Appendix

Contents

In this appendix, we first present theoretical proofs of our results stated in Section 4. We then discuss the connections between permutation weighting and other popular balancing weights. Finally, we present extended simulation results comparing weighting methods.

A. Proofs of Propositions

Before presenting our results, we first restate the definition of Bregman divergences, which we denote with B_g. Throughout, the specific form of B_g is a consequence of the choice of classifier.

Definition 3 (Bregman divergence (Bregman, 1967))**.** *Define the Bregman generator, $g : S \to \mathbb{R}$, to be a convex, differentiable function. The difference between the value of g at point s and the value of the first-order Taylor expansion of g around point s_0 evaluated at point s is given by $B_g(s, s_0) \equiv g(s) - g(s_0) - \langle s - s_0, \nabla g(s_0)\rangle$.*

We begin by bounding the difference between the estimated and true density ratio weights, $|w(a, \mathbf{x}) - \hat{w}(a, \mathbf{x})|$ in terms of the classifier regret by using a Pinsker-type inequality. This result will be used throughout in our technical proofs.

Lemma A.1. *For any $(a, \mathbf{x})$, we have that $|w(a, \mathbf{x}) - \hat{w}(a, \mathbf{x})| \leq \frac{2}{\sqrt{g''(1)}}\sqrt{\text{reg}(\hat{\eta}; \mathcal{D}, \lambda)}$.*

Proof. Following Appendix B of Menon and Ong (2016), let $\underline{L} : [0, 1] \to \mathbb{R}$ be a concave differentiable function which provides the conditional Bayes risk for the classifier loss, i.e.,

$$\lambda_{-1}(p) = \underline{\mathrm{L}}(p) - p \cdot \underline{\mathrm{L}}'(p) \text{ and } \lambda_1(p) = \underline{\mathrm{L}}(p) + (1 - p) \cdot \underline{\mathrm{L}}'(p)$$

Recall, that the risk for the classifier $\hat{\eta}$ under loss λ is then

$$\begin{aligned} 2 \cdot \mathbb{L}(\hat{\eta}; \mathcal{D}, \lambda) &= \mathbb{E}_P\left[\lambda_1(\hat{\eta}(a, \mathbf{x}))\right] + \mathbb{E}_Q\left[\lambda_{-1}(\hat{\eta}(a, \mathbf{x}))\right] \\ &= \mathbb{E}_Q\left[w(a, \mathbf{x}) \cdot \lambda_1(\hat{\eta}(a, \mathbf{x})) + \lambda_{-1}(\hat{\eta}(a, \mathbf{x}))\right] \end{aligned}$$

Let $g(z) = -(1+z) \cdot \underline{L}\left(\frac{z}{1+z}\right)$. Since $\hat{w} = \frac{\hat{\eta}}{1-\hat{\eta}}$,

$$g'(\hat{w}) = -(1-\hat{\eta}) \cdot \underline{\mathrm{L}}'(\hat{\eta}) - \underline{\mathrm{L}}(\hat{\eta}) = -\lambda_1(\hat{\eta})$$
$$\hat{w} \cdot g'(\hat{w}) - g(\hat{w}) = \lambda_{-1}(\hat{\eta})$$

Therefore, we can write

$$2 \cdot \mathbb{L}(\hat{\eta}; \mathcal{D}, \lambda) = \mathbb{E}_Q\left[-w \cdot g'(w) + \hat{w} \cdot g'(\hat{w}) - g(\hat{w})\right]$$

Then, by Reid and Williamson (2011), the Bayes risk for λ is then given as

$$\begin{aligned} 2 \cdot \mathbb{L}^*(\mathcal{D}, \lambda) &= 2 \cdot \min_{\hat{\eta}} \mathbb{L}(\hat{\eta}; \mathcal{D}, \lambda) \\ &= -I_g(P, Q) \\ &= \mathbb{E}_Q[-g(w)], \end{aligned}$$

where I_g is an f-divergence with generator g. Thus, the regret can be written as

$$\begin{aligned} 2 \cdot \mathrm{reg}(\hat{\eta}; \mathcal{D}, \lambda) &= 2 \cdot \mathbb{L}(\hat{\eta}; \mathcal{D}, \lambda) - 2 \cdot \mathbb{L}^*(\mathcal{D}, \lambda) \\ &= \mathbb{E}_Q\left[-w \cdot g'(w) + \hat{w} \cdot g'(\hat{w}) - g(\hat{w}) + g(w)\right] \\ &= \mathbb{E}_Q\left[B_g(w, \hat{w})\right] \end{aligned} \tag{3}$$

This, in turn, implies that the f-divergence, I_g can be written as

$$I_g(P, Q) = -2 \cdot \mathbb{L}^*(\mathcal{D}, \lambda) = \mathbb{E}_Q\left[B_g(w, \hat{w})\right] - 2 \cdot \mathbb{L}(\hat{\eta}; \mathcal{D}, \lambda) \tag{4}$$

Under additional assumptions on g, the following theorem, due to Gilardoni (2008), provides a bound in terms of the total variation distance, $V := V(P, Q)$.

Theorem A.1 (Pinsker type inequality for f-divergences.)**.** *Suppose that the convex function g is differentiable up to order 3 at $u = 1$ with $g''(1) > 0$. Let D_g denote the f-divergence generated by g. Then, $D_g \geq \frac{g''(1)}{2} V^2$. The constant $\frac{g''(1)}{2}$ is best possible.*

Applying Theorem A.1 to equation (4) gives

$$I_g(P, Q) = \mathbb{E}_Q\left[B_g(w, \hat{w})\right] - 2 \cdot \mathbb{L}(\hat{\eta}; \mathcal{D}, \lambda) \geq \frac{g''(1)}{2} V^2.$$

Therefore.

$$\begin{aligned} V &\leq \sqrt{\frac{2\left(\mathbb{E}_Q\left[B_g(w, \hat{w})\right] - 2 \cdot \mathbb{L}(\hat{\eta}; \mathcal{D}, \lambda)\right)}{g''(1)}} \\ &\leq \sqrt{\frac{2}{g''(1)} \mathbb{E}_Q\left[B_g(w, \hat{w})\right]}. \end{aligned}$$

This can be related directly to the classification problem considered in this paper with an application of equation (3) as shown above, yielding

$$V \leq \frac{2}{\sqrt{g''(1)}} \sqrt{\mathrm{reg}(\hat{\eta}; \mathcal{D}, \lambda)}$$

Then, by definition of the total variation distance, this inequality holds for any $|w(a, \mathbf{x}) - \hat{w}(a, \mathbf{x})|$. □

In the table below, we verify the conditions of the generator g required for A.1 corresponding to different f-divergences:

With this lemma, we can prove our previously stated results on the bias and variance of the permutation weighting estimator, leading to consistency.

