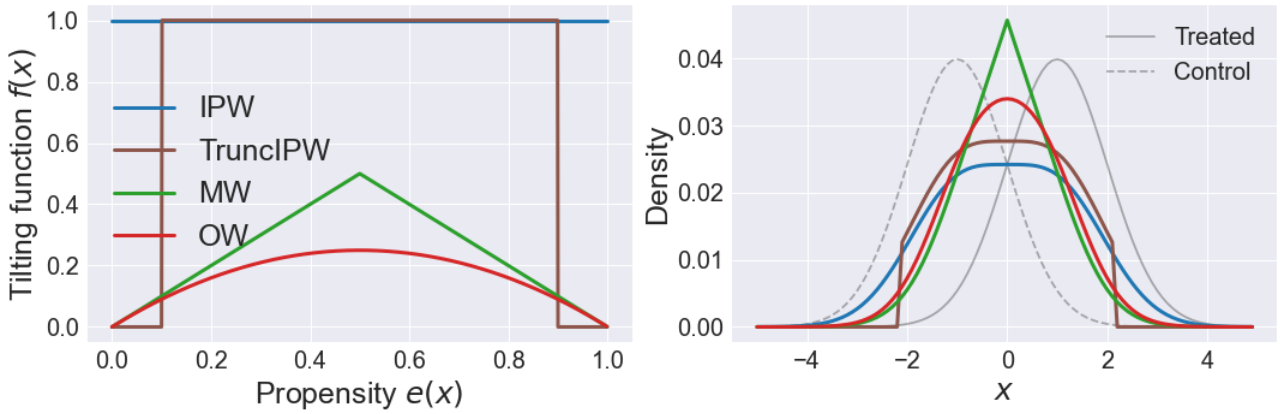

Figure S1: (Left) Tilting functions $f(x)$ used. (Right) Illustrative treatment group densities $p(x|T=t)$, and reweighted densities $g(x) \propto f(x)p(x)$ for different $f(x)$. TruncIPW, MW, and OW specifically emphasize regions of good overlap between the treatment and control groups.

Definition 2 (δ-strict overlap)**.** $\exists\delta \in (0, 0.5) : \forall x \in \mathcal{X}\ \delta < e(x) < 1-\delta$.

Definition 3 ($\epsilon_{\text{PEHE},g}$)**.**

$$\epsilon_{\text{PEHE},g}(\hat{\tau}) \triangleq \int_{\mathcal{X}} (\tau(x) - \hat{\tau}(x))^2 g(x)dx,$$

where $\tau(x) \triangleq \mathbb{E}[Y(1) - Y(0)|X = x]$ is the true individual treatment effect, and $\hat{\tau}$ is an estimate of $\tau(x)$. We often omit $\hat{\tau}$ from $\epsilon_{\text{PEHE}}(\hat{\tau})$ for brevity.

Proposition 4. *Assuming δ-strict overlap, for all the tilting functions presented in Table S1 (for $f(x) = \mathbb{1}(\xi < e(x) < 1 - \xi)$, the additional condition $\delta \geq \xi$ is required), we have:*

$$A_f \cdot \epsilon_{PEHE,g}(\hat{\tau}) \leq \epsilon_{PEHE,p}(\hat{\tau}) \leq B_f \cdot \epsilon_{PEHE,g}(\hat{\tau}),$$

where A_f and B_f are constants depending on f, $p(x) \triangleq \Pr(X_i = x)$, and $g(x) \overset{\triangle}{\propto} f(x)p(x)$

Proof. For all the tilting functions $f(x)$ in Table 1, we have $\sup_x f(x) < \infty$. Assuming δ-strict overlap (and for $f(x) = \mathbb{1}(\xi < e(x) < 1 - \xi)$, assuming $\delta \geq \xi$) we also get (for all $f(x)$ in Table S1) that $\inf_x f(x) > 0$. Since $g(x) \overset{\triangle}{\propto} f(x)p(x)$ by definition (with $p(x) \triangleq \Pr(X = x)$ the marginal covariate density), we may write:

$$\epsilon_{PEHE,g} \triangleq \int_{\mathcal{X}} (\tau(x) - \hat{\tau}(x))^2 g(x) dx \tag{20}$$

$$= \int_{\mathcal{X}} (\tau(x) - \hat{\tau}(x))^2 \frac{f(x)p(x)}{Z_f} dx \tag{21}$$

where $Z_f \triangleq \int_{\mathcal{X}} f(x)p(x)dx$. We may bound this expression above and below via:

$$\int_{\mathcal{X}} (\tau(x) - \hat{\tau}(x))^2 \frac{\inf_x[f(x)]p(x)}{Z_f} dx \leq \epsilon_{PEHE,g} \leq \int_{\mathcal{X}} (\tau(x) - \hat{\tau}(x))^2 \frac{\sup_x[f(x)]p(x)}{Z_f} dx \tag{22}$$

$$\Rightarrow \frac{\inf_x[f(x)]}{Z_f} \int_{\mathcal{X}} (\tau(x) - \hat{\tau}(x))^2 p(x) dx \leq \epsilon_{PEHE,g} \leq \frac{\sup_x[f(x)]}{Z_f} \int_{\mathcal{X}} (\tau(x) - \hat{\tau}(x))^2 p(x) dx \tag{23}$$

$$\Rightarrow \frac{\inf_x[f(x)]}{Z_f} \cdot \epsilon_{PEHE,p} \leq \epsilon_{PEHE,g} \leq \frac{\sup_x[f(x)]}{Z_f} \cdot \epsilon_{PEHE,p} \tag{24}$$

Defining $B_f \triangleq \frac{Z_f}{\inf_x[f(x)]}$ and $A_f \triangleq \frac{Z_f}{\sup_x[f(x)]}$:

$$\frac{1}{B_f} \cdot \epsilon_{PEHE,p} \leq \epsilon_{PEHE,g} \leq \frac{1}{A_f} \cdot \epsilon_{PEHE,p} \tag{25}$$

Which we may also write as:

$$A_f \cdot \epsilon_{PEHE,g} \leq \epsilon_{PEHE,p} \leq B_f \cdot \epsilon_{PEHE,g} \tag{26}$$

□

Proposition 4 gives a "two birds, one stone" property, whereby $\epsilon_{\text{PEHE},p}$ may also be minimized when $\epsilon_{\text{PEHE},g}$ is minimized. This is a possible justification for why targeting the population $g(x)$ (via minimizing an upper bound on $\epsilon_{\text{PEHE},g}$) may also benefit ITE estimation on the observed population $p(x)$.

