LEARNING BALANCED CAUSAL REPRESENTATIONS VIA DIFFERENTIABLE PROPENSITY-SCORE SUBCLASSIFICATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Neural causal representation learning is increasingly used for average treatment effect (ATE) estimation. Many existing methods estimate propensity scores using binary cross-entropy loss, treating treatment-prediction accuracy as a proxy for adjustment quality. However, accurate propensity prediction from a learned representation does not in general ensure covariate balance or conditional independence between treatment assignment and covariates given the estimated score. We propose a plug-and-play subclassification-based balancing module that promotes covariate balance within propensity-score subclasses. The method incorporates subclassification into end-to-end neural training by jointly using a differentiable relaxation, dynamic programming to select subclass intervals, and iterative optimization. On the ACIC and IHDP benchmarks, our method reduces absolute ATE bias by approximately 0.035 and 0.062, respectively, compared with the strongest baseline. These results demonstrate that explicitly enforcing covariate balance can provide benefits beyond propensity-score prediction accuracy alone.

1 INTRODUCTION

In observational studies, estimating the average treatment effect (ATE) using binary treatment T, outcome Y and high-dimensional covariate X is one of the core problems in causal inference. When identification conditions such as no unmeasured confounding hold, existing methods can be roughly categorized into two lines of methods. The first class of methods directly learns the propensity score $e(X)$ and outcome regression $\mu_t(X)$ based on the raw covariates X, and then estimates the ATE via outcome regression, including IPW(Rosenbaum & Rubin (1983)), AIPW(Rosenbaum & Rubin (1984)), or modern frameworks such as cross-fitted DML(Chernozhukov et al. (2018)). The second class first learns a shared representation $Z = \psi(X)$ from observational data, then estimates the propensity score and outcome regression based on this representation, and finally completes causal effect estimation with an appropriate ATE estimator.

Compared with learning nuisance functions directly on raw high-dimensional covariates, shared representation learning can reduce the effective complexity of subsequent nuisance estimation by extracting structural information jointly underlying treatment assignment and outcome prediction, and facilitate modeling nonlinear relationships and treatment-effect heterogeneity. When the representation preserves key information relevant to causal adjustment, it may also yield better generalization under finite samples or certain distribution shifts.

Existing causal representation learning methods have explored this problem from various perspectives. TARNet (Shalit et al. (2017))models the two potential outcomes using a shared representation and treatment-specific outcome heads, enhancing the flexibility of outcome modeling under high-dimensional covariates and heterogeneous treatment effects. DragonNet(Shi et al. (2019)) further introduces a propensity-score head on this basis, emphasizing the role of treatment assignment modeling in learning representations that preserve confounding-related information. BCAUSS(Tesei et al. (2023)) additionally incorporates a global covariate-balance objective into causal representation learning to reduce overall covariate imbalance between the treated and control groups.

Nevertheless, existing methods based primarily on propensity-score prediction may still face a critical issue: propensity-score prediction accuracy is not equivalent to covariate balance. For the true

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

propensity score $e(X) = \mathbb{P}(T = 1 \mid X)$, its balancing-score property can be expressed as

$$T \perp\!\!\!\perp X \mid e(X).$$

When learning the propensity score $e(\psi(X)) = \mathbb{P}(T = 1 \mid \psi(X))$ based on the representation $\psi(X)$, the theoretically guaranteed balancing property applies primarily to $\psi(X)$, i.e.,

$$T \perp\!\!\!\perp \psi(X) \mid e(\psi(X)),$$

which does not necessarily ensure

$$T \perp\!\!\!\perp X \mid e(\psi(X)).$$

If the representation compresses or discards information relevant to confounding adjustment, adequate balance on raw covariates may not be achieved even with high propensity-score prediction accuracy, thereby degrading ATE estimation.

Several existing methods have attempted to address this limitation by incorporating global covariate-balance objectives. However, global balance metrics may mask local discrepancies across different propensity-score regions: achieving small overall imbalance does not guarantee favorable treated-control balance within every local propensity-score stratum. Consequently, reporting only global balance may fail to reveal the specific balancing mechanism of the learned representation and limit model interpretability and diagnosability.

Motivated by the above observations, this paper builds upon the idea of propensity-score subclassification(Rosenbaum & Rubin (1984)) and introduces local covariate balance objectives during shared representation and nuisance-function learning. The proposed method aims to make the propensity score estimated from the learned representation not only support treatment and outcome prediction but also achieve assessable and interpretable local covariate balance in the raw covariate space. Our method operates at the nuisance learning stage and can be further combined with ATE estimators such as AIPW.

Technically, we introduce a differentiable subclassification mechanism that incorporates propensity-score-based local covariate balance into shared representation and nuisance-function learning. By optimizing local balance in the original covariate space, the proposed framework enables the learned representation to preserve confounding-relevant information while providing interpretable balance diagnostics beyond global metrics. Extensive experiments on benchmark datasets, including ACIC and IHDP, demonstrate the effectiveness of the proposed method for treatment effect estimation. Further ablation studies validate the contribution of the proposed local balance module and optimization strategy.

2 Notation and Preliminary Definitions

Define sample size n. For each sample i:

- $X_i \in \mathbb{R}^p$: p-dimensional raw covariate vector;
- $T_i \in \{0, 1\}$: binary treatment assignment indicator;
- Y_i: observed outcome for unit i;
- $Y_i(0), Y_i(1)$: the potential outcomes under control ($T = 0$) and treatment ($T = 1$), respectively.

Define the true propensity score as $e(X) = \mathbb{P}(T = 1 \mid X)$. We use $\psi_{n,j} : \mathbb{R}^p \to \mathbb{R}^d$ to denote the neural network representation mapping, and $e(\psi_{n,j}(X))$ denotes the propensity score evaluated over the learned representation. The network-estimated propensity score is denoted $\hat{e}(\psi_{n,j}(X))$.

Given propensity-score subclassification with K_n disjoint blocks:

- The hard block indicator function for the j-th bin:

$$B_j(X) \triangleq \mathbf{1}\big\{e(\psi(X)) \in \text{Block } j\big\},$$

  where $B_j(X) = 1$ if the representation-induced propensity score falls within the j-th propensity block, and $B_j(X) = 0$ otherwise.

- $\Psi_{n,j}(X)$: smoothed, differentiable approximation of $B_j(X)$, acting as soft binning weight; both $B_j(X)$ and $\Psi_{n,j}(X)$ are scalar-valued functions of X.

During network training, whenever the argument of an ordinary absolute-value term is zero, its backpropagated gradient is set to zero. Equivalently, we use the convention

$$\frac{\mathrm{d}}{\mathrm{d}u}|u|\Big|_{u=0} = 0, \tag{1}$$

where this equality denotes the selected subgradient convention rather than a classical derivative. Additional notation:

- $\|\cdot\|$: standard Euclidean norm defined on $\mathbb{R}^p$;

3 Proposed General

This framework is compatible with a broad class of causal representation learning models equipped with a propensity estimation branch, without relying on a specific network backbone.

3.1 Differentiable Propensity Subclassification

The goal of propensity-score subclassification is to construct strata within which the covariate distributions of treated and control units are comparable. We measure the residual imbalance in each stratum by the discrepancy between the covariate mean vectors of the treatment and control groups.

The hard indicator provides an interpretable subclassification scheme, but it is not differentiable with respect to the representation and propensity-score network. We therefore introduce a differentiable approximation $\Psi_{n,j}(X_i)$, which assigns a nonnegative soft weight to observation i for stratum j.To state the hard and soft objectives in a unified form, let $W_j(X_i)$ denote a generic stratum weight, instantiated by $B_j(X_i)$ for hard subclassification and by $\Psi_{n,j}(X_i)$ for soft subclassification. For any collection of stratum weights $W_j(X_i)$, define the weighted covariate means and feature-wise variance vectors as

$$\hat{\mu}_{1j}(W_j) = \frac{\sum_{i=1}^n W_j(X_i)T_iX_i}{\sum_{i=1}^n W_j(X_i)T_i}, \qquad \hat{V}_{1j}(W_j) = \frac{\sum_{i=1}^n W_j(X_i)T_i\left(X_i - \hat{\mu}_{1j}(W_j)\right)^{\odot 2}}{\sum_{i=1}^n W_j(X_i)T_i},$$

$$\hat{\mu}_{0j}(W_j) = \frac{\sum_{i=1}^n W_j(X_i)(1-T_i)X_i}{\sum_{i=1}^n W_j(X_i)(1-T_i)}, \qquad \hat{V}_{0j}(W_j) = \frac{\sum_{i=1}^n W_j(X_i)(1-T_i)\left(X_i - \hat{\mu}_{0j}(W_j)\right)^{\odot 2}}{\sum_{i=1}^n W_j(X_i)(1-T_i)},$$

where $\odot 2$ denotes element-wise squaring. Define the aggregated standardized mean-difference discrepancy as

$$d_{\mathrm{SMD}}(\mu_1, \mu_0, V_1, V_0) = \left\| (\mu_1 - \mu_0) \oslash \sqrt{\frac{V_1 + V_0}{2}} \right\|_2,$$

where $\oslash$ denotes element-wise division. The hard subclassification balance loss is

$$L_B = \sum_{j=1}^{K_n} d_{\mathrm{SMD}}\left(\hat{\mu}_{1j}(B_j), \hat{\mu}_{0j}(B_j), \hat{V}_{1j}(B_j), \hat{V}_{0j}(B_j)\right),$$

whereas the differentiable soft balance loss is

$$L_\Psi = \sum_{j=1}^{K_n} d_{\mathrm{SMD}}\left(\hat{\mu}_{1j}(\Psi_{n,j}), \hat{\mu}_{0j}(\Psi_{n,j}), \hat{V}_{1j}(\Psi_{n,j}), \hat{V}_{0j}(\Psi_{n,j})\right).$$

The remaining challenge is how to determine differentiable subclass boundaries during training, which is addressed in the following section.

3.2 Dynamic Programming for Optimal Propensity Subclassification

With the network parameters fixed, we obtain the current propensity scores$\{\hat{e}_i\}_{i=1}^n$ and sort the observations in nondecreasing order,

$$\hat{e}_{(1)} \le \cdots \le \hat{e}_{(n)},$$

where $(X_{(k)}, T_{(k)})$ denotes the covariates and treatment indicator of the observation associated with $\hat{e}_{(k)}$. We restrict the subclassification to K_n contiguous intervals in this ordering. This restriction preserves the ordering induced by the propensity score while making the partition problem amenable to dynamic programming.