Divergence	$g(u)$	$g''(1)$	$g'''(1)$
Kullback-Leibler	$u\log(u)$	1	-1
Triangular Discrimination	$\frac{(u-1)^2}{u+1}$	1	-3/2
Jensen-Shannon	$\frac{u}{2}\log\frac{u}{u+1} - \frac{1}{2}\log\frac{u+1}{4}$	1/4	-3/8
Jeffreys	$(u-1)\log(u)$	2	-3
Hellinger	$(\sqrt{u}-1)^2$	1/2	-3/4
Pearson $\mathcal{X}^2$	$(u-1)^2$	2	0

Proposition 4.1 (Bias of PW)**.** *Let $\mathbb{E}_P$ and $\mathbb{E}_Q$ denote the expectation under the distributions P and Q, respectively. The bias of the dose response function $\mathbb{E}_P[y\hat{w}]$ with respect to $\mathbb{E}_Q[y]$ is bounded by*

$$|\mathbb{E}_Q[y] - \mathbb{E}_P[y\hat{w}]| \leq \mathbb{E}_P\left[\frac{2|y|}{\sqrt{g''(1)}}\kappa_r\right],$$

where $g(\cdot)$ is a Bregman generator and $\kappa_r = \sqrt{\text{reg}(\hat{\eta};\mathcal{D},\lambda)}$

Proof. Let $w(a_i,\mathbf{x}_i) = \frac{p(a_i)p(\mathbf{x}_i)}{p(a_i,\mathbf{x}_i)}$ and $\hat{w}(a(a_i,\mathbf{x}_i)$ be the empirical estimate of $w(a_i,\mathbf{x}_i)$.

$$\begin{aligned}
\mathbb{E}_P[y\hat{w}(a,\mathbf{x})] &= \mathbb{E}_P\left[y_i(w(a_i,\mathbf{x}_i) + (\hat{w}(a_i,\mathbf{x}_i) - w(a_i,\mathbf{x}_i)))\right] \\
&= \mathbb{E}_P[y_i w(a_i,\mathbf{x}_i)] + \mathbb{E}_P\left[y_i(\hat{w}(a_i,\mathbf{x}_i) - w(a_i,\mathbf{x}_i))\right] \\
&= \mathbb{E}_Q[y_i] + \mathbb{E}_P\left[y_i(\hat{w}(a_i,\mathbf{x}_i) - w(a_i,\mathbf{x}_i))\right]
\end{aligned}$$

By Lemma A.1, the bias can then be written as

$$\begin{aligned}
\mathbb{E}_P\left[y_i(\hat{w}(a_i,\mathbf{x}_i) - w(a_i,\mathbf{x}_i))\right] &\leq \mathbb{E}_P\left[|y_i||\hat{w}(a_i,\mathbf{x}_i) - w(a_i,\mathbf{x}_i|\right] \\
&\leq \mathbb{E}_P\left[|y_i|\frac{2}{\sqrt{g''(1)}}\sqrt{\text{reg}(\hat{\eta};\mathcal{D},\lambda)}\right]
\end{aligned}$$

□

Proposition 4.2 (Variance)**.** *Let $\mathbb{V}_Q[y]$ denote the variance of Y under the distribution q. $\mathbb{V}_Q[y]$ is bounded by*

$$\mathbb{V}_Q[y] \leq \frac{1}{n}\mathbb{E}_Q[y^2] + \frac{4\kappa_r}{n\sqrt{g''(1)}}\mathbb{E}_P\left[y^2 w + \frac{y^2}{\sqrt{g''(1)}}\kappa_r\right]$$
$$\text{where } \kappa_r = \sqrt{\text{reg}(\hat{\eta};\mathcal{D},\lambda)}$$

Proof. The proofs follows via Lemma A.1. We first note that a trivial upper bound for the variance is given by the second moment. We consider the second moment of the estimator given as

$$\begin{aligned}
\mathbb{E}_P\left[(y\hat{w})^2\right] &= \mathbb{E}_P\left[y^2\hat{w}^2\right] \\
&= \mathbb{E}_P\left[y^2(w + (\hat{w} - w))^2\right] \\
&= \mathbb{E}_P\left[y^2 w^2\right] + \mathbb{E}_P\left[y^2\left(2w\left(\hat{w} - w\right) + \left(\hat{w} - w\right)^2\right)\right] \\
&\leq \mathbb{E}_P\left[y^2 w^2\right] + \mathbb{E}_P\left[y^2\left(\frac{4w}{\sqrt{g''(1)}}\sqrt{\text{reg}(\hat{\eta};\mathcal{D},\lambda)} + \frac{4}{g''(1)}\text{reg}(\hat{\eta};\mathcal{D},\lambda)\right)\right] \\
&= \mathbb{E}_P\left[y^2 w^2\right] + \frac{4}{\sqrt{g''(1)}}\sqrt{\text{reg}(\hat{\eta};\mathcal{D},\lambda)}\mathbb{E}_P\left[y^2\left(w + \frac{1}{\sqrt{g''(1)}}\sqrt{\text{reg}(\hat{\eta};\mathcal{D},\lambda)}\right)\right]
\end{aligned}$$

□

Proposition 4.3 (Consistency)**.** *Under Assumptions A1-A5, and bounded outcomes y, the permutation weighting dose-response estimator is consistent, i.e., as $n \longrightarrow \infty$, $\mathbb{E}_P[y\hat{w}] \longrightarrow \mathbb{E}_Q[y]$.*

Proof. From Propositions 4.1 and 4.2, by minimizing an appropriate classifier loss, we can bound the bias and variance in terms of the classifier regret. Thus, under Assumption A5, and given bounded y, the bias and variance tend to 0 as $n \longrightarrow \infty$. □

Proposition 4.4 (Minimizing Imbalance)**.** *The L_p functional discrepancy between the observed data drawn from $p(\mathbf{a}, \mathbf{x})$ and the proposed distribution $q(\mathbf{a}, \mathbf{x})$ under permutation weighting is*

$$\begin{aligned}&\left\|\mathbb{E}_{p(\mathbf{a},\mathbf{x})}\left[\phi(\mathbf{a}_i) \otimes \psi(\mathbf{x}_i)(\hat{w}(\mathbf{a}_i, \mathbf{x}_i) - w(\mathbf{a}_i, \mathbf{x}_i))\right]\right\|_p \\ &\leq \frac{2}{\sqrt{g''(1)}}\kappa_r \left\|\mathbb{E}_{p(\mathbf{a},\mathbf{x})}\left[\phi(\mathbf{a}_i) \otimes \psi(\mathbf{x}_i)\right]\right\|_p\end{aligned}$$

where $p \geq 0$ and $\kappa_r = \sqrt{\text{reg}(\hat{\eta}; \mathcal{D}, \lambda)}$.

Proof. We will focus on the case of attaining balance between a source distribution, P and a target distribution Q defined over a common support. Define $w(a_i, x_i) = \frac{q(a_i,x_i)}{p(a_i,x_i)}$ and $\hat{w}(a_i, x_i)$ be an estimate of w.

$$\begin{aligned}&\left\|\mathbb{E}_P\left[\phi(a_i) \otimes \psi(x_i)\hat{w}(a_i, x_i)\right] - \mathbb{E}_Q\left[\phi(a_i) \otimes \psi(x_i)\right]\right\|_p \\ &= \left\|\mathbb{E}_P\left[\phi(a_i) \otimes \psi(x_i)\hat{w}(a_i, x_i)\right] - \mathbb{E}_P\left[\phi(a_i) \otimes \psi(x_i)\frac{q(a_i, x_i)}{p(a_i, x_i)}\right]\right\|_p \\ &= \left\|\sum_i^N \phi(a_i) \otimes \psi(x_i)\hat{w}(a_i, x_i)p(a_i, x_i) - \sum_i^N \phi(a_i) \otimes \psi(x_i)\frac{q(a_i, x_i)}{p(a_i, x_i)}p(a_i, x_i)\right\|_p \\ &= \left\|\sum_i^N \phi(a_i) \otimes \psi(x_i)p(a_i, x_i)\hat{w}(a_i, x_i) - \phi(a_i) \otimes \psi(x_i)q(a_i, x_i)\right\|_p \\ &= \left\|\sum_i^N \phi(a_i) \otimes \psi(x_i)p(a_i, x_i)(w(a_i, x_i) + (\hat{w}(a_i, x_i) - w(a_i, x_i))) - \phi(a_i) \otimes \psi(x_i)q(a_i, x_i)\right\|_p \\ &= \left\|\sum_i^N \phi(a_i) \otimes \psi(x_i)(p(a_i, x_i)w(a_i, x_i) + p(a_i, x_i)(\hat{w}(a_i, x_i) - w(a_i, x_i))) - \phi(a_i) \otimes \psi(x_i)q(a_i, x_i)\right\|_p \\ &= \left\|\sum_i^N \phi(a_i) \otimes \psi(x_i)q(a_i, x_i) + \phi(a_i) \otimes \psi(x_i)p(a_i, x_i)(\hat{w}(a_i, x_i) - w(a_i, x_i)) - \phi(a_i) \otimes \psi(x_i)q(a_i, x_i)\right\|_p \\ &= \left\|\sum_i^N \phi(a_i) \otimes \psi(x_i)p(a_i, x_i)(\hat{w}(a_i, x_i) - w(a_i, x_i))\right\|_p \\ &= \left\|\mathbb{E}_P\left[\phi(a_i) \otimes \psi(x_i)(\hat{w}(a_i, x_i) - w(a_i, x_i))\right]\right\|_p\end{aligned}$$