1.5 $\epsilon_{\text{PEHE},g}$ bound

Here, we establish conditions for which the bound on $\epsilon_{\text{PEHE},g}$ (equation (8) in the main text) holds.

Proposition 5. *Assuming the encoder Φ is invertible, and assuming $\frac{1}{\alpha}\ell_{h,\Phi} \in G$ for a function class G and a constant α, we have:*

$$\epsilon_{\text{PEHE},g} \leq 2 \cdot (\epsilon_{F,g}^{T=1} + \epsilon_{F,g}^{T=0}) + \alpha \cdot \text{IPM}_G(g_\Phi(r|T = 1), g_\Phi(r|T = 0)) + C \triangleq B, \tag{27}$$

where C is a constant w.r.t. model parameters, and $g_\Phi(r|T = t)$ is the distribution induced by the invertible map Φ from the distribution $g(x|T = t)$ (for $t \in \{0, 1\}$).

Proof. The proof follows straightforwardly by applying Theorem 1 from Shalit et al. (2017) on the target population $g(x)$. □

2 Finite-sample objective

From equation (8) in the main text, we know $\epsilon_{\text{PEHE},g} \leq B$. From equation (9) in the main text, we have:

$$B \approx 2 \cdot (\epsilon_{F,g_\eta}^{T=1} + \epsilon_{F,g_\eta}^{T=0}) + \alpha \cdot \text{IPM}_G(g_{\Phi,\eta}(r|T=1), g_{\Phi,\eta}(r|T=0)) + C \tag{28}$$

We would like to obtain a finite-sample estimate of B (shown in equation (11) in the main text).

2.1 Finite-sample factual error terms $\epsilon_{F,g_\eta}^{T=1}$,$\epsilon_{F,g_\eta}^{T=0}$

We will start by estimating the first 2 terms in (28), choosing $\epsilon_{F,g_\eta}^{T=1}$ WLOG.

$$\epsilon_{F,g_\eta}^{T=1} \triangleq \int_{\mathcal{X}} \ell_{h,\Phi}(x,1) g_\eta(x|T=1) dx = \int_{\mathcal{X}} \ell_{h,\Phi}(x,1) \frac{w_\eta(x,1)p(x|T=1)}{Z_1} dx \tag{29}$$

where $Z_1 = \int_{\mathcal{X}} w_\eta(x,1)p(x|T=1)dx$, and $\ell_{h,\Phi}(x,t) \triangleq \int_{\mathcal{Y}} L(y, h(\Phi(x),t))\Pr(Y(t)=y|X=x)dy$, with $L(y,y') = (y-y')^2$.

We may approximate $\epsilon_{F,g_\eta}^{T=1}$ as:

$$\epsilon_{F,g_\eta}^{T=1} \approx \frac{1}{Z_1 \cdot n_1} \sum_{i \in \mathcal{B}: T_i=1} w_\eta(X_i,1)(Y_i - h(\Phi(X_i),1))^2 \tag{30}$$

where $\mathcal{B}$ is a sampled batch, and $n_1 \triangleq \sum_{i\in\mathcal{B}} T_i$.
The target distribution $g(x)$ is defined as $g(x) \triangleq \frac{f(x)p(x)}{Z}$ where $Z \triangleq \int_{\mathcal{X}} f(x)p(x)$. We make the following approximation for Z_1:

$$Z_1 \triangleq \int_{\mathcal{X}} w_\eta(x,1)p(x|T=1) \approx \int_{\mathcal{X}} w(x,1)p(x|T=1)dx = \int_{\mathcal{X}} \frac{f(x)}{e(x)} p(x|T=1)dx \tag{31}$$

$$= \int_{\mathcal{X}} \frac{f(x)}{\Pr(T=1)} p(x)dx = \frac{Z}{\Pr(T=1)} \int_{\mathcal{X}} \frac{f(x)p(x)}{Z} dx = \frac{Z}{\Pr(T=1)} \int_{\mathcal{X}} g(x)dx \tag{32}$$

$$= \frac{Z}{\Pr(T=1)} \approx \frac{Z \cdot N}{N_1} \tag{33}$$

where N is the number of samples in the dataset, and $N_1 = \sum_{i=1}^{N} T_i$ is the number of treatment samples in the dataset. We explicitly construct the batches $\mathcal{B}$ of size n such that $n_1/n = N_1/N$, so we get:

$$Z_1 \approx \frac{Z \cdot n}{n_1} \tag{34}$$

Finally, we plug in the above approximation of Z_1 into (30) to get:

$$\epsilon_{F,g_\eta}^{T=1} \approx \frac{1}{Z \cdot n} \sum_{i \in \mathcal{B}: T_i=1} w_\eta(X_i,1)(Y_i - h(\Phi(X_i),1))^2 \tag{35}$$

Similarly, we may approximate $\epsilon_{F,g_\eta}^{T=0}$ as:

$$\epsilon_{F,g_\eta}^{T=0} \approx \frac{1}{Z \cdot n} \sum_{i \in \mathcal{B}: T_i=0} w_\eta(X_i,0)(Y_i - h(\Phi(X_i),0))^2 \tag{36}$$

We also tried the approximations $Z_1 \approx \frac{1}{n_1} \sum_{i\in\mathcal{B}:T_i=1} w_\eta(X_i,1)$ and $Z_1 \approx \frac{1}{N_1} \sum_{i:T_i=1} w_\eta(X_i,1)$ (and similar approximations for Z_0), but they did not work well in practice.

2.2 Finite-sample IPM term

Finally, we seek a Monte-Carlo approximation of the third term in (28). Recalling the definition of $g_\eta(x|T=1)$, we have:

$$g_\eta(x|T=1) \triangleq \frac{w_\eta(x,1)p(x|T=1)}{Z_1} \tag{37}$$

where $Z_1 \triangleq \int_{\mathcal{X}} w_\eta(x,1)p(x|T=1)dx$.