For $0 \le \ell < r \le n$, let the candidate interval consist of the ordered observations $\{\ell+1, \ldots, r\}$. Define the treated and control counts by

$$N_1(\ell, r) = \sum_{k=\ell+1}^{r} T_{(k)}, \qquad N_0(\ell, r) = \sum_{k=\ell+1}^{r} (1 - T_{(k)}).$$

We regard the interval as feasible if $N_1(\ell, r) \ge m_{1,n}$ and $N_0(\ell, r) \ge m_{0,n}$,where $m_{1,n}$ and $m_{0,n}$ denote minimum treated and control sample sizes required within each subclass.For a feasible interval, define the treated and control covariate means and coordinate-wise variance vectors as

$$\bar{X}_1(\ell, r) = \frac{\sum_{k=\ell+1}^{r} T_{(k)} X_{(k)}}{N_1(\ell, r)}, \qquad \bar{V}_1(\ell, r) = \frac{\sum_{k=\ell+1}^{r} T_{(k)} \left(X_{(k)} - \bar{X}_1(\ell, r)\right)^{\odot 2}}{N_1(\ell, r)},$$

$$\bar{X}_0(\ell, r) = \frac{\sum_{k=\ell+1}^{r} (1 - T_{(k)}) X_{(k)}}{N_0(\ell, r)}, \qquad \bar{V}_0(\ell, r) = \frac{\sum_{k=\ell+1}^{r} (1 - T_{(k)}) \left(X_{(k)} - \bar{X}_0(\ell, r)\right)^{\odot 2}}{N_0(\ell, r)},$$

The interval imbalance cost is defined as

$$C(\ell, r) = \begin{cases} d_{\mathrm{SMD}}\left(\bar{X}_1(\ell, r), \bar{X}_0(\ell, r), \bar{V}_1(\ell, r), \bar{V}_0(\ell, r)\right), & \begin{array}{l} N_1(\ell, r) \ge m_{1,n}, \\ N_0(\ell, r) \ge m_{0,n}, \end{array} \\ +\infty, & \text{otherwise.} \end{cases}$$

For boundaries$0 = b_0 < b_1 < \cdots < b_K = n$, the finite-sample partitioning objective is

$$\min_{0=b_0<b_1<\cdots<b_K=n} \sum_{j=1}^{K_n} C(b_{j-1}, b_j).$$

Thus, each subclass contributes one local imbalance cost with equal weight. Let $D[r, j]$ denote the minimum cost of partitioning the first r ordered observations into j feasible contiguous subclasses. The optimal-substructure property gives the recursion

$$D[r, j] = \min_{j-1 \le \ell < r} \left\{ D[\ell, j-1] + C(\ell, r) \right\},$$

with the optimal value given by $D[n, K]$. The associated split points are stored and backtracked to recover the optimal boundaries. In implementation, subclass boundaries are restricted to legal gaps in the ordered propensity scores, including the requirement that observations with identical scores are not separated. The resulting boundaries are then used to construct the hard indicators B_j and the differentiable subclassification weights in Section 3.1. The pseudocode and implementation details are provided in Appendix B. Remark: After sorting and interval-statistic precomputation, the dynamic-programming recursion has time complexity $O(K_n n^2)$.

3.3 Network Training and Alternating Optimization

Let θ denote the parameters of the network. Let

$$\mathcal{G} = \{g_0, g_1, \ldots, g_K\}, \qquad 0 = g_0 < g_1 < \cdots < g_K = 1,$$

denote the set of ordered interval boundaries induced by index boundaries $\{b_0, \ldots, b_K\}$ obtained in Section 3.2, which define the current subclassification partition. For a fixed set of boundaries $\mathcal{G}$, the network is trained using an augmented loss function

$$L_{\mathrm{aug}}(\theta; \mathcal{G}) = L_{\mathrm{base}}(\theta) + \lambda_\Psi L_\Psi(\theta; \mathcal{G}),$$

where $L_{\text{base}}(\theta)$ is the original network loss, $L_\Psi(\theta; \mathcal{G})$ is the subclassification balancing loss defined in Section 3.1, and $\lambda_\Psi \geq 0$ controls the contribution of the balancing loss.

At each outer iteration, the subclassification boundaries are held fixed while the network parameters are updated by minimizing $L_{\text{aug}}(\theta; \mathcal{G})$ for a prescribed number of inner optimization steps. After the network update, the estimated propensity scores are recomputed using the updated network. Conditioning on these scores, the subclassification boundaries are then updated by solving the dynamic-programming problem described in Section 3.2.

Let $\mathcal{B}^{(t)} = \{b_0^{(t)}, \ldots, b_K^{(t)}\}$ denote the index boundaries at outer iteration t, $\mathcal{G}^{(t)}$ similarly.The network parameters and the subclassification boundaries are alternately updated until the index boundaries $\mathcal{B}^{(t)}$ remain unchanged or the maximum number of outer iterations is reached. Since the cut points between adjacent estimated propensity scores may vary across iterations, convergence is assessed using the index boundaries rather than the numerical values of the cut points. The complete procedure is summarized in Appendix B.

3.4 Theoretical Guarantees

This section establishes the asymptotic equivalence between the balance loss based on hard propensity-score subclasses and its differentiable smooth counterpart. The result justifies replacing the non-differentiable hard subclassification indicators with the smooth weighting functions used during network optimization.

For a generic random variable Z, write

$$\widehat{\mathbb{E}}[Z] = \frac{1}{n} \sum_{i=1}^{n} Z_i.$$

The hard and smooth empirical balance losses can be rewritten as

$$L_B = \sum_{j=1}^{K_n} d_{SMD} \left(\frac{\widehat{\mathbb{E}}\left[B_j(X)TX\right]}{\widehat{\mathbb{E}}\left[B_j(X)T\right]}, \frac{\widehat{\mathbb{E}}\left[B_j(X)(1-T)X\right]}{\widehat{\mathbb{E}}\left[B_j(X)(1-T)\right]}, \hat{V}_{1j}(B_j), \hat{V}_{0j}(B_j) \right),$$

$$L_\Psi = \sum_{j=1}^{K_n} d_{SMD} \left(\frac{\widehat{\mathbb{E}}\left[\Psi_{n,j}(X)TX\right]}{\widehat{\mathbb{E}}\left[\Psi_{n,j}(X)T\right]}, \frac{\widehat{\mathbb{E}}\left[\Psi_{n,j}(X)(1-T)X\right]}{\widehat{\mathbb{E}}\left[\Psi_{n,j}(X)(1-T)\right]}, \hat{V}_{1j}(\Psi_j), \hat{V}_{0j}(\Psi_j) \right).$$

3.4.1 Assumptions

The following regularity conditions are used to establish the equivalence between L_B and L_Ψ.

Assumption 3.1 Independent Samples Assumption : All individual tuples (X_i, T_i, Y_i) are mutually independent across samples, while heterogeneous marginal distributions are permitted

Assumption 3.2 (Uniformly bounded moments of order greater than two). There exist constants $\eta > 0$ and $C_{2+\eta} < \infty$, independent of n and i, such that

$$\sup_{n \geq 1} \sup_{1 \leq i \leq n} \mathbb{E}\left[\|X_i\|^{2+\eta}\right] \leq C_{2+\eta}.$$

Assumption 3.3 (Average convergence of smooth subclass weights). For every subclass j and every $\varepsilon > 0$,

$$\frac{1}{n} \sum_{i=1}^{n} \Pr\left(|B_j(X_i) - \Psi_{n,j}(X_i)| > \varepsilon\right) \longrightarrow 0. \tag{2}$$

Assumption 3.4 (Bounded number of subclasses). There exists a finite constant $K_{\max}$ such that

$$K_n \leq K_{\max}$$

for all n.

Assumption 3.5 (Empirical within-subclass positivity). For each $j = 1, \ldots, K$, there exist constants $\delta_{1,j} > 0$ and $\delta_{0,j} > 0$ such that

$$\widehat{\mathbb{E}}\left[B_j(X)T\right] \geq \delta_{1,j}$$

and

$$\widehat{\mathbb{E}}\left[B_j(X)(1-T)\right] \geq \delta_{0,j}.$$

This condition ensures that the treated and control subgroup means appearing in L_B are well defined with probability approaching one. It is consistent with the minimum treated and control sample-size constraints imposed in the dynamic-programming partitioning procedure.

Assumption 3.6 (Nondegenerate pooled within-subclass variances). For a vector $a \in \mathbb{R}^p$, let $[a]_r$ denote its rth coordinate, where $r \in \{1, \ldots, p\}$ indexes the covariates. There exists a constant $\varepsilon_V > 0$, independent of n, such that

$$\min_{1\leq j\leq K} \min_{1\leq r\leq p} \left[\frac{\hat{V}_{1j}(B_j) + \hat{V}_{0j}(B_j)}{2}\right]_r \geq \varepsilon_V$$

3.4.2 Main Theorem

Theorem 3.1 (Asymptotic equivalence of hard and smooth balance losses)**.** *Suppose that Assumptions 3.1-3.6 hold. Then*

$$L_B - L_\Psi \xrightarrow{p} 0.$$

The theorem states that the differentiable loss optimized during network training is asymptotically equivalent to the original balance loss defined using hard propensity-score subclasses. Thus, as the sample size n grows, replacing hard subclass membership by smooth weights does not asymptotically alter the empirical balance objective.

Remark:The complete proof is provided in Appendix A, and whether the assumptions can hold is discussed in Appendix C.

3.4.3 Construction of Smooth Subclassification Functions

For the jth subclass, let $c_{n,j}$ denote its center and let $h_{n,j} > 0$ denote its half-width on the estimated propensity-score scale. The corresponding hard subclass indicator can be written as

$$B_j(X) = \mathbf{1}\left\{|\widehat{e}(\psi_{n,j}(X)) - c_{n,j}| \leq h_{n,j}\right\}.$$

Let $\delta_n > 0$ be a smoothing parameter satisfying

$$\delta_n \longrightarrow 0.$$

Define

$$z_{n,j}(X) = \frac{h_{n,j} - |\widehat{e}(\psi_{n,j}(X)) - c_{n,j}|}{\delta_n}.$$

When an observation lies inside the jth hard subclass, $z_{n,j}(X) > 0$; when it lies outside the subclass, $z_{n,j}(X) < 0$. As δ_n decreases, the example smooth function below approaches the corresponding hard indicator away from the interval boundary.

Sigmoid smoothing. The sigmoid-based smooth weight is

$$\Psi_{n,j}^{\text{sigmoid}}(X) = \frac{1}{1 + \exp\{-z_{n,j}(X)\}}.$$

This example construction is differentiable with respect to the estimated propensity score and therefore permits gradient-based optimization of the propensity network. It also satisfies

$$0 < \Psi_{n,j}(X) < 1,$$

and becomes increasingly close to hard membership indicators as δ_n decreases.

Remark:We propose more constructions in Appendix C.

Table 1: Performance comparison of baseline and enhanced models on the ACIC and IHDP datasets. Results are reported as mean ± standard deviation.