Then, by Lemma A.1, the result follows. □

Finally, we show minimization of imbalance through averaging of weights over multiple permutations.

Corrollary A.1. *With a single permutation as described above, as $n \to \infty$, and assuming the classifier is consistent, the functional imbalance is minimized.*

Proof. The weight vector that minimizes

$$\left\|\mathbb{E}_P\left[\phi(\mathbf{a}) \otimes \psi(\mathbf{x})w\right] - \mathbb{E}_P\left[\phi(\mathbf{a})\right] \otimes \mathbb{E}_P\left[\psi(\mathbf{x})\right]\right\|_p,$$

is the density ratio w. By definition, a single permutation $(\mathbf{a}_{\boldsymbol{\pi}}, \mathbf{x})$ is drawn from the product distribution $Q(\mathbf{a}, \mathbf{x}) = P(\mathbf{a})P(\mathbf{x})$. Therefore, the trained classifier $\hat{\eta}(\mathbf{a}, \mathbf{x})$ is estimating the probability of $(\mathbf{a}, \mathbf{x}) \sim Q$ instead of $(\mathbf{a}, \mathbf{x}) \sim P$. As $n \to \infty$ we have that the error of the classifier probability $\hat{\eta}$ tends to zero by assumption. Then, by proposition 6, we have that $\hat{w}(\mathbf{a}_{\boldsymbol{\pi}}, \mathbf{x}) \to w$. □

Corrollary A.2. *For fixed n, as the number of permutations increases, the functional imbalance is minimized.*

Proof. For each $j = 1, \ldots, B$, where B is the number of permutations, consider the empirical imbalance minimization problem:

$$\min_{\hat{w}_j} \left\| \sum_{i=1}^{n} \phi(\mathbf{a}_i) \otimes \psi(\mathbf{x}_i) \hat{w}_j(\mathbf{a}_i, \mathbf{x}_i) - \phi(\mathbf{a}_{\boldsymbol{\pi}_j(i)}) \otimes \psi(\mathbf{x}_i) \right\|_p .$$

From proposition 4.4, we know that each imbalance is minimized by minimizing the error of the classifier trained in permutation weighting. Since each permutation π_j is drawn from the $n-$sample product distribution $Q_n(\mathbf{a}, \mathbf{x}) = P_n(\mathbf{a})P_n(\mathbf{x})$, as $B \to \infty$, the empirical distribution of $\hat{w}_j$ tends to the distribution of weights computed under this distribution. Therefore, the empirical average of weight vectors tends to the weight vector that minimizes

$$\min_{\hat{w}} \int_{(\mathbf{a}', \mathbf{x}') \sim Q_n} \left\| \sum_{i=1}^{n} \phi(\mathbf{a}_i) \otimes \psi(\mathbf{x}_i) \hat{w}(\mathbf{a}_i, \mathbf{x}_i) - \phi(\mathbf{a}') \otimes \psi(\mathbf{x}') \right\|_p .$$

□

B. Connections to Existing Weighting Estimators

We now examine the connection between permutation and a number of covariate balancing estimators in the literature. In what follows, we first revisit the relationship with stabiized inverse propensity score weighting, look at score based estimates, i.e. estimators which can be estimated using the generalized method of moments with a particular focus on the covariate balancing propensity score (Imai and Ratkovic, 2014), then examine margin based estimators and their kernel extension, establishing an equivalence between permutation weighting and kernel mean matching (Huang et al., 2007), kernel balancing (Hazlett, 2016), and Kernel based covariate functional balancing (Wong and Chan, 2017) and relate permutation to stabilized balancing weights (Zubizarreta, 2015).

B.1. Stabilized Inverse Propensity Score

Perhaps the most immediately evident equivalence is to the stabilized inverse propensity score (IPSW) (Robins, 1997). This relationship is addressed in the main text; it is included in this section for the benefit of completeness. Recall the definition of the stabilized propensity score weight is simply the marginal density of treatment divide by the conditional density of treatment given covariates, i.e., $\frac{p(a)}{p(a|\mathbf{x})}$. Employing simply algebra we see that this quantity will be equivalent to permutation weights, i.e. $\frac{p(a)p(\mathbf{x})}{p(a,\mathbf{x})}$, under correct specification of the conditional density. The crucial difference between permutation weighting and IPSW comes under mis-specification. Permutation weighting will still seek balance with respect to the conditions implied by the classifier. IPSW, on the other hand, may fail to seek balance under mis-specification. This can result in substantial bias, as we have seen in the empirical results of the main text.

B.2. Covariate Balancing Propensity Scores

We first note the score condition of the covariate balancing propensity score (Imai and Ratkovic, 2014):

$$0 = \sum_{i}^{N} \frac{a_i \mathbf{x}_i}{(1 + \exp(-\theta_{\mathbf{x}_i} \mathbf{x}_i))} - \sum_{i}^{N} \frac{(1 - a_i)\mathbf{x}_i}{(1 + \exp(\theta_{\mathbf{x}_i} \mathbf{x}_i))} \tag{5}$$

The score condition for PW with logistic loss provides a simple comparison:

$$0 = \sum_i^N \left(\frac{C_i a_i \tilde{\mathbf{x}}_i}{(1+\exp(\theta_{a,\mathbf{x}} a_i \mathbf{x}_i + \theta_{\mathbf{x}_i} \mathbf{x}_i))} - \frac{(1-C_i) a_i \tilde{\mathbf{x}}_i}{(1+\exp(-\theta_{a,\mathbf{x}} a_i \mathbf{x}_i - \theta_{\mathbf{x}_i} \mathbf{x}_i))} \right) + \left(\frac{C_i (1-a) \tilde{\mathbf{x}}_i}{\exp(1+\theta_{\mathbf{x}_i} \mathbf{x}_i))} - \frac{(1-C_i)(1-a_i) \tilde{\mathbf{x}}_i}{(1+\exp(-\theta_{\mathbf{x}_i} \mathbf{x}_i))} \right)$$

Here we can see that both estimators are explicitly minimizing a balance condition, PW to the product, i.e. independent, distribution and CBPS between classes. In large samples these are equivalent conditions. However, an interesting interpretation emerges when we consider what happens in smaller samples. Permutation weighting will attempt to match the level of balance that exists in the empirical sample, which provides a data dependent regularization. This regularization likely explains the improvement of PW over CBPS in the case of the synthetic experiments involving mis-specification.