We assume that $\Phi(\cdot) : \mathcal{X} \to \mathcal{R}$ is an invertible transformation (with inverse Ψ), so it induces distributions $g_{\Phi,\eta}(r|T=1)$ and $p_\Phi(r|T=1)$ from $g_\eta(x|T=1)$ and $p(x|T=1)$, respectively. From the change of variables formula:

$$g_{\Phi,\eta}(r|T=1) = g_\eta(\Psi(r)|T=1) \cdot |\det(\Psi')| \tag{38}$$

where Ψ' is the Jacobian of Ψ, and $\det(\cdot)$ is the determinant. From (37), we get:

$$g_{\Phi,\eta}(r|T=1) = \frac{w_\eta(\Psi(r),1)}{Z_1} \cdot p(\Psi(r)|T=1) \cdot |\det(\Psi')| \tag{39}$$

By the change of variables formula on the last 2 terms above, we get:

$$g_{\Phi,\eta}(r|T=1) = \frac{w_\eta(\Psi(r),1)}{Z_1} \cdot p_\Phi(r|T=1) \tag{40}$$

We may approximate $g_{\Phi,\eta}(r|T=1)$ from samples in a batch $\mathcal{B}$ as:

$$g_{\Phi,\eta}(r|T=1) \approx \frac{1}{\sum_{i\in\mathcal{B}:T_i=1} w_\eta(X_i,1)/Z_1} \sum_{i\in\mathcal{B}:T_i=1} \frac{w_\eta(X_i,1)}{Z_1}\delta(r-\Phi(X_i)) \tag{41}$$

$$= \frac{1}{\sum_{i\in\mathcal{B}:T_i=1} w_\eta(X_i,1)} \sum_{i\in\mathcal{B}:T_i=1} w_\eta(X_i,1)\delta(r-\Phi(X_i)) \triangleq \hat{g}_{\Phi,\eta}(r|T=1) \tag{42}$$

Where $\delta(r-z)$ is a point-mass centered at z. Similarly, we can approximate $g_{\Phi,\eta}(r|T=0)$ as:

$$g_{\Phi,\eta}(r|T=0) \approx \hat{g}_{\Phi,\eta}(r|T=0) \triangleq \frac{1}{\sum_{i\in\mathcal{B}:T_i=0} w_\eta(X_i,0)} \sum_{i\in\mathcal{B}:T_i=0} w_\eta(X_i,0)\delta(r-\Phi(X_i)) \tag{43}$$

2.3 Putting it all together

Plugging (35), (36), (42), and (43) into (28), we may write an approximation of the bound B (from (28)) as:

$$\frac{2}{Z\cdot n}\sum_{i\in\mathcal{B}} w_\eta(X_i,T_i)(Y_i - h(\Phi(X_i),T_i))^2 + \alpha\cdot \text{IPM}_G(\hat{g}_{\Phi,\eta}(r|T=1), \hat{g}_{\Phi,\eta}(r|T=0)) + C \tag{44}$$

The above has the same argmin as:

$$\mathcal{L}(h,\Phi,\mathcal{B}) = \frac{1}{n}\sum_{i\in\mathcal{B}} w_\eta(X_i,T_i)(Y_i - h(\Phi(X_i),T_i))^2 + \alpha'\cdot \text{IPM}_G(\hat{g}_{\Phi,\eta}(r|T=1), \hat{g}_{\Phi,\eta}(r|T=0)) \tag{45}$$

for some constant α' (which we leave as α in the main text to avoid introducing more notation).
This is the finite-sample objective presented in equation (11) of the main text – the version presented here is over a mini-batch $\mathcal{B}$, but we omitted this detail from the main text for simplicity.

2.4 Weighted Integral Probability Metric (IPM) computation

As a reminder, IPMs (Müller, 1997) are defined as follows:

$$\text{IPM}_G(u,v) = \sup_{m\in G} \int_{\mathcal{R}} m(r)[u(r)-v(r)]dr \tag{46}$$

where G is a function class, and u and v are probability measures. In our implementation, similar to Shalit et al. (2017), we use two kinds of IPMs: namely, the Wasserstein distance, by setting the function class $G = \{m : ||m||_L \leq 1\}$ to be the set of 1-Lipschitz functins, and the Maximum Mean Discrepancy (MMD; Gretton et al., 2012), by setting $G = \{m : ||m||_{\mathcal{H}} = 1\}$ to be the set of norm-1 functions in a reproducing kernel Hilbert space $\mathcal{H}$. In this section, we provide details for how to compute these IPMs between the reweighted distributions $g_{\Phi,\eta}(r|T=1)$ and $g_{\Phi,\eta}(r|T=0)$, which is the last term in our objective in equation (11) of the main text.

Finite-sample weighted MMD First, suppose the class of functions $G = \{m : ||m||_{\mathcal{H}} = 1\}$ is the set of norm-1 functions in a reproducing kernel Hilbert space (RKHS) $\mathcal{H}$ with corresponding kernel $k(\cdot,\cdot)$. IPM_G is then equivalent to the Maximum-Mean Discrepancy (MMD). From Lemma 4 in Gretton et al. (2012), the squared MMD is equal to:

$$\text{MMD}^2(p,q) = ||\mu_p - \mu_q||^2_{\mathcal{H}} \tag{47}$$

$$= \langle \mu_p, \mu_p \rangle_{\mathcal{H}} + \langle \mu_q, \mu_q \rangle_{\mathcal{H}} - 2 \cdot \langle \mu_p, \mu_q \rangle_{\mathcal{H}} \tag{48}$$

where $\mu_p(\cdot) \triangleq \mathbb{E}_{x\sim p}[k(\cdot, x)]$ and μ_q is defined similarly.
We now wish to get a finite sample estimate of $\text{MMD}^2(g_{\Phi,\eta}(r|T=1), g_{\Phi,\eta}(r|T=0))$. Assuming Φ is invertible with inverse Ψ, from equation (40), we have:

$$g_{\Phi,\eta}(r|T=1) = \frac{w_\eta(\Psi(r),1)p_\Phi(r|T=1)}{Z_1} \tag{49}$$

$$g_{\Phi,\eta}(r|T=0) = \frac{w_\eta(\Psi(r),0)p_\Phi(r|T=0)}{Z_0} \tag{50}$$

Where $Z_t = \int_{\mathcal{R}} w_\eta(\Psi(r), t)p_\Phi(r|T=t)$ for $t \in \{0,1\}$.
WLOG, we now seek a finite-sample estimate of $\langle \mu_1, \mu_1 \rangle_{\mathcal{H}}$, where $\mu_1 \triangleq \mathbb{E}_{R\sim g_{\Phi,\eta}(r|T=1)}[k(\cdot, R)]$.

$$\mu_1(\cdot) \approx \sum_{i\in\mathcal{B}:T_i=1} \frac{w_\eta(X_i,1)}{\sum_{i\in\mathcal{B}:T_i=1} w_\eta(X_i,1)} k(\cdot, \Phi(X_i)) \tag{51}$$

$$\Rightarrow \langle \mu_1, \mu_1 \rangle_{\mathcal{H}} \approx \frac{\sum_{i\in\mathcal{B}:T_i=1}\sum_{j\in\mathcal{B}:T_j=1} w_\eta(X_i,1)w_\eta(X_j,1)k(\Phi(X_i),\Phi(X_j))}{[\sum_{i\in\mathcal{B}:T_i=1} w_\eta(X_i,1)]^2} \tag{52}$$