Model	ACIC		IHDP	
	PEHE	ATE Bias	PEHE	ATE Bias
DoubleRobust (Bang & Robins (2005))	3.098 ± 1.369	0.402 ± 0.365	2.952 ± 3.235	0.198 ± 0.255
Dragonnet	2.606 ± 1.133	0.393 ± 0.337	2.039 ± 1.121	0.419 ± 0.675
SLearner (Künzel et al. (2019))	2.495 ± 1.022	0.232 ± 0.192	3.097 ± 3.766	0.395 ± 0.616
Subclassification	3.731 ± 1.860	0.430 ± 0.395	5.448 ± 7.924	0.421 ± 0.395
TLearner (Künzel et al. (2019))	2.498 ± 0.586	0.233 ± 0.111	2.250 ± 1.822	0.170 ± 0.107
XLearner (Künzel et al. (2019))	2.218 ± 0.643	0.185 ± 0.145	2.384 ± 2.245	0.210 ± 0.175
BCAUSS	3.199 ± 1.103	0.318 ± 0.099	1.835 ± 0.625	0.154 ± 0.133
CausalForest (Wager & Athey (2018))	2.647 ± 1.161	0.168 ± 0.141	3.721 ± 4.663	0.351 ± 0.903
TARNet	3.054 ± 1.286	0.237 ± 0.265	1.964 ± 1.060	0.254 ± 0.336
VCNet (Nie et al. (2021))	3.202 ± 1.206	0.243 ± 0.185	1.960 ± 0.879	0.290 ± 0.324
BCAUSS + Subclassification	$\mathbf{2.255 \pm 0.961}$	$\mathbf{0.133 \pm 0.123}$	1.988 ± 1.002	0.146 ± 0.132
Dragonnet + Subclassification	2.317 ± 0.998	0.234 ± 0.177	$\mathbf{1.679 \pm 0.177}$	$\mathbf{0.092 \pm 0.082}$

4 Experiments

4.1 Experimental Setup

We evaluate the proposed model on the Infant Health and Development Program (IHDP)Hill (2011) and Atlantic Causal Inference Conference (ACIC) Dorie et al. (2019) benchmarks. For each benchmark, we use ten dataset realizations with fixed random seeds. In each realization, 20% of the samples are held out for testing, while the remaining 80% are used for hyperparameter tuning via five-fold cross-validation, with a budget of 50 hyperparameter configurations.

Let $\tau_i = Y_i(1) - Y_i(0)$ be the individual treatment effect. For a test set of size n_{test}, we compute

$$\text{ATE} = \frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} \tau_i, \qquad \widehat{\text{ATE}} = \frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} \hat{\tau}_i,$$

$$\text{ATE}_{\text{bias}} = \left| \widehat{\text{ATE}} - \text{ATE} \right|, \qquad \text{PEHE} = \left(\frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} (\hat{\tau}_i - \tau_i)^2 \right)^{1/2},$$

where $\hat{\tau}_i$ is the estimated individual treatment effect. We compute both metrics on the test set of each realization and report their means and standard deviations across the ten realizations.

4.2 Experimental Results

We compare our approach with ten baseline models, including DoubleRobust, Dragonnet, VCNet, and BCAUSS. To evaluate the effectiveness of our proposed improvements, we integrate our plug-in model with Dragonnet and BCAUSS, resulting in two enhanced models, namely, DragonnetWithSubclassification and BCAUSSWithSubclassification. The experimental results are reported in Table 1. The additional box plots are provided in Appendix D

On the IHDP dataset, incorporating subclassification consistently reduces ATE Bias for both Dragonnet and BCAUSS. However, the effect on PEHE is less consistent across models and datasets. A similar pattern is observed on the ACIC dataset. BCAUSS with subclassification achieves the lowest ATE Bias among all evaluated models, while Dragonnet also achieves a substantial improvement after incorporating subclassification.

ATE Bias consistently decreases, which directly aligns with the method's stated objective. PEHE shows a more heterogeneous response because the proposed module is designed to improve nuisance-function estimation and local covariate balance for ATE estimation, rather than directly optimizing individual treatment effects.

4.3 Ablation Study and Key Hyperparameter Selection

To investigate the effects of subclassification and the iterative procedure, we compare the following models:

- the base models, namely, Dragonnet and BCAUSS;
- the base models augmented with subclassification but without the iterative procedure; and
- the base models augmented with both subclassification and the iterative procedure.

The experimental results are presented in Table 2. By comparing these model variants, we observe that combining subclassification with the iterative procedure improves ATE estimation. In contrast, using subclassification without iteration leads to inconsistent changes in ATE estimation and degrades PEHE estimation for both base models. One possible explanation is that, without iterative updates, the subclassification intervals remain fixed while the learned propensity scores evolve during optimization, potentially making the intervals increasingly misaligned with the current propensity-score distribution.

Table 2: Ablation study of Dragonnet and BCAUSS on the IHDP dataset.

(a) Dragonnet

Model Variant	PEHE	ATE Bias
Dragonnet	2.039 ± 1.121	0.419 ± 0.675
Dragonnet + Subclassification	2.163 ± 1.364	0.369 ± 0.453
Dragonnet + Subclassification + Iteration	$\mathbf{1.679 \pm 0.177}$	$\mathbf{0.092 \pm 0.082}$

(b) BCAUSS

Model Variant	PEHE	ATE Bias
BCAUSS	1.835 ± 0.625	0.154 ± 0.133
BCAUSS + Subclassification	2.083 ± 1.082	0.198 ± 0.133
BCAUSS + Subclassification + Iteration	$\mathbf{1.988 \pm 1.002}$	$\mathbf{0.146 \pm 0.132}$

In addition, we perform a sensitivity analysis of two key hyperparameters: the number of subclasses K and the approximation parameter δ_n. The hyperparameters are selected using the same five-fold cross-validation protocol described in Section 4.1, with ATE Bias used as the selection criterion. The detailed results are provided in Appendix D. For BCAUSS, we select $K = 5$ and $\delta_n = n^{-1/3}$, whereas for Dragonnet, we select $K = 10$ and $\delta_n = n^{-1/3}$. The difference in the selected K may be related to the fact that BCAUSS already incorporates a global covariate-balancing objective. A larger number of subclasses introduces more localized balancing constraints, which may increase the interaction between the existing global balancing objective and the proposed local balancing objective, potentially leading to greater optimization variability.

4.4 Stability Verification

To verify the stability of the iterative procedure and examine whether the iterative training process exhibits noticeable oscillations, we recorded the loss value at every 10% of the total number of iterations in each experiment. For each corresponding iteration percentage, we then calculated the median, lower quartile, and upper quartile of the loss values across the ten experimental runs. Based on these statistics, we plotted the iteration progress versus loss value, as shown in Figure 1.

As shown in Figure 1, the overall loss decreases consistently throughout training, with no pronounced oscillations observed across the recorded iterations. This provides empirical evidence that the proposed iterative procedure can be trained stably.

4.5 Balance Test

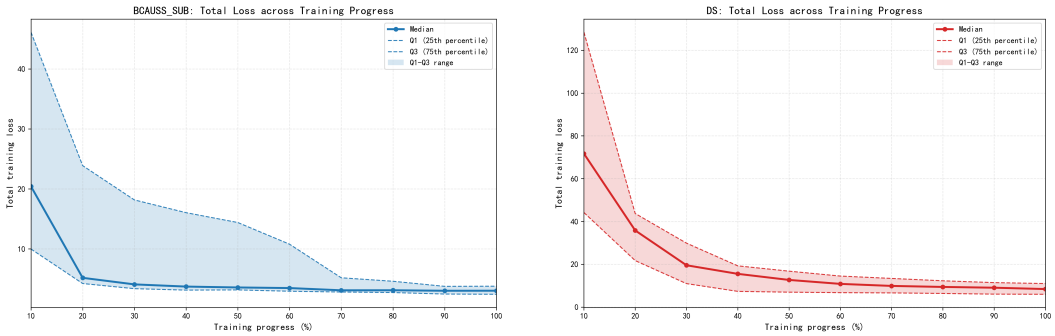
To verify the balance of our method, we conduct the following hypothesis test:

$$H_0 : \quad T \perp X \mid e(\psi(X))$$

for each of the ten experimental realizations.

To aggregate evidence across the ten experiments, we use Fisher's method ((Fisher, 1934)):

$$T_{\text{Fisher}} = -2 \sum_{i=1}^{10} \log(p_i),$$



(a) BCAUSS with subclassification

(b) Dragonnet with subclassification

Figure 1: Training loss curves with the median and interquartile range over ten experimental runs.

Table 3: Results of the balance test.

Model	Fisher pooled p-value
BCAUSS	0.7987
BCAUSSWithSubclassification	0.9691
Dragonnet	0.0179
DragonnetWithSubclassification	0.2283

where p_i is the p-value obtained by conducting a permutation test (Good (2005)) on the i-th experimental realization.

The results are reported in Table 3. Incorporating subclassification weakens the evidence against the conditional-independence null hypothesis for both Dragonnet and BCAUSS. Notably, this improvement is also observed for BCAUSS, which already incorporates a global covariate-balancing objective. This result provides empirical support for the benefit of introducing local covariate balance through subclassification.

5 Limitations and Future Work Directions

Our method has several limitations. First, we do not theoretically establish whether achieving zero d_{SMD} within each propensity-score subclass is equivalent to the conditional independence condition $T \perp X \mid e(\psi(X))$. Instead, we introduce regularization terms into the loss function to encourage the model toward better covariate balance during training. This is essentially a heuristic loss-function design and does not establish rigorous theoretical guarantees for the resulting balance.

Second, the convergence guarantee for our loss function relies on an assumption that may be reasonable in practice but is difficult to verify directly. Although Appendix C discusses the plausibility and rationale of this assumption, it does not provide a practical procedure for verifying, within the actual algorithm, whether the conditions required by the theorem are satisfied.

Future work will primarily focus on investigating whether heuristic search methods can be used to replace dynamic programming (DP) while ensuring consistency between the two loss functions used in the iterative procedure. If successful, this approach could provide the model with effective subclassification and reduce its computational cost.

6 Conclusion

This work presents a framework that incorporates subclassification into neural network training. We also provide a convergence proof for the loss function under the smoothed subclassification framework. Our experiments demonstrate the effectiveness of the proposed method and the stability of its training process. In addition, we provide practical guidance for selecting the key hyperparameters.

REPRODUCIBILITY STATEMENT

To facilitate the reproducibility of our work, we have taken the following measures:

1. All random seeds used in the experiments are fixed and explicitly reported.
2. All datasets used in this work are clearly cited and publicly available, subject to their respective licenses and usage restrictions. The data preprocessing procedures and dataset splits are also documented.
3. The complete experimental code is available, together with detailed instructions and commands for running the experiments.
4. We provide the complete software environment, including the programming language version, major library versions, and dependency specifications.
5. We report the hardware and runtime configurations used in the experiments, including the GPU model, number of GPUs, and other relevant computational resources.
6. We report the key experimental details and hyperparameters, including the optimizer, learning rate, batch size, number of training epochs, model configuration, and hyperparameter search ranges.
7. We provide the results of all experimental runs, including the results obtained under different random seeds whenever applicable.
8. We provide the scripts and instructions used to generate the figures and tables reported in the paper from the experimental results.

AI USE STATEMENT

In this work, we used generative AI tools for the following tasks:

- discussing possible relaxations of Assumption 3.3;
- converting handwritten proof procedures into LATEX code;
- discussing the design of ablation studies and the evaluation metrics used in the experiments;
- generating LATEX tables based on the experimental results;
- providing suggestions for the search ranges of model hyperparameters;
- editing and polishing the manuscript to improve readability; and
- drafting portions of the manuscript.