We note that a similar derivation yields the following for inverse propensity score weighting:

$$0 = \sum_i^N \frac{a_i \mathbf{x}_i}{(1+\exp(\theta_{\mathbf{x}_i} \mathbf{x}_i))} - \sum_i^N \frac{(1-a_i) \mathbf{x}_i}{(1+\exp(-\theta_{\mathbf{x}_i} \mathbf{x}_i))} \tag{6}$$

Here we can see that balance is *not* directly optimized for, which explains much of the poor performance of inverse propensity score weighting in the synthetic experiments with a misspecified estimator.

B.3. Stabilized Balancing Weights, Kernel Mean Matching, and Kernel Balancing

We will now briefly introduce MMD, weighting methods predicated on MMD, e.g. (Huang et al., 2007; Gretton et al., 2009), followed by a discussion of their connection to stabilized balancing weights (Zubizarreta, 2015).

The maximum mean discrepancy (MMD) (Gretton et al., 2012), is a two-sample test that distinguishes between two candidate distributions by finding the maximum mean distance between the means of the two samples after transformation, i.e.,

$$\sup_{f \in \mathcal{F}} \left(\mathbb{E}_{a \sim A} \left[f(a) \right] - \mathbb{E}_{b \sim B} \left[f(b) \right] \right) \tag{7}$$

When $\mathcal{F}$ is a reproducing kernel Hilbert space this can be estimated as the squared difference of their means in feature space. Letting $\phi(\cdot)$ be a kernel associated with a random variable A and $\psi(\cdot)$ be the kernel associated with random variable B, the finite sample estimate of equation 7 is given as

$$\text{MMD}(A, B) = \left\| \frac{1}{N} \sum_i^N \phi(a_i) - \frac{1}{M} \sum_j^M \psi(b_j) \right\|^2$$

with N and M being the size of the samples drawn from A and B, respectively. The value of MMD reflects the maximum distance between these means There are a couple of points worth noting. First, if the kernel being used obeys certain properties, i.e is characteristic (Sriperumbudur et al., 2008), the MMD is able to differentiate between two exponential-family distributions on an arbitrary number of moments (Gretton et al., 2012). Second, when a linear kernel is employed this is value is simply the squared difference in means between the two groups.

MMD has been used throughout the literature as an objective for minimizing imbalance. Within the context of domain adaptation, (Huang et al., 2007) introduce kernel mean matching (KMM) which defines an optimization procedure that seeks to find a set of weights such that the distance between the target and source distribution is minimized, specifically

$$\min_{\beta} \left\| \frac{1}{N} \sum_{i}^{N} \beta(a_i)\phi(a_i) - \frac{1}{M} \sum_{j} \psi(b_j) \right\|^2 \tag{8}$$
$$\text{Such that } \beta(a) > 0, \sum_{i}^{N} \beta(a_i) = 1$$

This procedure was later rediscovered for the task of balancing weights by (Hazlett, 2016). Somewhat surprisingly, the connection to permutation weighting can be easily obtained via results currently found in the literature. (Reid and Williamson, 2011) relate a pessimistic MMD to the support vector machine (SVM), seeking to maximize the MMD by solving the SVM problem

$$\min_{\alpha} \sum_{i}^{M} \sum_{j}^{M} \alpha_i \alpha_j c_i c_j k(x_i, x_j)$$
$$\text{Such that } \alpha \geq 0$$
$$\sum_{i}^{m} \alpha_i y_i = \frac{m^+ - m^-}{m}$$
$$\sum_{i}^{m} \alpha_i = 1$$

where c indicates which dataset a sample has been drawn from and is coded $\{-1, 1\}$. (Bickel et al., 2007) (Section 8) shows that solving the KMM objective is equivalent to solving the above SVM problem under the additional modification that the values of α for are fixed to a constant for one class, producing a Rocchio-style approximation (Joachims, 1997) to the SVM. In both cases the final weighting is given directly by taking the value of the dual weights (α).

The aforementioned classifiers may be employed within the context of permutation weighting by considering the two samples to be the observed data with a target distribution of the resampled data (as we have assumed throughout). In order to use the dual weights directly the bootstrap procedure is replaced by an average over permutations. Alternatively the weight function may be used directly by considering $\exp(w(\mathbf{x_i}, a_i)$ as in (Bickel et al., 2007). The benefit of the latter approach is the ability to use cross validation for setting the hyper-parameters of the classifiers. Asymptotically, as the independence property is obeyed by the resampled data, permutation weighting and these procedures are equivalent in the binary treatment setting. To see why this is the case, consider the explicit form of permutation weighting under MMD loss:

$$\sup_{f \in \mathcal{F}} \left(\mathbb{E}_{a,x \sim p(A)p(\mathbf{X})} \left[f(a,x) \right] - \mathbb{E}_{a,x \sim p(A,\mathbf{X})} \left[f(a,x) \right] \right)$$

where we have again assumed that $\mathcal{F}$ is a reproducing kernel Hilbert space. This is precisely the Hilbert-Schmidt independence criterion (Gretton et al., 2005), which was shown by (Song, 2008) to be equivalent to the maximum mean discrepancy between the two $\mathbf{X}$ samples associated with treatment and control, respectively, when A is binary. In the non-binary case the permutation weighting setup defines a novel kernel balancing estimator for general treatments.

Finally, we examine the relationship between permutation weighting and the stable balancing weights of Zubizarreta (2015). Zubizarreta (2015) defined the following quadratic program to infer what he refers to as stable balancing weights:

$$\min_{w} \|\mathbf{w} - \overline{\mathbf{w}}\|_2^2$$
Subject to
$$\left|\mathbf{w}^T X_{\text{control}_p} - \overline{\mathbf{X}}_{\text{test}_p}\right| < \delta, p = 1, \ldots, k$$
$$\sum w = 1$$
$$w \geq 0$$

Intuitively this attempts to minimize the variance of the weights subject to constraints on marginal balance conditions. Comparing this to the kernel mean matching problem (equation 8), we see that stable balancing weights emphasize uniform weights subject to a constraint of predetermined levels of marginal balance. Kernel mean matching on the other hand, seeks to minimize the maximum discrepancy between the two distributions. Minimizing the discrepancy rather than setting it to a fixed level is that it removes a large amount of possible human induced error in the form of additional hyperparameters. While the MMD approach does not have an explicit mechanism to reduce variance, an approximation can be applied by solving the SVM problem using a ν-SVM (Scholkopf and Smola, 2001) which imposes an additional constraints that limits the size of individual weights.

C. Continuous Kang-Schafer Data Generating Process

The same basic setup can be extended to the continuous treatment case by replacing the Bernoulli treatment assignment with a continuous analogue. For the following simulations, we simulate treatment dosage as a linear function as in (Kang and Schafer, 2007), but adding in standard normal noise. Finally, the dosage enters the outcome model through a logit function to introduce a small non-linearity in response to dose.

$$\begin{aligned} X_k &= \mathcal{N}(0,1) \quad \forall k \in \{1,2,3,4\} \\ \epsilon &\sim \mathcal{N}(0,1) \\ A \mid X &= X_1 - 0.5X_2 + 0.25X_3 + 0.1X_4 + \epsilon \\ \mathbb{E}[Y \mid A, X] &= 210 + \text{logit}(A) + 27.4X_1 + 13.7X_2 + 13.7X_3 + 13.7X_4 \\ Y \mid A, X &= \mathcal{N}(\mathbb{E}[Y \mid A, X], 1) \end{aligned}$$

Misspecification is handled identically to the binary case.