Using the V-statistic version (Gretton et al., 2012) of the above, we get:

$$\langle \mu_1, \mu_1 \rangle_{\mathcal{H}} \approx \frac{\sum_{i\in\mathcal{B}:T_i=1}\sum_{j\in\mathcal{B}:T_j=1,j\neq i} w_\eta(X_i,1)w_\eta(X_j,1)k(\Phi(X_i),\Phi(X_j))}{\sum_{i\in\mathcal{B}:T_i=1}\sum_{j\in\mathcal{B}:T_j=1,j\neq i} w_\eta(X_i,1)w_\eta(X_j,1)} \tag{53}$$

Similarly, we can approximate $\langle \mu_0, \mu_0 \rangle_{\mathcal{H}}$ as:

$$\langle \mu_0, \mu_0 \rangle_{\mathcal{H}} \approx \frac{\sum_{i\in\mathcal{B}:T_i=0}\sum_{j\in\mathcal{B}:T_j=0,j\neq i} w_\eta(X_i,0)w_\eta(X_j,0)k(\Phi(X_i),\Phi(X_j))}{\sum_{i\in\mathcal{B}:T_i=0}\sum_{j\in\mathcal{B}:T_j=0,j\neq i} w_\eta(X_i,0)w_\eta(X_j,0)] } \tag{54}$$

Finally, we similarly approximate $\langle \mu_1, \mu_0 \rangle_{\mathcal{H}}$ as:

$$\langle \mu_1, \mu_0 \rangle_{\mathcal{H}} \approx \frac{\sum_{i\in\mathcal{B}:T_i=1}\sum_{j\in\mathcal{B}:T_j=0} w_\eta(X_i,1)w_\eta(X_j,0)k(\Phi(X_i),\Phi(X_j))}{\sum_{i\in\mathcal{B}:T_i=1}\sum_{j\in\mathcal{B}:T_j=0} w_\eta(X_i,1)w_\eta(X_j,0)] } \tag{55}$$

Finally, we get the finite-sample estimate of $\text{MMD}^2(g_\eta(r|T=1), g_\eta(r|T=0))$ via:

$$\text{MMD}^2(g_\eta(r|T=1), g_\eta(r|T=0)) \approx (53) + (54) - 2 \cdot (55) \tag{56}$$

In practice we set $k(\cdot,\cdot)$ to either be a linear kernel, i.e. $k(R_i, R_j) = R_i^T R_j$, or a RBF kernel, i.e. $k(R_i, R_j) = \exp(-\frac{||R_i - R_j||_2^2}{\sigma^2})$, where σ is set to 0.1.

Finite-sample weighted Wasserstein distance For the finite sample approximation of the weighted Wasserstein distance, we use Algorithm 3 of Cuturi & Doucet (2013) (shown here in Algorithm 1 for convenience), with the entropic regularization strength set to $\lambda = 10$, and vectors $a \in \mathbb{R}^{n_1}$, $b \in \mathbb{R}^{n_0}$ and matrix $M \in \mathbb{R}^{n_1 \times n_0}$ set to:

$$a^{(i)} = \frac{w_\eta(X_i, 1)}{\sum_{k \in \mathcal{B}: T_k = 1} w_\eta(X_k, 1)}; \quad b^{(j)} = \frac{w_\eta(X_j, 0)}{\sum_{k \in \mathcal{B}: T_k = 0} w_\eta(X_k, 0)}; \quad M^{(i,j)} = ||\Phi(X_i) - \Phi(X_j)||_2 \tag{57}$$

We fix the number of Sinkhorn iterations to $S = 10$.

Algorithm 1: Sinkhorn-Knopp Algorithm for weighted Wasserstein distance approximation

Input batch $\mathcal{B}$, entropic regularization parameter $\lambda \in \mathbb{R}$, number of Sinkhorn iterations S, encoder $\Phi(\cdot)$, propensity score parameters η

$n_1 = \sum_{i \in \mathcal{B}} T_i; \quad n_0 = \sum_{i \in \mathcal{B}} (1 - T_i);$

Compute weight vectors $a \in \mathbb{R}^{n_1}, b \in \mathbb{R}^{n_0}$ of empirical approximations $\hat{g}_{\Phi,\eta}(r|T=1), \hat{g}_{\Phi,\eta}(r|T=0)$, as:
$a^{(i)} = \frac{w_\eta(X_i,1)}{\sum_{k \in \mathcal{B}: T_k = 1} w_\eta(X_k,1)} \ \ \forall i \in \mathcal{B} : T_i = 1; \quad b^{(j)} = \frac{w_\eta(X_j,0)}{\sum_{k \in \mathcal{B}: T_k = 0} w_\eta(X_k,0)} \ \ \forall j \in \mathcal{B} : T_j = 0;$

Compute pairwise distance matrix $M \in \mathbb{R}^{n_1 \times n_0}$ between treatment & control representations, as:
$M^{(i,j)} = ||\Phi(X_i) - \Phi(X_j)||_2 \ \ \forall i \in \mathcal{B} : T_i = 1, \forall j \in \mathcal{B} : T_j = 0;$

$K = \exp(-\lambda M);$ % elementwise exponential
$\tilde{K} = \text{diag}(a^{-1})K;$
Initialize $u = a;$
for $s \in [0, ..., S-1]$ **do**
  $u = 1./(\tilde{K}(b./(K^T u)));$ % Sinkhorn iterations
end for
$v = b./(K^T u).$
$T^\star_\lambda = \text{diag}(u) K \text{diag}(v);$

return $\text{Wass}(\hat{g}_{\Phi,\eta}(r|T=1), \hat{g}_{\Phi,\eta}(r|T=0)) \approx \sum_{i,j} T^{\star(i,j)}_\lambda M^{(i,j)}$

3 Experimental details

3.1 Toy experiment

Data-generating parameters We specify β_0, β_τ,and $\gamma \in \mathbb{R}^p$ (from Section 4.1 in the main text) as follows:

$$\beta_0 \triangleq \tilde{\beta_0} \cdot \mathbb{1}_{\mathcal{B}}; \qquad \beta_\tau \triangleq \tilde{\beta_\tau} \cdot \mathbb{1}_{\mathcal{B}}; \qquad \gamma \triangleq \tilde{\gamma} \cdot \mathbb{1}_{\mathcal{G}} \tag{58}$$

where $\tilde{\beta_0}, \tilde{\beta_\tau}, \tilde{\gamma} \in \mathbb{R}$, $\mathcal{B} \triangleq supp(\beta_0) = supp(\beta_\tau)$, and $\mathcal{G} \triangleq supp(\gamma)$.