All AI-assisted content was carefully reviewed and verified by the authors. In particular, the authors checked the correctness of the mathematical arguments, experimental designs, experimental results, tables, hyperparameter settings, and the final manuscript. The authors take full responsibility for the final content of this work, including all text, mathematical claims, experimental results, and other artifacts produced with the assistance of generative AI.

REFERENCES

Heejung Bang and James M. Robins. Doubly robust estimation in missing data and causal inference models. *Biometrics*, 61(4):962–973, 2005. doi: 10.1111/j.1541-0420.2005.00377.x.

Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. Double/debiased machine learning for treatment and structural parameters. *The Econometrics Journal*, 21(1):C1–C68, 2018. doi: 10.1111/ectj.12097.

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

Ronald A. Fisher. *Statistical Methods for Research Workers*. Oliver and Boyd, Edinburgh, 5 edition, 1934.

Phillip I. Good. *Permutation, parametric and bootstrap tests of hypotheses*. Springer, New York, 3 edition, 2005.

Jennifer L. Hill. Bayesian nonparametric modeling for causal inference. *Journal of Computational and Graphical Statistics*, 20(1):217–240, 2011. doi: 10.1198/jcgs.2010.08162.

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*, 116(10):4156–4165, 2019. doi: 10.1073/pnas.1804597116.

Lizhen Nie, Mao Ye, Qiang Liu, and Dan Nicolae. VCNet and functional targeted regularization for learning causal effects of continuous treatments. *arXiv preprint arXiv:2103.07861*, 2021. doi: 10.48550/arXiv.2103.07861.

Paul R. Rosenbaum and Donald B. Rubin. The central role of the propensity score in observational studies for causal effects. *Biometrika*, 70(1):41–55, 1983.

Paul R. Rosenbaum and Donald B. Rubin. Reducing bias in observational studies using subclassification on the propensity score. *Journal of the American Statistical Association*, 79(387):516–524, 1984. doi: 10.1080/01621459.1984.10478078.

Uri Shalit, Fredrik D. Johansson, and David Sontag. Estimating individual treatment effect: Generalization bounds and algorithms. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 3076–3085. PMLR, 2017.

Claudia Shi, David M. Blei, and Victor Veitch. Adapting neural networks for the estimation of treatment effects. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Gino Tesei, Stefanos Giampanis, Jingpu Shi, and Beau Norgeot. Learning end-to-end patient representations through self-supervised covariate balancing for causal treatment effect estimation. *Journal of Biomedical Informatics*, 140:104339, March 2023. ISSN 1532-0464. doi: 10.1016/j.jbi.2023.104339.

Stefan Wager and Susan Athey. Estimation and inference of heterogeneous treatment effects using random forests. *Journal of the American Statistical Association*, 113(523):1228–1242, 2018. doi: 10.1080/01621459.2017.1319839.

A Complete Proof of the Asymptotic Equivalence Theorem

A.1 Notation and Setup

Let $\{(X_i, T_i)\}_{i=1}^n$ be a sequence of independent but not necessarily identically distributed random vectors, where

$$X_i \in \mathbb{R}^p, \qquad T_i \in \{0, 1\}.$$

For notational simplicity, write $T_{1i} = T_i$ and $T_{0i} = 1 - T_i$. Let K_n denote the current number of propensity-score subclasses. By Assumption 3.4, the number of subclasses is uniformly bounded, meaning that there exists a finite constant $K_{\max}$ such that

$$K_n \leq K_{\max}, \qquad \text{for all } n.$$

For a generic subclass weight $W_j(\cdot)$, we define the empirical weighted mass for treatment group $t \in \{0, 1\}$ by

$$\widehat{C}_{tj}(W_j) = \widehat{\mathbb{E}}[W_j(X)T_t] = \frac{1}{n}\sum_{i=1}^{n} W_j(X_i)T_{ti}.$$

For $q \in \{0, 1, 2\}$, define the empirical weighted raw moments by

$$\widehat{R}_{tj}^{(q)}(W_j) = \widehat{\mathbb{E}}\big[W_j(X)T_t X^{\odot q}\big] = \frac{1}{n}\sum_{i=1}^{n} W_j(X_i)T_{ti}X_i^{\odot q},$$

where

$$X_i^{\odot 0} = 1, \qquad X_i^{\odot 1} = X_i, \qquad X_i^{\odot 2} = (X_{i1}^2, \ldots, X_{ip}^2)^\top.$$

Thus,

$$\widehat{R}_{tj}^{(0)}(W_j) = \widehat{C}_{tj}(W_j).$$

When $\widehat{C}_{tj}(W_j) > 0$, define the weighted treatment-group mean vector and the weighted second-moment vector as

$$\widehat{\mu}_{tj}(W_j) = \frac{\widehat{R}_{tj}^{(1)}(W_j)}{\widehat{C}_{tj}(W_j)}, \qquad \widehat{M}_{tj}(W_j) = \frac{\widehat{R}_{tj}^{(2)}(W_j)}{\widehat{C}_{tj}(W_j)}.$$

The corresponding coordinate-wise weighted variance vector is

$$\widehat{V}_{tj}(W_j) = \widehat{M}_{tj}(W_j) - \widehat{\mu}_{tj}(W_j)^{\odot 2}.$$

Let $B_j(\cdot)$ denote the hard subclassification indicator and $\Psi_{n,j}(\cdot)$ its smooth approximation. The hard and smooth balance losses are

$$L_B = \sum_{j=1}^{K_n} d_{\mathrm{SMD}}\Big(\widehat{\mu}_{1j}(B_j), \widehat{\mu}_{0j}(B_j), \widehat{V}_{1j}(B_j), \widehat{V}_{0j}(B_j)\Big),$$

and

$$L_\Psi = \sum_{j=1}^{K_n} d_{\mathrm{SMD}}\Big(\widehat{\mu}_{1j}(\Psi_{n,j}), \widehat{\mu}_{0j}(\Psi_{n,j}), \widehat{V}_{1j}(\Psi_{n,j}), \widehat{V}_{0j}(\Psi_{n,j})\Big).$$

For the j-th subclass, define the hard–soft discrepancy by

$$D_{n,i,j} := B_j(X_i) - \Psi_{n,j}(X_i).$$

Since both $B_j(X_i)$ and $\Psi_{n,j}(X_i)$ take values in $[0, 1]$, we have, almost surely,

$$|D_{n,i,j}| \le 1.$$

A.2 A Ratio-Difference Lemma

The following elementary lemma will be used repeatedly to compare quantities of the form A_n/C_n and B_n/D_n.

Lemma 1 (Ratio-difference lemma)**.** *Let $A_n, B_n \in \mathbb{R}^p$ and $C_n, D_n \in \mathbb{R}$ be sequences. Suppose that*

$$A_n - B_n = o_p(1), \qquad C_n - D_n = o_p(1), \qquad A_n = O_p(1), \qquad C_n = O_p(1).$$

Assume also that there exists a constant $c > 0$ such that

$$\Pr(|C_n| \ge c) \to 1, \qquad \Pr(|D_n| \ge c) \to 1.$$

Then

$$\frac{A_n}{C_n} - \frac{B_n}{D_n} = o_p(1).$$

Proof. First, write the difference with a common denominator:

$$\frac{A_n}{C_n} - \frac{B_n}{D_n} = \frac{A_n D_n - B_n C_n}{C_n D_n}.$$

The numerator can be decomposed by adding and subtracting the term $A_n C_n$:

$$A_n D_n - B_n C_n = A_n D_n - A_n C_n + A_n C_n - B_n C_n.$$

Thus,

$$A_n D_n - B_n C_n = A_n(D_n - C_n) + C_n(A_n - B_n).$$

By assumption, $A_n = O_p(1)$ and $D_n - C_n = o_p(1)$, so their product is $o_p(1)$. Similarly, $C_n = O_p(1)$ and $A_n - B_n = o_p(1)$, hence

$$A_n(D_n - C_n) + C_n(A_n - B_n) = o_p(1) + o_p(1) = o_p(1).$$

Therefore,

$$A_n D_n - B_n C_n = o_p(1).$$

On the other hand, since

$$\Pr(|C_n| \geq c) \to 1, \qquad \Pr(|D_n| \geq c) \to 1,$$

we have

$$\Pr(|C_n D_n| \geq c^2) \to 1.$$

It follows that

$$\frac{1}{C_n D_n} = O_p(1).$$

Combining the two results gives

$$\frac{A_n D_n - B_n C_n}{C_n D_n} = o_p(1) \cdot O_p(1) = o_p(1).$$

Hence,

$$\frac{A_n}{C_n} - \frac{B_n}{D_n} = o_p(1),$$

which proves the lemma. □

A.3 Convergence of Weighted Discrepancy Moment Bounds

We next show that the average discrepancy between the hard and soft weights is negligible even after weighting by covariate moments.

For any $\delta > 0$, define

$$p_{n,i,j}(\delta) := \Pr\left(|D_{n,i,j}| > \delta\right).$$

Let

$$\alpha := \frac{\eta}{2+\eta} \in (0,1),$$

where $\eta > 0$ is given by Assumption 3.2. By splitting according to the events $\{|D_{n,i,j}| \leq \delta\}$ and $\{|D_{n,i,j}| > \delta\}$, we obtain

$$\mathbb{E}\left[|D_{n,i,j}| \|X_i\|^2\right] \leq \delta \mathbb{E}\left[\|X_i\|^2\right] + \mathbb{E}\left[\|X_i\|^2 \mathbf{1}\{|D_{n,i,j}| > \delta\}\right].$$

By Lyapunov's inequality and Assumption 3.2,

$$\mathbb{E}\left[\|X_i\|^2\right] \leq \left\{\mathbb{E}\left[\|X_i\|^{2+\eta}\right]\right\}^{2/(2+\eta)} \leq C_{2+\eta}^{2/(2+\eta)}.$$

Next, by Hölder's inequality with conjugate exponents

$$q = \frac{2+\eta}{2}, \qquad q' = \frac{2+\eta}{\eta},$$

we have

$$\mathbb{E}\left[\|X_i\|^2 \mathbf{1}\{|D_{n,i,j}| > \delta\}\right] \leq \left\{\mathbb{E}\left[\|X_i\|^{2+\eta}\right]\right\}^{2/(2+\eta)} \left\{\Pr\left(|D_{n,i,j}| > \delta\right)\right\}^{\eta/(2+\eta)}.$$

Therefore,

$$\mathbb{E}\left[\|X_i\|^2 \mathbf{1}\{|D_{n,i,j}| > \delta\}\right] \leq C_{2+\eta}^{2/(2+\eta)} p_{n,i,j}(\delta)^{\alpha}.$$

Combining the preceding display with the bound for $\mathbb{E}[\|X_i\|^2]$ yields

$$\mathbb{E}\left[|D_{n,i,j}| \|X_i\|^2\right] \leq \delta C_{2+\eta}^{2/(2+\eta)} + C_{2+\eta}^{2/(2+\eta)} p_{n,i,j}(\delta)^{\alpha}.$$

Averaging over $i = 1, \ldots, n$, we get

$$\frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\left[|D_{n,i,j}| \|X_i\|^2\right] \leq \delta C_{2+\eta}^{2/(2+\eta)} + C_{2+\eta}^{2/(2+\eta)} \left(\frac{1}{n}\sum_{i=1}^{n} p_{n,i,j}(\delta)^{\alpha}\right).$$