Figure 4 shows the results when treatment is a linear function of the observed covariates. Propensity score weighting does quite well in terms of reducing bias (consistently with the lowest amount of bias out of all methods), but permutation weighting (particularly using a boosted model) does a better job of trading off bias and variance so as to reduce IRMSE (outperforming the normal-linear IPSW model by around 15% at $n = 2000$, for instance). Thus, even when the propensity score model is well specified, boosting is able to outperform it by more smartly regularizing (and, thereby, reducing variance). The level of regularization may be tuned rigorously using cross-validation.

In the case of misspecification, figure 5 shows the learning curves of the various methods. Permutation weighting outperforms all methods at all examined sample sizes in both bias and accuracy. While PW with a logistic classifier has very similar levels of bias as does (Fong et al., 2018), it does so with sufficiently lower variance that even at $n = 500$ it reduces IRMSE by around 15%. The boosting model improves upon the linear model both in terms of bias and IRMSE; at $n = 2000$, boosting provides estimates with around 40% lower IRMSE than does npCBPS. A useful point of comparison is that permutation weighting outperforms the current state of the art by around four times as much as the state of the art improves on no weighting whatsoever.

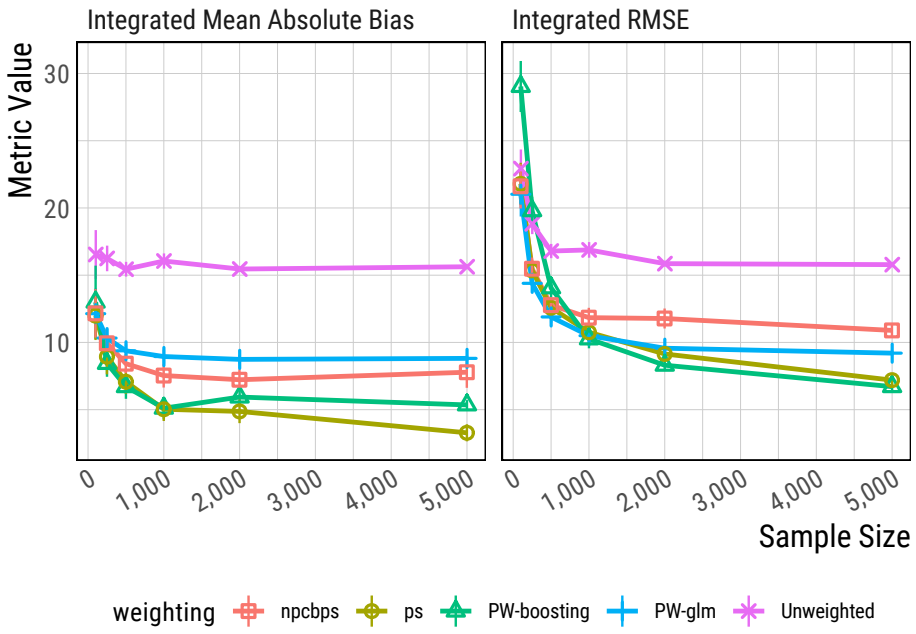

Figure 4: Continuous Kang-Schafer simulation under correct specification of confounding variables. `npcbps` is non-parametric covariate balancing propensity scores (Fong et al., 2018). `ps` is a normal-linear regression based propensity score model. `PW-boosting` is a permutation weighting model using a gradient boosted decision tree. `PW-glm` is a permutation weighting model using a logistic regression. `Unweighted` uses no weighting.

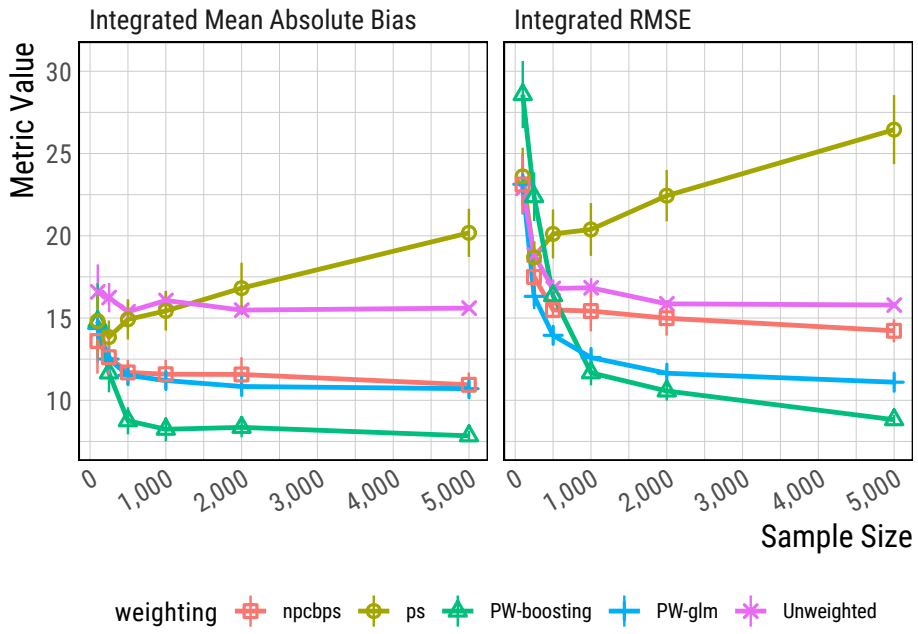

Figure 5: Continuous Kang-Schafer simulation under misspecification of confounding variables. `npcbps` is non-parametric covariate balancing propensity scores (Fong et al., 2018). `ps` is a normal-linear regression based propensity score model. `PW-boosting` is a permutation weighting model using a gradient boosted decision tree. `PW-glm` is a permutation weighting model using a logistic regression. `Unweighted` uses no weighting.

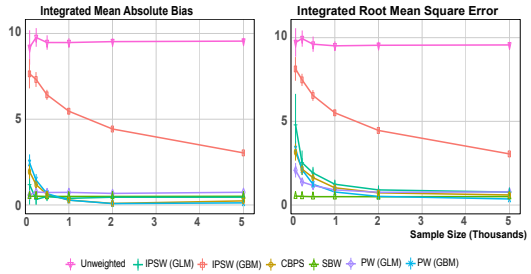

Figure 6: Kang-Schafer simulation under correct specification of confounding variables. `Unweighted` uses no weighting. `IPSW (GLM)` is a logistic-regression-based propensity score model. `IPSW (GBM)` is a propensity score model trained with a gradient boosted decision tree. `CBPS` is covariate balancing propensity scores (Imai and Ratkovic, 2014). `SBW` is stable balancing weights (Zubizarreta, 2015). `PW (GLM)` is a permutation weighting model using a logistic regression. `PW (GBM)` is a permutation weighting model using a gradient boosted decision tree.