Note that $\tilde{\beta_0}, \tilde{\beta_\tau}, \tilde{\gamma}$ can be used to control the magnitudes of $\beta_0, \beta_\tau, \gamma$ respectively. Table S2 indicates the value of every parameter used to generate the toy dataset described in the previous section. With the values in Table S2, we get 33 datasets (each simulated 20 times), indexed by $\tilde{\gamma}$ (which controls the imbalance) and $\Omega \triangleq |\mathcal{B} \cap \mathcal{G}|$ (which controls the level of confounding).

Model hyperparameters For the purposes of the toy experiment, we fix a regression neural network architecture, as well as a propensity score network architecture. The only hyperparameter we vary is α, which is the strength of the IPM regularization term. This was done for reasons of time efficiency, as well as to have an "apples-to-apples" comparison between different weighting schemes used in the regression loss. The model hyperparameter values used are shown in Table S3.

Table S2: Data-generating parameters for Section 4.1 in the main text

Parameter	**Description**	**Value/Range**
N	number of data points	525/225/250 (train/val/test)
p	dimension of covariates	50
p^*	non-zero dimensions in $\beta_0, \beta_\tau, \gamma$	20
σ_X^2	variance of covariates	0.05
σ_Y	variance of additive Gaussian noise in potential outcomes $Y_i(0), Y_i(1)$	1.0
ρ	correlation between covariates	0.3
$\tilde{\beta}_0$	effective magnitude of β_0	1.0
$\tilde{\beta}_\tau$	effective magnitude of β_τ	0.3
$\tilde{\gamma}$	imbalance parameter	$\{0, 0.5, 1.00, ..., 5.00\}$
$\mathcal{B}$	support of β_0, β_τ	
$\mathcal{G}$	support of γ	
Ω	confounding parameter: $\lvert\mathcal{B} \cap \mathcal{G}\rvert$	$\{0, 10, 20\}$
θ	True ATE	3.0

Table S3: Model hyperparameter ranges for toy experiment (middle column) and "real" datasets (IHDP/ACIC, right column). "Wass" is the Wasserstein distance, "MMD-linear" is the MMD with a linear kernel, "MMD-RBF" is the MMD with an RBF kernel. $e_\eta(\cdot)$ is the fully-connected neural network predicting the propensity score. "ELU" is the exponential linear unit activation, "ReLU" is the rectified linear unit activation.

Hyperparameter	**Value/Range**	
	Toy experiment	IHDP & ACIC2016
α (strength of IPM term)	{0, 0.01, 0.1, 1, 10, 100}	$\{10^{k/2}\}_{k=-10}^{6}$
IPM used	Wass	{Wass, MMD-linear, MMD-RBF}
Num. hidden layers in $\Phi(\cdot)$	1	{1,2,3}
Num. hidden layers in $h(\cdot, t)$	1	{1,2,3}
Num. hidden layers in $e_\eta(\cdot)$	1	{1,2,3}
$\Phi(\cdot)$ hidden layer dim.	100	{20,50,100,200}
$h(\cdot, t)$ hidden layer dim.	100	{20,50,100,200}
$e_\eta(\cdot)$ hidden layer dim.	10	{10,20,30}
$h(\cdot, t), \Phi$ hidden-layer activations	ELU	ELU
$e_\eta(\cdot)$ hidden-layer activations	ReLU	ReLU
Batch size	200	200
Learning rate	0.001	0.001
Optimizer	Adam	Adam

3.2 Infant Health and Development Program (IHDP)

From the IHDP dataset (Hill, 2011), Shalit et al. (2017) made 2 datasets, named IHDP100 and IHDP1000[1]. We used the former (IHDP100) for parameter tuning/model selection, and the latter (IHDP1000) for evaluation. For the IHDP dataset, we randomly sampled 100 hyperparameter configurations (the hyperparameter ranges are shown in Table S3) – for each sampled configuration, we train 3 models (with respective weight schemes MW, OW, TruncIPW). We train on the IHDP100 dataset, perform early stopping based on the validation loss, and we

[1]both datasets were downloaded from https://www.fredjo.com/

select 3 best models (one for each weighting scheme) according to $\epsilon^{\text{NN}}_{\text{PEHE},p}$ on the validation set, where:

$$\epsilon^{\text{NN}}_{\text{PEHE},p} \triangleq \frac{1}{N}\sum_{i=1}^{N}[(1-2\cdot T_i)(Y_{j(i)}-Y_i) - (h(\Phi(X_i,1)-h(\Phi(X_i,0)]^2 \tag{59}$$

$$j(i) \triangleq \underset{j:T_j=1-T_i}{\text{argmin}}\ ||X_i - X_j||_2 \tag{60}$$

This is a proxy for $\epsilon_{\text{PEHE},p}$ which does not make use of counterfactual information. After the model tuning stage on IHDP100, we report 3 results (1 for each weight scheme) on the IHDP1000 dataset.

For the causal forest results in Section 4.3 of the main text, we obtained the representations and weights (obtained from our 3 best models) for IHDP100, and used them as input to a causal forest (CF) algorithm. We then compared the augmented CF models to a vanilla CF model on IHDP100. More details are provided in the section below.

3.3 Causal Forests

IHDP100 weight ablation In addition to comparing the vanilla CF with the CF augmented with learned weights representations, we add a comparison to the CF augmented with the representations only (*i.e.*, without weights). We find that the unweighted augmented CF ("CF+Φ" in Table S4) performs similarly to its weighted counterparts for the IHDP100 dataset.

Table S4: Causal forest (CF) results for IHDP100. The top block is a vanilla CF model. The middle block is a causal forest model using learned representations (denoted Φ) without weights (*i.e.*, the equivalent of CFRNet). The bottom block consists of causal forest models using the learned representations and weights. The bottom block rows are the weights used in the training objective and as the per-sample weights to train the CF.

	$\sqrt{\epsilon_{\text{PEHE},p}}$	$\epsilon_{\text{ATE},p}$
CF	3.54 ± .58	.47 ± .06
CF + Φ	1.52 ± .35	.20 ± .04
CF + Φ + MW	1.51 ± .31	.20 ± .03
CF + Φ + OW	1.59 ± .31	.19 ± .03
CF + Φ + TruncIPW	1.55 ± .35	.22 ± .03

Atlantic Causal Inference Competition 2016 (ACIC2016) In Section 4.3 of the main text, we considered the ACIC2016 dataset (Dorie et al., 2019), which comprises 77 datasets (we use 10 repetitions of each)[2], each with 4802 samples, and 58-dimensional covariates. ACIC2016 uses the same covariates for all the datasets, but different data-generating mechanisms for potential outcomes and treatment across datasets. We removed the categorical covariates (named x_2, x_{21}, x_{24} in the dataset), since our models are not equipped to handle categorical data. We standardized the remaining 55 covariate dimensions (*i.e.*, for each dimension we subtract the mean and divide by the standard deviation). We used the first 4000 samples for training, and the remaining 802 for testing. We used 30% of the training set for validation.