Because the function $x \mapsto x^\alpha$ is concave on $[0, \infty)$ for $\alpha \in (0, 1)$, Jensen's inequality gives

$$\frac{1}{n}\sum_{i=1}^{n} p_{n,i,j}(\delta)^\alpha \le \left(\frac{1}{n}\sum_{i=1}^{n} p_{n,i,j}(\delta)\right)^\alpha.$$

By Assumption 3.3, for any fixed $\delta > 0$,

$$\frac{1}{n}\sum_{i=1}^{n} \Pr\left(|D_{n,i,j}| > \delta\right) \to 0.$$

Since the number of subclasses is uniformly bounded, the convergence holds uniformly over $j = 1, \ldots, K_n$. Consequently,

$$\max_{1\le j\le K_n} \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\left[|D_{n,i,j}|\|X_i\|^2\right] \le \delta C_{2+\eta}^{2/(2+\eta)} + o(1).$$

Letting $\delta \downarrow 0$ gives

$$\max_{1\le j\le K_n} \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\left[|D_{n,i,j}|\|X_i\|^2\right] \to 0. \tag{3}$$

The corresponding zeroth-moment bound follows from the simple inequality

$$|D_{n,i,j}| \le \delta + \mathbf{1}\{|D_{n,i,j}| > \delta\}.$$

Taking expectations and averaging over i, we have

$$\frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\left[|D_{n,i,j}|\right] \le \delta + \frac{1}{n}\sum_{i=1}^{n} p_{n,i,j}(\delta).$$

Assumption 3.3 makes the second term vanish uniformly over j, and then $\delta \downarrow 0$ yields

$$\max_{1\le j\le K_n} \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\left[|D_{n,i,j}|\right] \to 0. \tag{4}$$

For the first moment, since

$$\|X_i\| \le 1 + \|X_i\|^2,$$

we have

$$|D_{n,i,j}|\|X_i\| \le |D_{n,i,j}| + |D_{n,i,j}|\|X_i\|^2.$$

Therefore, combining equation 4 and equation 3,

$$\max_{1\le j\le K_n} \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\left[|D_{n,i,j}|\|X_i\|\right] \to 0. \tag{5}$$

Equations equation 4–equation 5 will be the main moment controls used in the remaining steps.

A.4 Convergence of Unnormalized Weighted Moments

Recall that

$$D_{n,i,j} = B_j(X_i) - \Psi_{n,j}(X_i).$$

For $q \in \{0, 1, 2\}$, $t \in \{0, 1\}$, and subclass j, define

$$U_{n,i,j,t}^{(q)} := D_{n,i,j} T_{ti} X_i^{\odot q}.$$

Because $|D_{n,i,j}| \le 1$, $T_{ti} \in \{0, 1\}$, and

$$\|X_i^{\odot 2}\| = \left(\sum_{r=1}^{p} X_{ir}^4\right)^{1/2} \le \sum_{r=1}^{p} X_{ir}^2 = \|X_i\|^2,$$

we obtain the bounds

$$\left\|U^{(0)}_{n,i,j,t}\right\| \le |D_{n,i,j}|,$$
$$\left\|U^{(1)}_{n,i,j,t}\right\| \le |D_{n,i,j}|\|X_i\|,$$

and

$$\left\|U^{(2)}_{n,i,j,t}\right\| \le |D_{n,i,j}|\|X_i\|^2.$$

Using the definitions of $\widehat{R}^{(q)}_{tj}$, we have, for every $q \in \{0,1,2\}$,

$$\widehat{R}^{(q)}_{tj}(B_j) - \widehat{R}^{(q)}_{tj}(\Psi_{n,j}) = \frac{1}{n}\sum_{i=1}^{n} U^{(q)}_{n,i,j,t}.$$

By equation 3, equation 4, and equation 5, it follows that

$$\max_{1\le j\le K_n} \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\left[\left\|U^{(q)}_{n,i,j,t}\right\|\right] \to 0, \qquad q \in \{0,1,2\}. \tag{6}$$

To convert this bound into convergence in probability, decompose

$$\frac{1}{n}\sum_{i=1}^{n} U^{(q)}_{n,i,j,t} = \frac{1}{n}\sum_{i=1}^{n}\left(U^{(q)}_{n,i,j,t} - \mathbb{E}\left[U^{(q)}_{n,i,j,t}\right]\right) + \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left[U^{(q)}_{n,i,j,t}\right].$$

For the second term, by equation 6,

$$\max_{1\le j\le K_n}\left\|\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left[U^{(q)}_{n,i,j,t}\right]\right\| \le \max_{1\le j\le K_n}\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left[\left\|U^{(q)}_{n,i,j,t}\right\|\right] \to 0. \tag{7}$$

For the first term, fix $\varepsilon > 0$. By Markov's inequality and the triangle inequality,

$$\Pr\left(\left\|\frac{1}{n}\sum_{i=1}^{n}\left(U^{(q)}_{n,i,j,t} - \mathbb{E}\left[U^{(q)}_{n,i,j,t}\right]\right)\right\| > \varepsilon\right)$$
$$\le \frac{1}{\varepsilon}\mathbb{E}\left[\left\|\frac{1}{n}\sum_{i=1}^{n}\left(U^{(q)}_{n,i,j,t} - \mathbb{E}\left[U^{(q)}_{n,i,j,t}\right]\right)\right\|\right].$$

Using subadditivity of the norm,

$$\left\|\frac{1}{n}\sum_{i=1}^{n}\left(U^{(q)}_{n,i,j,t} - \mathbb{E}\left[U^{(q)}_{n,i,j,t}\right]\right)\right\| \le \frac{1}{n}\sum_{i=1}^{n}\left|\left\|U^{(q)}_{n,i,j,t} - \mathbb{E}\left[U^{(q)}_{n,i,j,t}\right]\right\|\right|.$$

Therefore,

$$\Pr\left(\left\|\frac{1}{n}\sum_{i=1}^{n}\left(U^{(q)}_{n,i,j,t} - \mathbb{E}\left[U^{(q)}_{n,i,j,t}\right]\right)\right\| > \varepsilon\right) \le \frac{1}{\varepsilon n}\sum_{i=1}^{n}\mathbb{E}\left[\left\|U^{(q)}_{n,i,j,t} - \mathbb{E}\left[U^{(q)}_{n,i,j,t}\right]\right\|\right].$$

Again applying the triangle inequality,

$$\left\|U^{(q)}_{n,i,j,t} - \mathbb{E}\left[U^{(q)}_{n,i,j,t}\right]\right\| \le \left\|U^{(q)}_{n,i,j,t}\right\| + \left\|\mathbb{E}\left[U^{(q)}_{n,i,j,t}\right]\right\| \le 2\left\|U^{(q)}_{n,i,j,t}\right\|,$$

where the last inequality follows by Jensen's inequality for conditional expectations or directly by taking norms inside the expectation. Hence,

$$\Pr\left(\left\|\frac{1}{n}\sum_{i=1}^{n}\left(U^{(q)}_{n,i,j,t} - \mathbb{E}\left[U^{(q)}_{n,i,j,t}\right]\right)\right\| > \varepsilon\right) \le \frac{2}{\varepsilon n}\sum_{i=1}^{n}\mathbb{E}\left[\left\|U^{(q)}_{n,i,j,t}\right\|\right].$$

By equation 6, the right-hand side converges to zero uniformly over j. Therefore,

$$\max_{1\le j\le K_n}\left\|\frac{1}{n}\sum_{i=1}^{n}\left(U^{(q)}_{n,i,j,t} - \mathbb{E}\left[U^{(q)}_{n,i,j,t}\right]\right)\right\| = o_p(1). \tag{8}$$

Combining equation 7, equation 8, and the identity

$$\widehat{R}_{tj}^{(q)}(B_j) - \widehat{R}_{tj}^{(q)}(\Psi_{n,j}) = \frac{1}{n}\sum_{i=1}^{n} U_{n,i,j,t}^{(q)},$$

we conclude that, for $q \in \{0,1,2\}$,

$$\max_{1\le j\le K_n} \left\| \widehat{R}_{tj}^{(q)}(B_j) - \widehat{R}_{tj}^{(q)}(\Psi_{n,j}) \right\| = o_p(1). \tag{9}$$

In particular,

$$\max_{1\le j\le K_n} \left| \widehat{C}_{tj}(B_j) - \widehat{C}_{tj}(\Psi_{n,j}) \right| = o_p(1), \tag{10}$$
$$\max_{1\le j\le K_n} \left\| \widehat{R}_{tj}^{(1)}(B_j) - \widehat{R}_{tj}^{(1)}(\Psi_{n,j}) \right\| = o_p(1),$$

and

$$\max_{1\le j\le K_n} \left\| \widehat{R}_{tj}^{(2)}(B_j) - \widehat{R}_{tj}^{(2)}(\Psi_{n,j}) \right\| = o_p(1).$$

These equations state that the unnormalized empirical weighted masses, first moments, and second moments induced by the hard and smooth weights are asymptotically equivalent uniformly over subclasses.

A.5 Positivity of Empirical Denominators and Boundedness of Numerators

By Assumption 3.5, for each subclass j and treatment group $t \in \{0,1\}$, there exists a constant $\delta_{t,j} > 0$ such that

$$\Pr\left(\widehat{C}_{tj}(B_j) \ge \delta_{t,j}\right) = 1.$$

Using equation 10, we also obtain

$$\widehat{C}_{tj}(B_j) - \widehat{C}_{tj}(\Psi_{n,j}) = o_p(1).$$

Therefore, for each subclass j and each t,

$$\Pr\left(\widehat{C}_{tj}(\Psi_{n,j}) \ge \frac{\delta_{t,j}}{2}\right) \to 1. \tag{11}$$

Together with Assumption 3.5, equations equation 11 imply that all hard and soft denominators are bounded away from zero in probability.