D. Extended Kang and Schafer (2007) simulation results

The data is simulated according to the following:

$$\begin{aligned} X_k &= \mathcal{N}(0,1) \quad \forall k \in \{1,2,3,4\} \\ p(A=1 \mid X) &= \text{logit}^{-1}\{X_1 - 0.5X_2 + 0.25X_3 + 0.1X_4\} \\ \mathbb{E}[Y \mid A, X] &= 210 + A + 27.4X_1 + 13.7X_2 + 13.7X_3 + 13.7X_4 \\ Y \mid A, X &= \mathcal{N}(\mathbb{E}[Y \mid A, X], 1) \end{aligned}$$

Under the non-linear "misspecification", the covariates are not observed directly, but instead only the following transformations:

$$X_1 = \exp\left\{\frac{X_1}{2}\right\} \qquad X_2 = \frac{X_2}{1+\exp\{X_1\}} + 10$$

$$X_3 = \left(\frac{X_1 X_3}{25} + .6\right)^3 \qquad X_4 = (X_2 + X_4 + 20)^2$$

Results from simulations based on the correctly specified data generating process are shown in figure 6. The correct treatment and outcome models are linear in this simulation. As such, the propensity score model is specified correctly and therefore performs well. Stable balancing weights also perform very strongly in this case (particularly in low sample sizes), as they both explicitly reduce the variance of the weights and correctly specify the form of the confounding relationship. These results show that, given sufficient sample size, PW with a logistic regression classifier (or boosting) replicates and eventually outperforms stable balancing weights, even when the true relationships are linear. Note, however, that bias under the boosted model is driven to the minimum very quickly, even though the more complicated model increases the variance (hurting the overall IRMSE). PW with logistic regression typically outperforms CBPS in this simulation, particularly at low sample sizes, despite the fact that both seek to balance the correct specification of confounding. This difference in performance comes from the minor data-dependent regularization in the score condition estimated by permutation weighting (details in appendix B.2). Machine learning the propensity score performs consistently poorly relative to all other methods examined here, as it imposes no structure on the data. This demonstrates the importance of seeking balance rather than seeking correct specification. The latter is very hard; while the former is achievable.

Figure 7 shows the results of the binary model when the covariates are misspecified. In this figure, only the best performing methods are shown.

The following tables show IRMSE and Bias estimates (defined identically as in the main text) for all combinations of outcome estimation strategies (weighting only, direct method and doubly-robust), models (OLS and random forests) and weighting methods. Estimates of bias or IRMSE are followed by the standard error (estimated via non-parametric bootstrap).

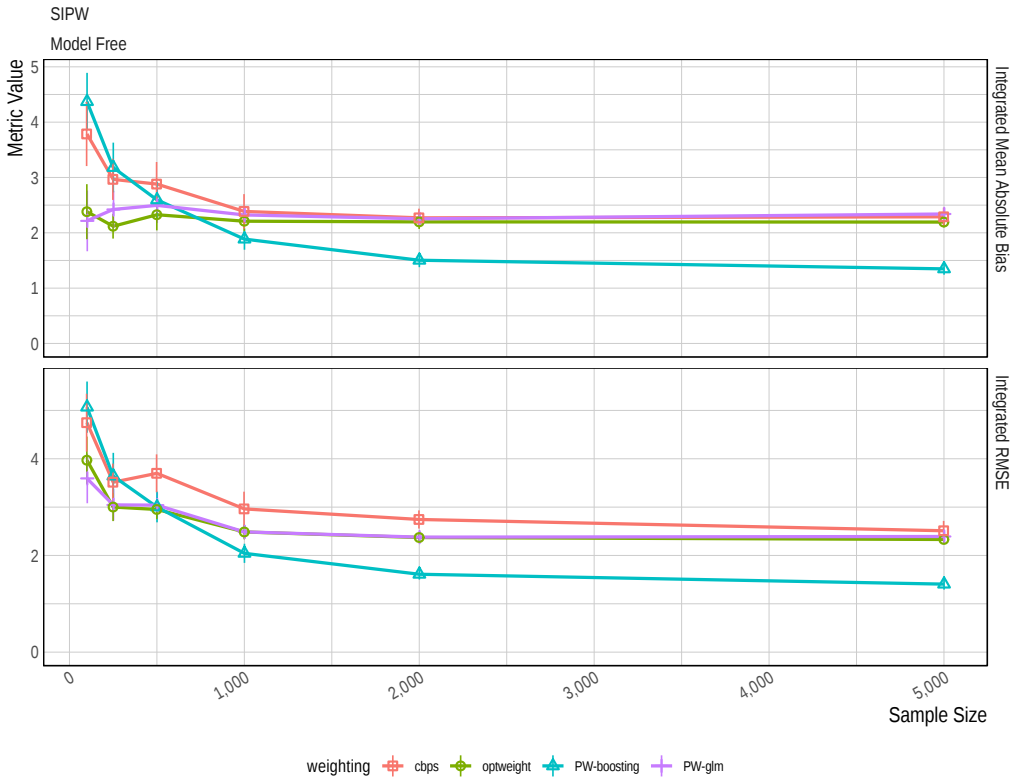
SIPW
Model Free
Metric Value
Integrated Mean Absolute Bias
Integrated RMSE
Sample Size
0
1,000
2,000
3,000
4,000
5,000
weighting
cbps
optweight
PW-boosting
PW-glm

Figure 7

D.1. Binary treatment – Kang and Schafer (2007)

Well specified
N = 2000

	Metric	Unweighted	Logit	Boosting	CBPS	SBW	PW (Logit)	PW (Boosting)
IPSW								
Model Free	Bias	9.52 ± 0.10	0.45 ± 0.07	4.42 ± 0.06	0.08 ± 0.04	0.50 ± 0.00	0.67 ± 0.03	0.08 ± 0.05
	IRMSE	9.55 ± 0.10	0.92 ± 0.06	4.46 ± 0.06	0.74 ± 0.03	0.50 ± 0.00	0.77 ± 0.03	0.52 ± 0.06
Direct Method								
OLS	Bias	0.50 ± 0.00	0.50 ± 0.00	0.49 ± 0.00	0.50 ± 0.00	0.50 ± 0.00	0.50 ± 0.00	0.50 ± 0.00
	IRMSE	0.50 ± 0.00	0.50 ± 0.00	0.49 ± 0.00	0.50 ± 0.00	0.50 ± 0.00	0.50 ± 0.00	0.50 ± 0.00
Random Forest	Bias	0.42 ± 0.01	0.03 ± 0.01	0.13 ± 0.01	0.06 ± 0.02	0.06 ± 0.00	0.03 ± 0.01	0.03 ± 0.00
	IRMSE	0.44 ± 0.02	0.09 ± 0.01	0.15 ± 0.01	0.18 ± 0.02	0.16 ± 0.01	0.09 ± 0.01	0.08 ± 0.01
Doubly Robust								
OLS	Bias	0.50 ± 0.00	0.50 ± 0.00	0.49 ± 0.00	0.50 ± 0.00	0.50 ± 0.00	0.50 ± 0.00	0.50 ± 0.00
	IRMSE	0.50 ± 0.00	0.50 ± 0.00	0.49 ± 0.00	0.50 ± 0.00	0.50 ± 0.00	0.50 ± 0.00	0.50 ± 0.00
Random Forest	Bias	0.47 ± 0.02	0.18 ± 0.01	0.26 ± 0.01	0.18 ± 0.01	0.03 ± 0.01	0.19 ± 0.01	0.18 ± 0.01
	IRMSE	0.50 ± 0.02	0.22 ± 0.01	0.28 ± 0.02	0.27 ± 0.02	0.16 ± 0.01	0.22 ± 0.01	0.21 ± 0.01