The tuning procedure is similar to the one described for IHDP. We use the first 10 (out of 77) datasets, with 1 repetition of each, as a tuning set. We pick 3 best models (one for each weighting scheme) according to the average $\epsilon^{\text{NN}}_{\text{PEHE},p}$ (across the 10 datasets) on the validation set. The hyperparameter ranges used for tuning are shown in Table S3. Hence, after tuning, we have 3 "best" models (1 for each weighting scheme), which we apply to the 77 datasets (with 10 repetitions).

We then use the obtained representations and weights as input to a causal forest (CF) model[3], and we compare the performance of the "vanilla" causal forest (*i.e.*, CF using only the original covariates) with the performance of the "augmented" CF models (*i.e.*, which use the learned representations and weights as input). More specifically,

[2]generated using https://github.com/vdorie/aciccomp/tree/master/2016, and setting `parameterNum` between 1 and 77, and `simulationNum` between 1 and 10.

[3]Implemented using the `causal_forest` function of the `GRF` package in R: https://CRAN.R-project.org/package=grf

for the augmented models, we use the learned representations as the "covariates", and we use the propensity-based weights as the per-sample weights to train the CF.

4 Additional Results

4.1 Toy experiment

Performance on target populations $g(x)$ In Section 4.1 of the main text, we measured the performance of our models on the observed population $p(x)$. Here, we extend this evaluation to the different target populations $g(x)$. We report performance using $\sqrt{\epsilon_{\text{PEHE},g}}$, computed via equation (5) from the main text. Figure S2 shows a plot of $\epsilon_{\text{PEHE},g}$ for all choices of $g(x)$, for all toy datasets, and for all weight schemes used during training. From Figure S2, we can see that each weighting scheme (row) tends to do well for the target population it was trained for – *i.e.*, within each row, the corresponding metric (color) is lowest (or close to the lowest). This provides some evidence for the fact that the models perform well on the target population they were trained for.

Double-Robust ATE estimation We may easily enhance the ATE estimate in equation (3) of the main text by accounting for model bias, using equation (6) from Mao et al. (2018). Specifically, we can define biases as:

$$b^{(t)} = \frac{1}{\sum_{i:T_i=t} w_\eta(X_i,t)} \sum_{i:T_i=t} w_\eta(X_i,t)[h(\Phi(X_i),t) - Y_i], \quad \text{for} \quad t \in \{0,1\} \tag{61}$$

which we may use to obtain a doubly-robust (Lunceford & Davidian, 2004) ATE estimate via:

$$\hat{\tau}^{\text{DR}}_{\text{ATE},g} = \hat{\tau}_{\text{ATE},g} - b^{(1)} + b^{(0)} \tag{62}$$

Note that $b^{(1)}, b^{(0)}$ are calculated using the training set only. We compute the target population ATE error via:

$$\epsilon^{\text{DR}}_{\text{ATE},g} = |\tau_{\text{ATE},g} - \hat{\tau}^{\text{DR}}_{\text{ATE},g}| \tag{63}$$

Figure S3 shows percent improvement of the double-robust estimator, computed as

$$\Delta^{\text{DR}}_g \triangleq \frac{\epsilon_{\text{ATE},g} - \epsilon^{\text{DR}}_{\text{ATE},g}}{\epsilon_{\text{ATE},g}} \tag{64}$$

on the (test) toy datasets. The double-robust ATE estimator (i.e. $\hat{\tau}^{\text{DR}}_{\text{ATE},g}$) enjoys some improvement in the ATE estimation error in most cases, though this is not true across all the toy datasets.

4.2 IHDP100 additional comparisons

In Table 2 of the main text, some of the listed methods (namely, RCFR and CFR-ISW) actually reported their results on IHDP100, whereas we reported performance of our methods on IHDP1000. For the sake of completeness, we add the comparison between our methods, RCFR, and CFR-ISW on the IHDP100 dataset, reported in the table below. These results are consistent with the results from Table 2 in the main text, namely that our proposed methods perform on-par with state-of-the-art methods from recent work.

Table S5: Results on IHDP100 test set. The top block consists of baselines from recent work. The bottom block is our proposed methods. Lower is better.

Model	$\sqrt{\epsilon_{\text{PEHE},p}}$	$\epsilon_{\text{ATE},p}$
CFR-ISW (Hassanpour & Greiner, 2019)	$.70 \pm .1$	$.19 \pm .03$
RCFR (Johansson et al., 2018)	$.67 \pm .05$	-
BWCFR-MW (Ours)	$.66 \pm .06$	$.18 \pm .02$
BWCFR-OW (Ours)	$.66 \pm .06$	$.16 \pm .02$
BWCFR-TruncIPW (Ours)	$.65 \pm .05$	$.16 \pm .02$