We next show that the numerators appearing in the weighted means are bounded in probability. For any weight W_j satisfying $0 \le W_j(X_i) \le 1$, we have, for $q \in \{1,2\}$,

$$\left\| \widehat{R}_{tj}^{(q)}(W_j) \right\| \le \frac{1}{n}\sum_{i=1}^{n} \left\| X_i^{\odot q} \right\|.$$

For $q = 1$,

$$\|X_i\| \le \frac{1}{2}\left(1 + \|X_i\|^2\right).$$

For $q = 2$, as shown above,

$$\|X_i^{\odot 2}\| \le \|X_i\|^2.$$

Thus,

$$\mathbb{E}\left[\left\| \widehat{R}_{tj}^{(q)}(W_j) \right\|\right] \le \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\left[1 + \|X_i\|^2\right] \le C_X < \infty,$$

where the uniform boundedness follows from Assumption 3.2. By Markov's inequality,

$$\max_{1\le j\le K_n} \left\| \widehat{R}_{tj}^{(q)}(B_j) \right\| = O_p(1), \qquad q \in \{1,2\}, \tag{12}$$

and similarly,

$$\max_{1\le j\le K_n} \left\| \widehat{R}_{tj}^{(q)}(\Psi_{n,j}) \right\| = O_p(1), \qquad q \in \{1,2\}.$$

Also, since the denominator $\widehat{C}_{tj}(B_j)$ is bounded in probability because the weights are bounded and $0 \le \widehat{C}_{tj}(B_j) \le 1$, we have

$$\widehat{C}_{tj}(B_j) = O_p(1).$$

A.6 Convergence of Weighted Means and Weighted Second Moments

We now compare the actual weighted means used in the balance loss. Recall

$$\widehat{\mu}_{tj}(W_j) = \frac{\widehat{R}^{(1)}_{tj}(W_j)}{\widehat{C}_{tj}(W_j)}.$$

For fixed t and j, apply Lemma 1 with

$$A_n = \widehat{R}^{(1)}_{tj}(B_j), \qquad B_n = \widehat{R}^{(1)}_{tj}(\Psi_{n,j}),$$
$$C_n = \widehat{C}_{tj}(B_j), \qquad D_n = \widehat{C}_{tj}(\Psi_{n,j}).$$

The conditions of the lemma are satisfied because of:

- equation 9, which gives
$$\widehat{R}^{(1)}_{tj}(B_j) - \widehat{R}^{(1)}_{tj}(\Psi_{n,j}) = o_p(1);$$
- equation 10, which gives
$$\widehat{C}_{tj}(B_j) - \widehat{C}_{tj}(\Psi_{n,j}) = o_p(1);$$
- equation 12, which implies
$$\widehat{R}^{(1)}_{tj}(B_j) = O_p(1);$$
- the positivity Assumption 3.5 and equation 11, which ensure that both denominators are bounded away from zero in probability.

Hence,

$$\widehat{\mu}_{tj}(B_j) - \widehat{\mu}_{tj}(\Psi_{n,j}) = o_p(1).$$

Because the number of subclasses is uniformly bounded, we obtain the stronger uniform statement

$$\max_{1\le j\le K_n} \|\widehat{\mu}_{tj}(B_j) - \widehat{\mu}_{tj}(\Psi_{n,j})\| = o_p(1). \tag{13}$$

The same argument applies to the weighted raw second-moment vectors. Recall

$$\widehat{M}_{tj}(W_j) = \frac{\widehat{R}^{(2)}_{tj}(W_j)}{\widehat{C}_{tj}(W_j)}.$$

By applying Lemma 1 with

$$A_n = \widehat{R}^{(2)}_{tj}(B_j), \qquad B_n = \widehat{R}^{(2)}_{tj}(\Psi_{n,j}),$$

and the same C_n, D_n, we obtain

$$\max_{1\le j\le K_n} \left\|\widehat{M}_{tj}(B_j) - \widehat{M}_{tj}(\Psi_{n,j})\right\| = o_p(1). \tag{14}$$

For any two vectors $a, b \in \mathbb{R}^p$, the element-wise square difference satisfies the identity

$$a^{\odot 2} - b^{\odot 2} = (a-b)\odot(a+b).$$

Therefore, using the Cauchy–Schwarz inequality for element-wise products,

$$\left\|\widehat{\mu}_{tj}(B_j)^{\odot 2} - \widehat{\mu}_{tj}(\Psi_{n,j})^{\odot 2}\right\| \le \|\widehat{\mu}_{tj}(B_j) - \widehat{\mu}_{tj}(\Psi_{n,j})\| \, \|\widehat{\mu}_{tj}(B_j) + \widehat{\mu}_{tj}(\Psi_{n,j})\|.$$

The first factor is $o_p(1)$ uniformly over j by equation 13. The second factor is $O_p(1)$ because both weighted means are $O_p(1)$. Hence,

$$\left\|\widehat{\mu}_{tj}(B_j)^{\odot 2} - \widehat{\mu}_{tj}(\Psi_{n,j})^{\odot 2}\right\| = o_p(1),$$

uniformly over j.

Using

$$\widehat{V}_{tj}(W_j) = \widehat{M}_{tj}(W_j) - \widehat{\mu}_{tj}(W_j)^{\odot 2},$$

we finally get

$$\max_{1\le j\le K_n} \left\|\widehat{V}_{tj}(B_j) - \widehat{V}_{tj}(\Psi_{n,j})\right\| = o_p(1). \tag{15}$$

Equations equation 13 and equation 15 establish that the weighted means and weighted variance vectors entering the SMD loss are asymptotically equivalent under the hard and soft subclassification schemes.

A.7 Convergence of the Within-Subclass SMD Discrepancies

For each subclass j and coordinate $r = 1, \ldots, p$, define

$$\widehat{\Delta}_{j,r}(B_j) := [\widehat{\mu}_{1j}(B_j) - \widehat{\mu}_{0j}(B_j)]_r, \qquad \widehat{\Delta}_{j,r}(\Psi_{n,j}) := [\widehat{\mu}_{1j}(\Psi_{n,j}) - \widehat{\mu}_{0j}(\Psi_{n,j})]_r,$$

and

$$\widehat{S}_{j,r}(B_j) := \left[\frac{\widehat{V}_{1j}(B_j) + \widehat{V}_{0j}(B_j)}{2}\right]_r, \qquad \widehat{S}_{j,r}(\Psi_{n,j}) := \left[\frac{\widehat{V}_{1j}(\Psi_{n,j}) + \widehat{V}_{0j}(\Psi_{n,j})}{2}\right]_r.$$

By equation (13),

$$\max_{1\le j\le K_n} \max_{1\le r\le p} \left|\widehat{\Delta}_{j,r}(B_j) - \widehat{\Delta}_{j,r}(\Psi_{n,j})\right| = o_p(1). \tag{16}$$

Moreover, equation (15) implies

$$\max_{1\le j\le K_n} \max_{1\le r\le p} \left|\widehat{S}_{j,r}(B_j) - \widehat{S}_{j,r}(\Psi_{n,j})\right| = o_p(1). \tag{17}$$

By Assumption 3.6,

$$\Pr\left(\min_{1\le j\le K_n} \min_{1\le r\le p} \widehat{S}_{j,r}(B_j) \ge \varepsilon_V\right) = 1. \tag{18}$$

Equation equation 17 gives

$$\Pr\left(\max_{1\le j\le K_n} \max_{1\le r\le p} \left|\widehat{S}_{j,r}(B_j) - \widehat{S}_{j,r}(\Psi_{n,j})\right| \le \frac{\varepsilon_V}{2}\right) \longrightarrow 1. \tag{19}$$

On the intersection of the events in equations equation 18 and equation 19, for every j and r,

$$\widehat{S}_{j,r}(\Psi_{n,j}) \ge \widehat{S}_{j,r}(B_j) - \left|\widehat{S}_{j,r}(B_j) - \widehat{S}_{j,r}(\Psi_{n,j})\right| \ge \frac{\varepsilon_V}{2}.$$

Therefore,

$$\Pr\left(\min_{1\le j\le K_n} \min_{1\le r\le p} \widehat{S}_{j,r}(\Psi_{n,j}) \ge \frac{\varepsilon_V}{2}\right) \longrightarrow 1. \tag{20}$$

Since the square-root function is continuous on $[\varepsilon_V/2, \infty)$, equations equation 17, equation 18, and equation 20 imply

$$\max_{1\le j\le K_n} \max_{1\le r\le p} \left|\sqrt{\widehat{S}_{j,r}(B_j)} - \sqrt{\widehat{S}_{j,r}(\Psi_{n,j})}\right| = o_p(1). \tag{21}$$

By the boundedness results in Section A.5 and equations equation 16 and equation 17,

$$\max_{1\le j\le K_n} \max_{1\le r\le p} \left|\widehat{\Delta}_{j,r}(B_j)\right| = O_p(1), \qquad \max_{1\le j\le K_n} \max_{1\le r\le p} \left|\widehat{\Delta}_{j,r}(\Psi_{n,j})\right| = O_p(1). \tag{22}$$

Furthermore, the coordinate-wise variance vectors are $O_p(1)$ by Section A.5. Hence,

$$\max_{1\le j\le K_n} \max_{1\le r\le p} \sqrt{\widehat{S}_{j,r}(B_j)} = O_p(1), \qquad \max_{1\le j\le K_n} \max_{1\le r\le p} \sqrt{\widehat{S}_{j,r}(\Psi_{n,j})} = O_p(1). \tag{23}$$

Fix j and r. Apply Lemma 1 as a scalar result with

$$A_n = \widehat{\Delta}_{j,r}(B_j), \qquad B_n = \widehat{\Delta}_{j,r}(\Psi_{n,j}),$$

and

$$C_n = \sqrt{\widehat{S}_{j,r}(B_j)}, \qquad D_n = \sqrt{\widehat{S}_{j,r}(\Psi_{n,j})}.$$

Equation equation 16 gives $A_n - B_n = o_p(1)$, and equation equation 21 gives $C_n - D_n = o_p(1)$. The stochastic boundedness conditions follow from equations equation 22 and equation 23. Finally,

equations equation 18 and equation 20 imply that both denominators are bounded away from zero in probability. Thus, Lemma 1 yields

$$\frac{\widehat{\Delta}_{j,r}(B_j)}{\sqrt{\widehat{S}_{j,r}(B_j)}} - \frac{\widehat{\Delta}_{j,r}(\Psi_{n,j})}{\sqrt{\widehat{S}_{j,r}(\Psi_{n,j})}} = o_p(1).$$

Since $K_n \leq K_{\max}$ by Assumption 3.4 and p is fixed, a union bound over the finitely many pairs (j, r) gives

$$\max_{1\leq j\leq K_n} \max_{1\leq r\leq p} \left| \frac{\widehat{\Delta}_{j,r}(B_j)}{\sqrt{\widehat{S}_{j,r}(B_j)}} - \frac{\widehat{\Delta}_{j,r}(\Psi_{n,j})}{\sqrt{\widehat{S}_{j,r}(\Psi_{n,j})}} \right| = o_p(1). \tag{24}$$

Define

$$d_{\mathrm{SMD},j}(W_j) := d_{\mathrm{SMD}}\left(\widehat{\mu}_{1j}(W_j), \widehat{\mu}_{0j}(W_j), \widehat{V}_{1j}(W_j), \widehat{V}_{0j}(W_j)\right).$$

By the definition of d_{SMD} in Section 3.1,

$$d_{\mathrm{SMD},j}(W_j) = \frac{1}{\sqrt{p}} \left[\sum_{r=1}^{p} \left(\frac{\widehat{\Delta}_{j,r}(W_j)}{\sqrt{\widehat{S}_{j,r}(W_j)}} \right)^2 \right]^{1/2}.$$

For each subclass j, the reverse triangle inequality gives

$$|d_{\mathrm{SMD},j}(B_j) - d_{\mathrm{SMD},j}(\Psi_{n,j})| \leq \frac{1}{\sqrt{p}} \left[\sum_{r=1}^{p} \left(\frac{\widehat{\Delta}_{j,r}(B_j)}{\sqrt{\widehat{S}_{j,r}(B_j)}} - \frac{\widehat{\Delta}_{j,r}(\Psi_{n,j})}{\sqrt{\widehat{S}_{j,r}(\Psi_{n,j})}} \right)^2 \right]^{1/2}.$$