Misspecified
N = 2000

	Metric	Unweighted	Logit	Boosting	CBPS	SBW	PW (Logit)	PW (Boosting)
IPSW								
Model Free	Bias	9.66 ± 0.12	5.87 ± 0.71	4.72 ± 0.06	2.28 ± 0.08	2.22 ± 0.06	2.27 ± 0.06	1.52 ± 0.05
	IRMSE	9.68 ± 0.13	8.04 ± 1.27	4.75 ± 0.07	2.74 ± 0.08	2.38 ± 0.05	2.38 ± 0.07	1.62 ± 0.06
Direct Method								
OLS	Bias	2.78 ± 0.05	2.66 ± 0.08	1.04 ± 0.03	2.69 ± 0.09	2.22 ± 0.06	2.14 ± 0.05	1.13 ± 0.05
	IRMSE	2.79 ± 0.06	2.71 ± 0.08	1.11 ± 0.04	2.75 ± 0.10	2.38 ± 0.05	2.19 ± 0.06	1.19 ± 0.05
Random Forest	Bias	0.78 ± 0.02	0.25 ± 0.01	0.37 ± 0.01	0.29 ± 0.01	0.23 ± 0.02	0.24 ± 0.01	0.20 ± 0.01
	IRMSE	0.79 ± 0.02	0.27 ± 0.01	0.39 ± 0.02	0.33 ± 0.01	0.30 ± 0.01	0.27 ± 0.01	0.23 ± 0.01
Doubly Robust								
OLS	Bias	2.78 ± 0.05	2.66 ± 0.08	1.04 ± 0.03	2.69 ± 0.09	2.22 ± 0.06	2.14 ± 0.05	1.13 ± 0.05
	IRMSE	2.79 ± 0.06	2.71 ± 0.08	1.11 ± 0.04	2.75 ± 0.10	2.38 ± 0.05	2.19 ± 0.06	1.19 ± 0.05
Random Forest	Bias	0.90 ± 0.02	0.53 ± 0.02	0.60 ± 0.02	0.54 ± 0.02	0.39 ± 0.02	0.49 ± 0.02	0.48 ± 0.02
	IRMSE	0.92 ± 0.02	0.54 ± 0.02	0.61 ± 0.02	0.56 ± 0.02	0.44 ± 0.02	0.51 ± 0.02	0.49 ± 0.02

D.2. Continuous treatment – Kang and Schafer (2007)

Well specified
N = 2000

model	Metric	Unweighted	Normal-Linear	npCBPS	PW (Logit)	PW (Boosting)
IPSW						
Model Free	Bias	15.750 ± 0.114	4.113 ± 0.285	7.996 ± 0.377	8.862 ± 0.116	6.021 ± 0.149
	IRMSE	16.142 ± 0.114	8.685 ± 0.242	11.445 ± 0.191	9.697 ± 0.119	7.576 ± 0.143
Direct Method						
OLS	Bias	0.269 ± 0.000	0.269 ± 0.000	0.269 ± 0.000	0.269 ± 0.000	0.269 ± 0.000
	IRMSE	0.269 ± 0.000	0.284 ± 0.003	0.298 ± 0.009	0.270 ± 0.000	0.273 ± 0.001
Random Forest	Bias	2.507 ± 0.031	1.171 ± 0.019	1.444 ± 0.026	1.302 ± 0.014	1.280 ± 0.016
	IRMSE	2.574 ± 0.030	1.229 ± 0.021	1.499 ± 0.028	1.339 ± 0.015	1.321 ± 0.017
Doubly Robust						
OLS	Bias	0.249 ± 0.001	0.255 ± 0.003	0.259 ± 0.002	0.260 ± 0.001	0.257 ± 0.002
	IRMSE	0.272 ± 0.002	0.354 ± 0.008	0.372 ± 0.017	0.289 ± 0.001	0.304 ± 0.003
Random Forest	Bias	2.620 ± 0.033	1.423 ± 0.021	1.719 ± 0.027	1.603 ± 0.015	1.560 ± 0.017
	IRMSE	2.711 ± 0.031	1.500 ± 0.025	1.797 ± 0.029	1.663 ± 0.017	1.624 ± 0.019

Misspecified
N = 2000

model	Metric	Unweighted	Normal-Linear	npCBPS	PW (Logit)	PW (Boosting)
IPSW						
Model Free	Bias	15.549 ± 0.130	16.810 ± 0.526	10.821 ± 0.259	10.810 ± 0.126	8.406 ± 0.141
	IRMSE	16.002 ± 0.115	23.581 ± 0.881	14.747 ± 0.315	11.637 ± 0.143	9.418 ± 0.150
Direct Method						
OLS	Bias	0.269 ± 0.000	1.351 ± 0.676	0.551 ± 0.195	2.539 ± 0.043	1.276 ± 0.053
	IRMSE	0.269 ± 0.000	4.436 ± 2.533	1.687 ± 0.178	2.549 ± 0.049	1.323 ± 0.051
Random Forest	Bias	3.221 ± 0.034	6.865 ± 3.195	2.601 ± 0.051	2.544 ± 0.036	2.725 ± 0.056
	IRMSE	3.273 ± 0.032	24.459 ± 13.162	2.685 ± 0.056	2.600 ± 0.038	2.811 ± 0.059
Doubly Robust						
OLS	Bias	2.450 ± 0.050	1.709 ± 0.461	1.326 ± 0.105	2.382 ± 0.044	1.306 ± 0.059
	IRMSE	2.879 ± 0.061	6.012 ± 1.929	3.922 ± 0.211	2.692 ± 0.045	2.285 ± 0.047
Random Forest	Bias	3.427 ± 0.034	6.922 ± 3.121	2.896 ± 0.051	2.840 ± 0.037	3.027 ± 0.056
	IRMSE	3.503 ± 0.033	24.528 ± 13.104	2.994 ± 0.056	2.918 ± 0.039	3.130 ± 0.059

D.3. In-sample versus causal error

E. ROC Curve Interpretation

Figure 9 shows a well-tuned GBDT, a poorly tuned GBDT and a logistic regression's receiver-operating curves. For this linear data generating process, a poorly tuned GBDT is inadmissable no matter which proper scoring rule is used. On the other hand, either the linear model or the well-tuned GBDT would be an acceptable choice depending on the choice of scoring rule. Practitioners in this case would likely prefer to use the more parsimonious and interpretable linear model.

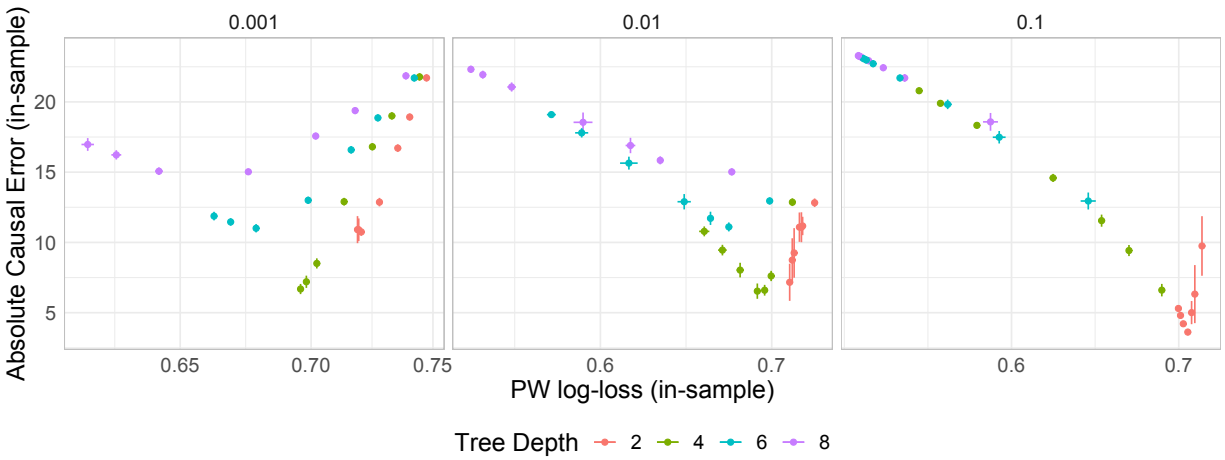

Figure 8: The estimated GBDT classifier error in-sample does not correlate with the error of the causal estimate over a grid of hyperparameter values. The y-axis represents in-sample causal error, while the x-axis is the in-sample PW loss (i.e. in-sample imbalance). Hyperparameters tuned were the tree-depth of each decision tree (color), the learning rate, ν (columns) and the number of trees (not annotated, from 100 to 5000).