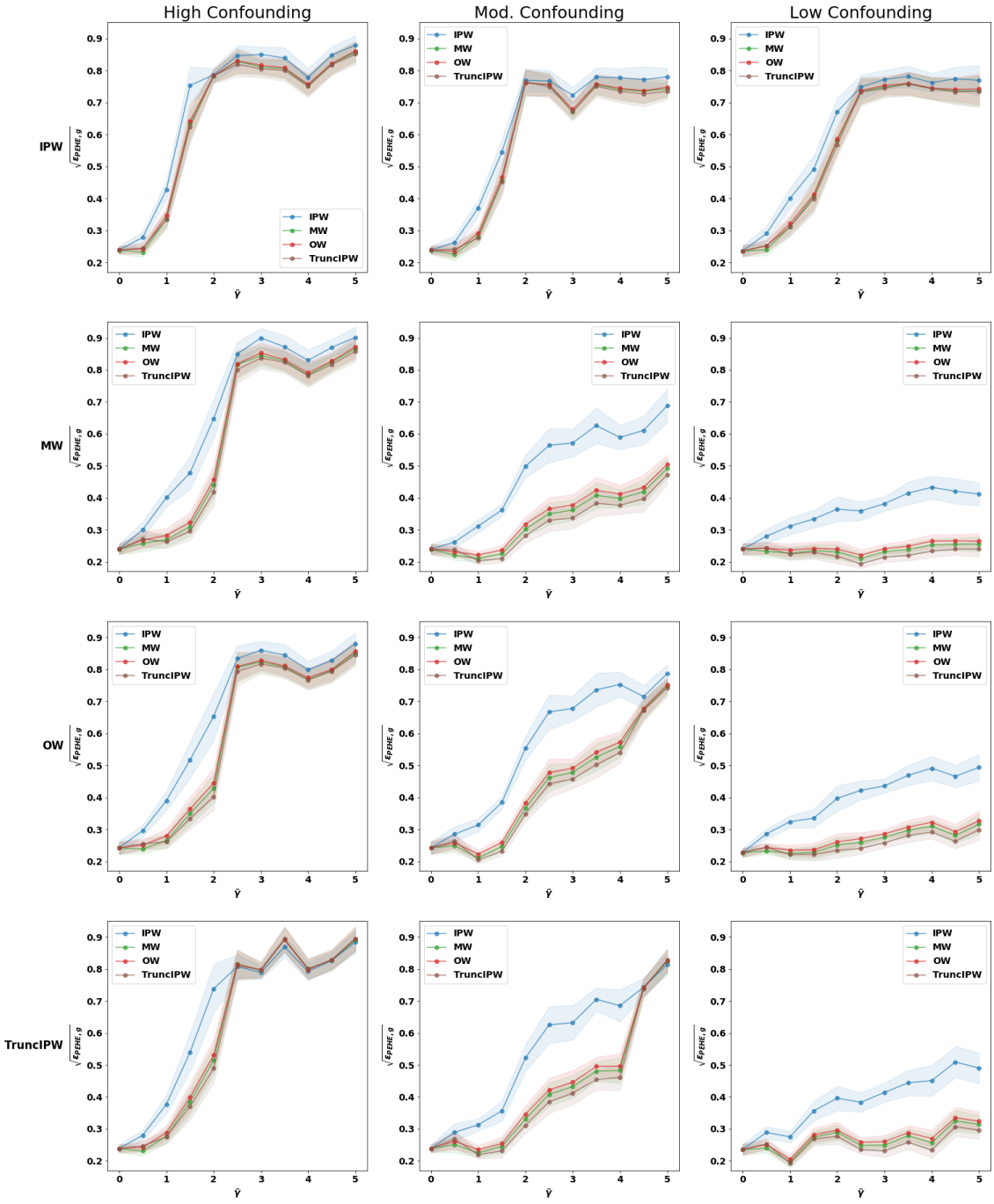

Figure S2: $\epsilon_{\text{PEHE},g}$ vs. $\tilde{\gamma}$. The colors are the different choices of $g(x)$ (to calculate $\sqrt{\epsilon_{\text{PEHE},g}}$ from equation (5) in the main text), and the rows are the weight schemes used during training (*i.e.*, in equation (11) of the main text).

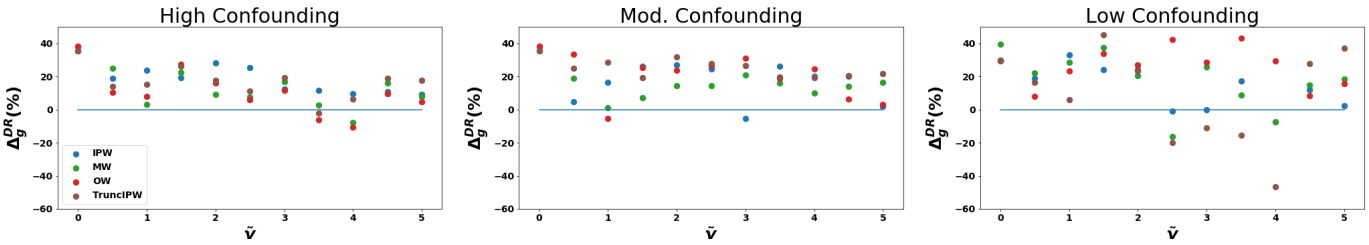

Figure S3: Improvement in ATE estimation (on the toy datasets) using the double-robust $\hat{\tau}^{\text{DR}}_{\text{ATE},g}$ from (62). The x-axis is the imbalance parameter $\tilde{\gamma}$, the y-axis is the percent improvement defined in (64). The colors are different choices of g (used for (i) the weight schemes during training, and (ii) to compute the target population metric $\epsilon_{\text{ATE},g}$. Positive values means the double-robust estimator $\hat{\tau}^{\text{DR}}_{\text{ATE},g}$ improves upon the vanilla estimator $\hat{\tau}_{\text{ATE},g}$.

4.3 Atlantic Causal Inference Competition 2016 (ACIC2016)

In this section, we carefully examine the results of Section 4.3 from the main text. Specifically, Figure S4 shows the performance of the proposed methods on each of the 77 datasets in ACIC2016, rather than an aggregate as shown in Table 3 of the main text. Figure S4 compares the performance of 3 types of models:

- A vanilla causal forest algorithm
- Our proposed deep methods
- A hybrid model consisting of a causal forest augmented with our learned representations and weights (obtained from $\Phi(x)$ and $w_\eta(x,t)$, respectively).

From Figure S4, we can see that the augmented causal forest consistently outperforms the vanilla causal forests for almost all of the 77 datasets, both in terms of $\sqrt{\epsilon_{\text{PEHE},p}}$ and $\epsilon_{\text{ATE},p}$. The neural network models perform (on average) better than the vanilla CF in terms of $\sqrt{\epsilon_{\text{PEHE},p}}$, but worse than the augmented CF. In terms of $\epsilon_{\text{ATE},p}$, the neural network models perform worse. A possible explanation for the poor performance of the neural network is that our tuning procedure was conducted only on the first 10 datasets (with 1 repetition each), whereas the ACIC2016 set comprises 77 datasets with 10 repetitions each. This suggests that we may stand to benefit from using hybrid approaches for ITE estimation, since the hybrid approach outperforms the 2 individual components it is comprised of, even though the deep models were not extensively tuned. Further, the hybrid models trained with Overlap Weights performed the best.

5 Computing infrastructure and details

All computation was done using Python and R. All neural network models were created and trained using Tensorflow 1.13.1 (Abadi et al., 2016). Computations were done on an NVIDIA Geforce GTX 1080 Ti. The reported results on IHDP1000, ACIC2016, and the toy dataset each took approximately 20 hours to run.