Because p is fixed, equation equation 24 implies

$$\max_{1\leq j\leq K_n} \left[\sum_{r=1}^{p} \left(\frac{\widehat{\Delta}_{j,r}(B_j)}{\sqrt{\widehat{S}_{j,r}(B_j)}} - \frac{\widehat{\Delta}_{j,r}(\Psi_{n,j})}{\sqrt{\widehat{S}_{j,r}(\Psi_{n,j})}} \right)^2 \right]^{1/2} = o_p(1). \tag{25}$$

Combining the preceding inequality with equation equation 25, we obtain

$$\max_{1\leq j\leq K_n} |d_{\mathrm{SMD},j}(B_j) - d_{\mathrm{SMD},j}(\Psi_{n,j})| = o_p(1). \tag{26}$$

A.8 Aggregation over Subclasses

Recall that the balance losses are finite sums over subclasses:

$$L_B = \sum_{j=1}^{K_n} d_{\mathrm{SMD},j}(B_j), \qquad L_\Psi = \sum_{j=1}^{K_n} d_{\mathrm{SMD},j}(\Psi_{n,j}).$$

Therefore, by the triangle inequality,

$$|L_B - L_\Psi| \leq \sum_{j=1}^{K_n} |d_{\mathrm{SMD},j}(B_j) - d_{\mathrm{SMD},j}(\Psi_{n,j})|.$$

Using the uniform bound in equation equation 26,we have

$$\sum_{j=1}^{K_n} |d_{\mathrm{SMD},j}(B_j) - d_{\mathrm{SMD},j}(\Psi_{n,j})| \leq K_n \max_{1\leq j\leq K_n} |d_{\mathrm{SMD},j}(B_j) - d_{\mathrm{SMD},j}(\Psi_{n,j})|.$$

By Assumption 3.4,

$$K_n \leq K_{\max}.$$

Thus,

$$|L_B - L_\Psi| \leq K_{\max} \max_{1\leq j\leq K_n} |d_{\mathrm{SMD},j}(B_j) - d_{\mathrm{SMD},j}(\Psi_{n,j})| = o_p(1).$$

Equivalently,

$$L_B - L_\Psi = o_p(1).$$

This completes the proof.

B PSEUDOCODE

Algorithm 1: Dynamic Programming for Equal-Weight Propensity-Score Partitioning

Require: Estimated propensity scores $\boldsymbol{e} = (e_1, \ldots, e_n)$; covariates $\boldsymbol{X} \in \mathbb{R}^{n \times p}$; treatment indicators $\boldsymbol{T} \in \{0,1\}^n$; number of subclasses K_n; minimum treated and control counts $m_{1,n}, m_{0,n}$; interval discrepancy function d_{SMD}

Ensure: Subclass centers $\{c_j\}_{j=1}^K$; half-widths $\{h_j\}_{j=1}^K$; cut points $\{g_j\}_{j=0}^K$ with $g_0 = 0$ and $g_K = 1$

Validate the dimensions of $\boldsymbol{e}$, $\boldsymbol{X}$, and $\boldsymbol{T}$.

Validate $1 \le K_n \le n$, $m_{1,n} \ge 1$, and $m_{0,n} \ge 1$.

Verify that $\boldsymbol{e} \in [0,1]^n$ and $\boldsymbol{T} \in \{0,1\}^n$.

Obtain a stable ordering

$$\pi \leftarrow \mathrm{argsort}(\boldsymbol{e}, \text{stable}).$$

Reorder the observations according to π:

$$e_{(i)} \leftarrow e_{\pi_i}, \qquad \boldsymbol{x}_{(i)} \leftarrow \boldsymbol{x}_{\pi_i}, \qquad t_{(i)} \leftarrow t_{\pi_i}.$$

Define the legal-gap indicator:

$$\mathrm{Legal}(r) \leftarrow \begin{cases} 1, & r = 0, \\ 1, & r = n, \\ \mathbb{I}\{e_{(r+1)} > e_{(r)}\}, & 1 \le r < n. \end{cases}$$

for $r = 1, \ldots, n$ **do**

Construct prefix sums:

$$P_T[r] \leftarrow \sum_{i=1}^{r} t_{(i)}, \qquad P_0[r] \leftarrow \sum_{i=1}^{r} (1 - t_{(i)}).$$

Construct covariate prefix sums:

$$P_{TX}[r] \leftarrow \sum_{i=1}^{r} t_{(i)} \boldsymbol{x}_{(i)},$$

$$P_{X0}[r] \leftarrow \sum_{i=1}^{r} (1 - t_{(i)}) \boldsymbol{x}_{(i)}.$$

Construct squared-covariate prefix sums:

$$P_{TX^2}[r] \leftarrow \sum_{i=1}^{r} t_{(i)} \boldsymbol{x}_{(i)}^{\odot 2},$$

$$P_{X0^2}[r] \leftarrow \sum_{i=1}^{r} (1 - t_{(i)}) \boldsymbol{x}_{(i)}^{\odot 2}.$$

end for

Set all prefix sums at index 0 to zero.

for $\ell = 0, \ldots, n-1$ **do**

for $r = \ell + 1, \ldots, n$ **do**

if $\mathrm{Legal}(\ell) = 1$ **and** $\mathrm{Legal}(r) = 1$ **then**

Compute treated and control counts:

$$N_1(\ell, r) \leftarrow P_T[r] - P_T[\ell],$$

$$N_0(\ell, r) \leftarrow P_0[r] - P_0[\ell].$$

if $N_1(\ell, r) \ge m_{1,n}$ **and** $N_0(\ell, r) \ge m_{0,n}$ **then**

Compute treated and control means:

$$\mu_1(\ell, r) \leftarrow \frac{P_{TX}[r] - P_{TX}[\ell]}{N_1(\ell, r)},$$

$$\mu_0(\ell, r) \leftarrow \frac{P_{X0}[r] - P_{X0}[\ell]}{N_0(\ell, r)}.$$

Compute coordinate-wise variances:

$$V_1(\ell, r) \leftarrow \frac{P_{TX^2}[r] - P_{TX^2}[\ell]}{N_1(\ell, r)} - \mu_1(\ell, r)^{\odot 2},$$

$$V_0(\ell, r) \leftarrow \frac{P_{X0^2}[r] - P_{X0^2}[\ell]}{N_0(\ell, r)} - \mu_0(\ell, r)^{\odot 2}.$$

Compute the interval cost:

$$C(\ell, r) \leftarrow d_{SMD}\big(\mu_1(\ell, r), \mu_0(\ell, r), V_1(\ell, r), V_0(\ell, r)\big).$$

else
  Set $C(\ell, r) \leftarrow +\infty$.
end if
else
  Set $C(\ell, r) \leftarrow +\infty$.
end if
end for
end for

Initialize the dynamic-programming table:

$$D[0, 0] \leftarrow 0, \qquad D[r, j] \leftarrow +\infty$$

for all other states.

Initialize all split pointers:

$$S[r, j] \leftarrow -1.$$

for $j = 1, \ldots, K_n$ **do**
  for $r = 1, \ldots, n$ **do**
    for $\ell = 0, \ldots, r - 1$ **do**
      if $D[\ell, j - 1] < +\infty$ **and** $C(\ell, r) < +\infty$ **then**

$$\text{candidate} \leftarrow D[\ell, j - 1] + C(\ell, r).$$

        if candidate $< D[r, j]$ **then**
          $D[r, j] \leftarrow$ candidate.
          $S[r, j] \leftarrow \ell$.
        end if
      end if
    end for
  end for
end for
if $D[n, K_n] = +\infty$ **then**
  return failure: no feasible partition into K_n subclasses exists.
end if

Recover the optimal ordered boundaries:

$$b_K \leftarrow n.$$

for $j = K_n, K_n - 1, \ldots, 1$ **do**
  $b_{j-1} \leftarrow S[b_j, j]$.
end for

The recovered boundaries satisfy

$$0 = b_0 < b_1 < \cdots < b_K = n.$$

for $j = 1, \ldots, K_n$ **do**

Define the propensity-score values in subclass j:

$$\mathcal{E}_j \leftarrow \left\{e_{(b_{j-1}+1)}, \ldots, e_{(b_j)}\right\}.$$

Compute the subclass center:

$$c_j \leftarrow \frac{\frac{e_{b_j}+e_{b_j+1}}{2} + \frac{e_{b_{j-1}}+e_{b_{j-1}+1}}{2}}{2}$$

Compute the subclass half-width:

$$h_j \leftarrow \frac{\frac{e_{b_j}+e_{b_j+1}}{2} - \frac{e_{b_{j-1}}+e_{b_{j-1}+1}}{2}}{2}$$

end for

Set the two outer propensity-score boundaries:

$$g_0 \leftarrow 0, \qquad g_K \leftarrow 1.$$

for $j = 1, \ldots, K_n - 1$ **do**

Compute the internal cut point:

$$g_j \leftarrow \frac{e_{(b_j)} + e_{(b_j+1)}}{2}.$$

end for

return $\{c_j\}_{j=1}^{K_n}$, $\{h_j\}_{j=1}^{K_n}$, and $\{g_j\}_{j=0}^{K_n}$.

Algorithm 2: Iterative Training with Propensity Subclassification

Require: Training samples $\{(X_i, T_i)\}_{i=1}^{n}$, number of subclasses K_n, balancing-loss weight λ_ψ, number of inner optimization steps N_{in}, maximum number of outer iterations $\mathrm{MaxIter}$

Ensure: Trained propensity network, optimal interval boundaries $\mathcal{G} = \{g_0, \ldots, g_K\}$, subclass centers $\{c_j\}_{j=1}^{K}$, and bandwidths $\{h_j\}_{j=1}^{K}$

Initialize the propensity network parameters θ

Initialize a feasible set of subclass boundaries $\mathcal{G}^{(0)} = \{g_0^{(0)}, \ldots, g_K^{(0)}\}$

Set $t \leftarrow 0$

while $t < \mathrm{MaxIter}$ **do**

Step 1: Network optimization with fixed boundaries

for $s = 1, \ldots, N_{\text{in}}$ **do**

Compute the base loss $L_{\text{base}}(\theta)$ and the subclassification balancing loss $L_\psi(\theta; \mathcal{G}^{(t)})$ using the current boundaries

Compute the augmented loss

$$L_{\text{aug}}\left(\theta; \mathcal{G}^{(t)}\right) = L_{\text{base}}(\theta) + \lambda_\psi L_\psi\left(\theta; \mathcal{G}^{(t)}\right)$$

Update the network parameters by one gradient step:

$$\theta \leftarrow \theta - \eta_\theta \nabla_\theta L_{\text{aug}}\left(\theta; \mathcal{G}^{(t)}\right)$$

end for

Step 2: Subclassification update with fixed network

Compute the estimated propensity scores $\widehat{e}_i = \widehat{e}_\theta(X_i)$ using the updated network

Obtain new index boundaries $\widetilde{\mathcal{G}}^{(t+1)}$ by Dynamic Programming Algorithm

Set $\mathcal{G}^{(t+1)} \leftarrow \widetilde{\mathcal{G}}^{(t+1)}$

Set $t \leftarrow t + 1$

if $\mathcal{G}^{(t)} = \mathcal{G}^{(t-1)}$ **then**

break

end if

end while

Compute the final subclass centers $\{c_j\}_{j=1}^{K_n}$ and bandwidths $\{h_j\}_{j=1}^{K_n}$ from the final boundaries $\mathcal{G}^{(t)}$

return θ, $\mathcal{G}^{(t)}$, $\{c_j\}_{j=1}^{K_n}$, and $\{h_j\}_{j=1}^{K_n}$

C Function Constructions And Verification of Assumptions

This appendix presents six differentiable constructions for smooth propensity-score subclassification. Section C.1 defines the six smooth subclassification functions. Section C.2 discusses their basic analytical properties. Section C.3 examines conditions under which Assumption 3.3 may hold for these constructions. Section C.4 discusses the remaining assumptions used in Theorem 3.1.