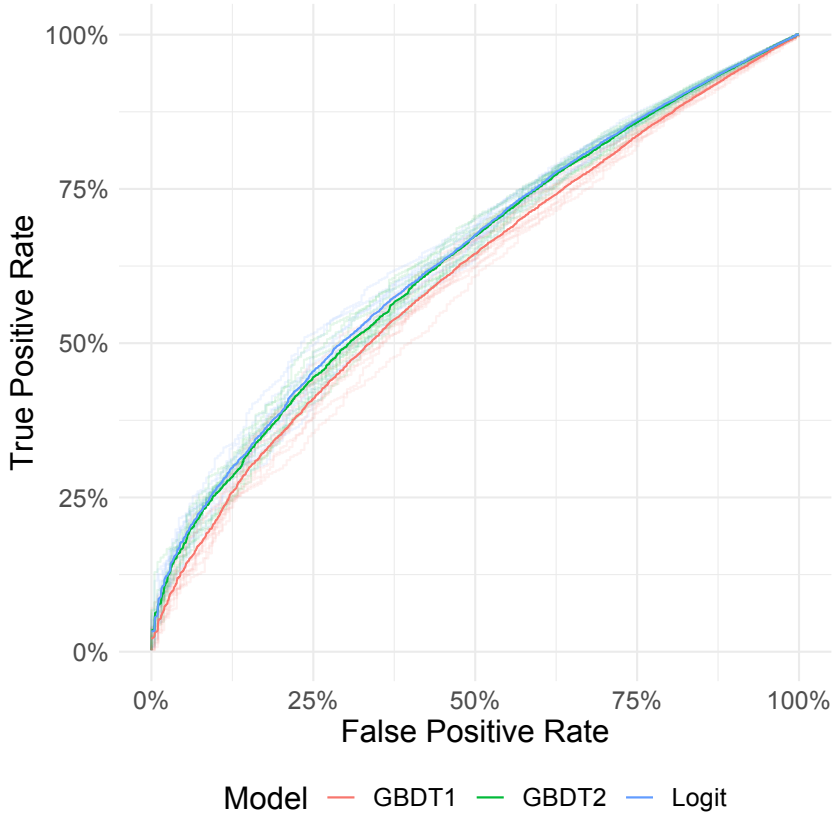

Figure 9: A ROC curve demonstrating that the linear model slightly out-performs the well-tuned GBDT across all false-positive rates. These two models have generally very similar performance, while the poorly tuned GBDT is substantially worse. The solid lines are the average over 10-fold cross-validation (the light lines in the background are each of the constituent folds).

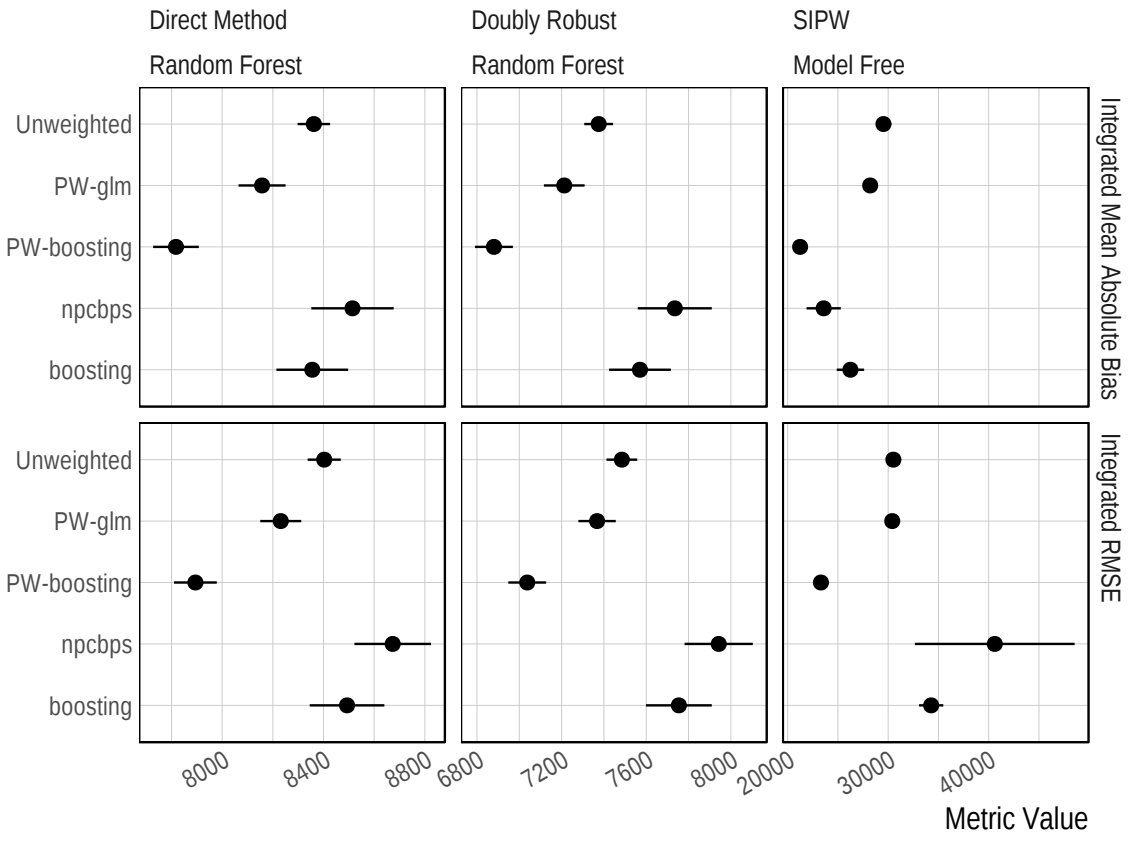

Figure 10: This figure shows additional results for the LaLonde simulation which incorporate Random Forest outcome models.

F. Extended LaLonde simulation results

$$
\begin{aligned}
\tilde{P} &= \text{Natural spline basis expansion of } \hat{g}(X) \\
\boldsymbol{\beta} &\approx [-0.502\ 0.132\ -0.079\ 0.887] \\
\mathbb{E}[A \mid X] &= 1 + \tilde{P}\beta + 0.01 * \text{age}^2 - .3\text{education}^2 - 0.01\log(\text{income74} + .01)^2 + 0.01\log(\text{income75} + 0.01)^2 \\
\mathbb{V}[A \mid X] &= 10^2 \\
\tilde{A} &= \text{Natural spline basis expansion of } A \\
\boldsymbol{\gamma} &\approx [0.117\ 0.319\ -0.582\ 0.715] \\
\mathbb{E}[Y \mid A, X] &= \text{income76} + \tilde{A}\gamma + .1\exp\{.7\log(\text{income74} + 0.01) + .7\log(\text{income75} + 0.01)\} \\
\mathbb{V}[Y \mid A, X] &= 10^2
\end{aligned}
$$

Figure 10 provides additional results for the LaLonde data generating process. In these results, we show bias and accuracy for a weighting-only model (replicated from the main body), a direct-method model (incorporating case-weights generated by the various methods) and a (Kennedy et al., 2016) style double robust estimator. In these plots, a linear propensity score model is left off of the plots as it performs so much more poorly that the x-axis is greatly skewed.