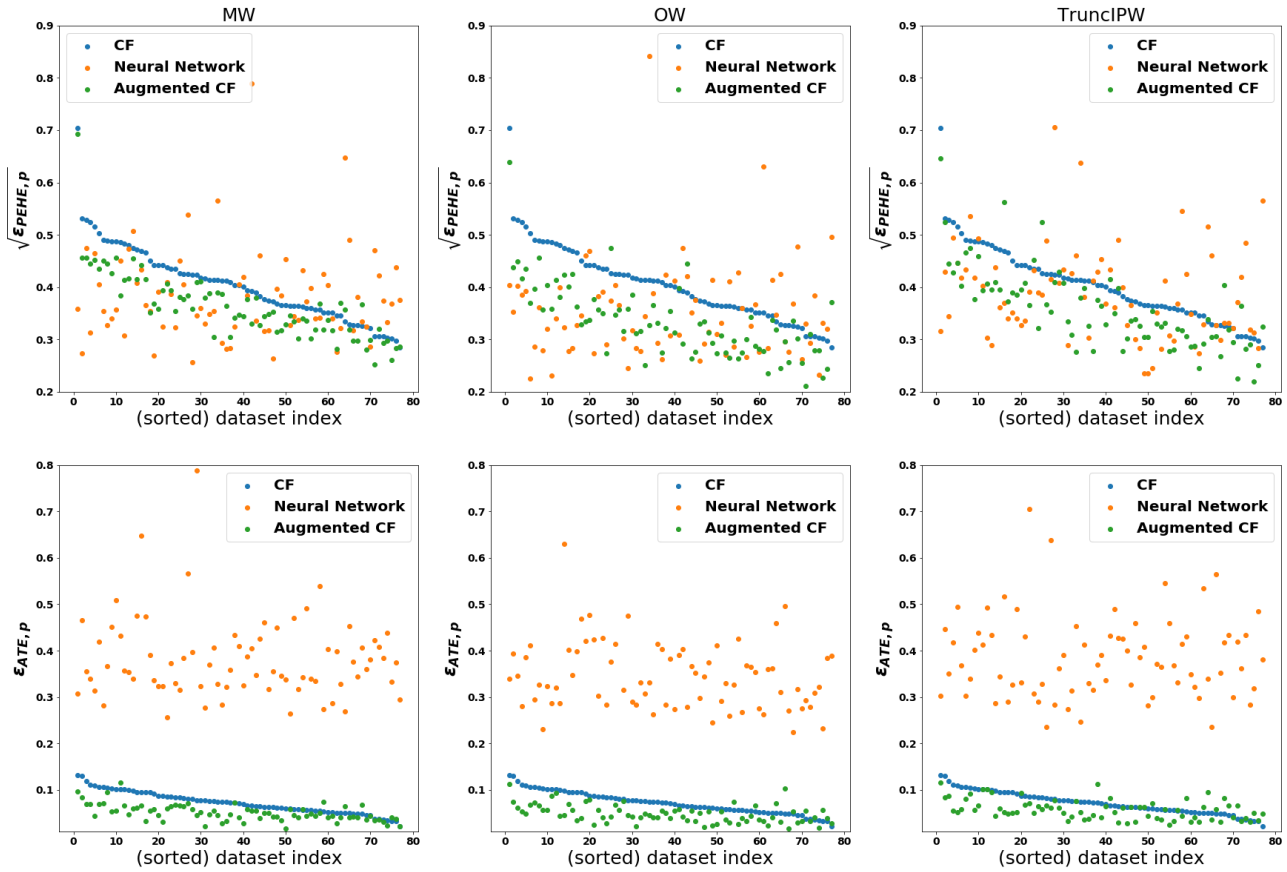

Figure S4: Per-dataset results on ACIC2016. The first row shows $\sqrt{\epsilon_{\text{PEHE},p}}$, and the second row shows $\epsilon_{\text{ATE},p}$. "CF" is the vanilla causal forest, "Augmented CF" is the CF trained using the learned representations and weights, and "Neural Network" are our proposed methods. The columns are the weight schemes used to (i) obtain the representations and weights for the augmented CF, and (ii) used to train the neural network models. The datasets were sorted in descending order according to the performance of the causal forest.

Bibliography

Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pp. 265–283, 2016.

Richard K. Crump, V. Joseph Hotz, Guido W. Imbens, and Oscar A. Mitnik. Dealing with limited overlap in estimation of average treatment effects. *Biometrika*, 96(1):187–199, 01 2009.

Marco Cuturi and Arnaud Doucet. Fast computation of wasserstein barycenters, 2013.

Vincent Dorie, Jennifer Hill, Uri Shalit, Marc Scott, Dan Cervone, et al. Automated versus do-it-yourself methods for causal inference: Lessons learned from a data analysis competition. *Statistical Science*, 34(1):43–68, 2019.

Alison L. Gibbs and Francis Edward Su. On choosing and bounding probability metrics. *International Statistical Review / Revue Internationale de Statistique*, 70(3):419–435, 2002. ISSN 03067734, 17515823. URL `http://www.jstor.org/stable/1403865`.

Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(25):723–773, 2012.

Negar Hassanpour and Russell Greiner. Counterfactual regression with importance sampling weights. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pp. 5880–5887, 2019.

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

Fredrik D Johansson, Nathan Kallus, Uri Shalit, and David Sontag. Learning weighted representations for generalization across designs. 2018.

Nathan Kallus, Xiaojie Mao, and Angela Zhou. Interval estimation of individual-level causal effects under unobserved confounding. volume 89 of *Proceedings of Machine Learning Research*, pp. 2281–2290. PMLR, 16–18 Apr 2019. URL `http://proceedings.mlr.press/v89/kallus19a.html`.

Fan Li, Kari Lock Morgan, and Alan M. Zaslavsky. Balancing covariates via propensity score weighting. *Journal of the American Statistical Association*, 113(521):390–400, 2018.

Liang Li and Tom Greene. A weighting analogue to pair matching in propensity score analysis. *The International Journal of Biostatistics*, 9(2):215 – 234, 2013.

J.K. Lunceford and M. Davidian. Stratification and weighting via the propensity score in estimation of causal treatment effects: A comparative study. *Statistics in Medicine*, 2004.

Huzhang Mao, Liang Li, and Tom Greene. Propensity score weighting analysis and treatment effect discovery. *Statistical Methods in Medical Research*, 28(8):2439–2454, June 2018. doi: 10.1177/0962280218781171. URL `https://doi.org/10.1177/0962280218781171`.

Alfred Müller. Integral probability metrics and their generating classes of functions. *Advances in Applied Probability*, 1997.

U. Shalit, F.D. Johansson, and D. Sontag. Estimating individual treatment effect: generalization bounds and algorithms. 2017.

Bharath K. Sriperumbudur, Kenji Fukumizu, Arthur Gretton, Bernhard Schölkopf, and Gert R. G. Lanckriet. On integral probability metrics, ϕ-divergences and binary classification, 2009.