C.1 Construction of Smooth Subclassification Functions

Let $c_{n,j}$ denote the center of the jth subclass and let $h_{n,j} > 0$ denote its half-width on the fitted propensity-score scale. Let

$$\widehat{e}(\psi_{n,j}(X))$$

denote the fitted propensity score associated with the representation $\psi_{n,j}(X)$ and network parameters θ_n. The corresponding hard subclassification indicator is

$$B_j(X) = \mathbf{1}\left\{|\widehat{e}(\psi_{n,j}(X)) - c_{n,j}| \le h_{n,j}\right\}. \tag{27}$$

Let $\delta_n > 0$ be a smoothing parameter satisfying

$$\delta_n \longrightarrow 0. \tag{28}$$

For the first three constructions, define

$$z_{n,j}(X) = \frac{h_{n,j} - |\widehat{e}(\psi_{n,j}(X)) - c_{n,j}|}{\delta_n}. \tag{29}$$

The first construction is based on the sigmoid function:

$$\Psi_{n,j}^{\text{sigmoid}}(X) = \frac{1}{1 + \exp\{-z_{n,j}(X)\}}. \tag{30}$$

The second construction uses a rescaled arctangent function:

$$\Psi_{n,j}^{\arctan}(X) = \frac{1}{2} + \frac{1}{\pi}\arctan\left(z_{n,j}(X)\right). \tag{31}$$

The third construction is based on the hyperbolic tangent function:

$$\Psi_{n,j}^{\tanh}(X) = \frac{1}{2}\left[1 + \tanh\left(z_{n,j}(X)\right)\right]. \tag{32}$$

For the remaining three constructions, define the left and right endpoints of the jth subclass by

$$\ell_{n,j} = c_{n,j} - h_{n,j}, \qquad r_{n,j} = c_{n,j} + h_{n,j}. \tag{33}$$

The fourth construction is obtained by taking the difference between two logistic cumulative distribution functions:

$$\Psi_{n,j}^{\text{logistic}}(X) = \sigma\left(\frac{\widehat{e}(\psi_{n,j}(X)) - \ell_{n,j}}{\delta_n}\right) - \sigma\left(\frac{\widehat{e}(\psi_{n,j}(X)) - r_{n,j}}{\delta_n}\right), \tag{34}$$

where

$$\sigma(u) = \frac{1}{1 + \exp(-u)} \tag{35}$$

denotes the logistic sigmoid function.

The fifth construction uses the difference between two hyperbolic tangent transitions:

$$\Psi_{n,j}^{\text{two-tanh}}(X) = \frac{1}{2}\left[\tanh\left(\frac{\widehat{e}(\psi_{n,j}(X)) - \ell_{n,j}}{\delta_n}\right) - \tanh\left(\frac{\widehat{e}(\psi_{n,j}(X)) - r_{n,j}}{\delta_n}\right)\right]. \tag{36}$$

Finally, define the cubic smoothstep function

$$\text{SmoothStep}(u) = \widetilde{u}^{\,2}\left(3 - 2\widetilde{u}\right), \qquad \widetilde{u} = \min\{1, \max\{0, u\}\}. \tag{37}$$

The sixth construction combines two smoothstep transitions:

$$\begin{aligned}\Psi_{n,j}^{\text{smoothstep}}(X) = \text{SmoothStep}&\left(\frac{\widehat{e}(\psi_{n,j}(X)) - \ell_{n,j} + \delta_n}{2\delta_n}\right) \\ &\times \text{SmoothStep}\left(\frac{r_{n,j} - \widehat{e}(\psi_{n,j}(X)) + \delta_n}{2\delta_n}\right).\end{aligned} \tag{38}$$

For convenience, we use the notation

$$\Psi_{n,j}(X) \in \left\{\Psi_{n,j}^{\text{sigmoid}}(X), \Psi_{n,j}^{\arctan}(X), \Psi_{n,j}^{\tanh}(X), \Psi_{n,j}^{\text{logistic}}(X), \Psi_{n,j}^{\text{two-tanh}}(X), \Psi_{n,j}^{\text{smoothstep}}(X)\right\} \tag{39}$$

when discussing a generic smooth subclassification weight. The first three constructions express subclass membership through a single scaled signed distance in equation 29. The latter three constructions explicitly smooth the two endpoints of the subclass interval defined in equation 33.

C.2 Boundary Localization of the Hard–Smooth Discrepancy

Let

$$I_{n,j} = [\ell_{n,j}, r_{n,j}] = [c_{n,j} - h_{n,j}, c_{n,j} + h_{n,j}]$$

denote the jth propensity-score subclass. For a propensity-score value $u \in [0,1]$, define the distance to the nearest subclass boundary by

$$d_{n,j}(u) = \min\{|u - \ell_{n,j}|,\ |u - r_{n,j}|\}.$$

The purpose of this section is to characterize the region in which the hard indicator and its smooth approximation can differ by more than a prescribed amount.

Throughout this section, let $\varepsilon \in (0, 1/2)$ be fixed. For each smooth construction, we derive a quantity $a_{n,j}^{(k)}(\varepsilon)$ such that

$$\left\{\left|B_{n,j}(x) - \Psi_{n,j}^{(k)}(x)\right| > \varepsilon\right\} \subseteq \left\{d_{n,j}(u_n(x)) < a_{n,j}^{(k)}(\varepsilon)\right\}. \tag{C.1}$$

Thus, a non-negligible hard–smooth discrepancy can only occur inside shrinking neighborhoods of the subclass boundaries.

Single-transition constructions. For the sigmoid, arctangent, and hyperbolic-tangent constructions, define

$$q_{n,j}(x) = h_{n,j} - |u_n(x) - c_{n,j}|\,.$$

Notice that

$$q_{n,j}(x) = \begin{cases} d_{n,j}(u_n(x)), & u_n(x) \in I_{n,j}, \\ -d_{n,j}(u_n(x)), & u_n(x) \notin I_{n,j}. \end{cases} \tag{C.2}$$

Consequently, the hard indicator can be written as

$$B_{n,j}(x) = \mathbf{1}\{q_{n,j}(x) \geq 0\}.$$

For the sigmoid construction,

$$\Psi_{n,j}^{\text{sigmoid}}(x) = \sigma\left(\frac{q_{n,j}(x)}{\delta_n}\right), \qquad \sigma(v) = \frac{1}{1+\exp(-v)}.$$

If $q_{n,j}(x) \geq 0$, then

$$\left|B_{n,j}(x) - \Psi_{n,j}^{\text{sigmoid}}(x)\right| = 1 - \sigma\left(\frac{q_{n,j}(x)}{\delta_n}\right).$$

If $q_{n,j}(x) < 0$, then

$$\left|B_{n,j}(x) - \Psi_{n,j}^{\text{sigmoid}}(x)\right| = \sigma\left(\frac{q_{n,j}(x)}{\delta_n}\right).$$

Therefore,

$$\left\{\left|B_{n,j}(x) - \Psi_{n,j}^{\text{sigmoid}}(x)\right| > \varepsilon\right\} \subseteq \left\{d_{n,j}(u_n(x)) < a_{n,j}^{\text{sigmoid}}(\varepsilon)\right\},$$

where

$$a_{n,j}^{\text{sigmoid}}(\varepsilon) = \delta_n \log\left(\frac{1-\varepsilon}{\varepsilon}\right). \tag{C.3}$$

For this single-transition construction, the above boundary-neighborhood characterization is exact up to the treatment of the boundary itself.

For the arctangent construction,

$$\Psi_{n,j}^{\arctan}(x) = \frac{1}{2} + \frac{1}{\pi}\arctan\left(\frac{q_{n,j}(x)}{\delta_n}\right).$$

The inverse of the transition function satisfies

$$\left(\Psi_{n,j}^{\arctan}(x) = 1 - \varepsilon\right) \iff \frac{q_{n,j}(x)}{\delta_n} = \tan\left(\frac{\pi}{2} - \pi\varepsilon\right).$$

It follows that

$$\left\{\left|B_{n,j}(x) - \Psi_{n,j}^{\arctan}(x)\right| > \varepsilon\right\} \subseteq \left\{d_{n,j}(u_n(x)) < a_{n,j}^{\arctan}(\varepsilon)\right\},$$

where

$$a_{n,j}^{\arctan}(\varepsilon) = \delta_n \tan\left(\frac{\pi}{2} - \pi\varepsilon\right) = \delta_n \cot(\pi\varepsilon). \tag{C.4}$$

For the hyperbolic-tangent construction,

$$\Psi_{n,j}^{\tanh}(x) = \frac{1}{2}\left[1 + \tanh\left(\frac{q_{n,j}(x)}{\delta_n}\right)\right].$$

Since

$$\frac{1}{2}\left[1 + \tanh(v)\right] = 1 - \varepsilon$$

is equivalent to

$$v = \operatorname{arctanh}(1 - 2\varepsilon),$$

we obtain

$$\left\{\left|B_{n,j}(x) - \Psi_{n,j}^{\tanh}(x)\right| > \varepsilon\right\} \subseteq \left\{d_{n,j}(u_n(x)) < a_{n,j}^{\tanh}(\varepsilon)\right\},$$

where

$$a_{n,j}^{\tanh}(\varepsilon) = \delta_n \operatorname{arctanh}(1 - 2\varepsilon). \tag{C.5}$$

Thus, for the first three constructions, the relevant boundary-neighborhood width is exactly of order

$$a_{n,j}^{(k)}(\varepsilon) = O(\delta_n).$$

Two-sided logistic construction. For the two-sided logistic construction, write

$$\Psi_{n,j}^{\text{logistic}}(x) = \sigma\left(\frac{u_n(x) - \ell_{n,j}}{\delta_n}\right) - \sigma\left(\frac{u_n(x) - r_{n,j}}{\delta_n}\right).$$

Because the second term is nonnegative, for a point to the left of the interval,

$$u_n(x) < \ell_{n,j},$$

we have

$$\Psi_{n,j}^{\text{logistic}}(x) \le \sigma\left(\frac{u_n(x) - \ell_{n,j}}{\delta_n}\right).$$

Hence,

$$\Psi_{n,j}^{\text{logistic}}(x) > \varepsilon \quad \Longrightarrow \quad u_n(x) - \ell_{n,j} > -\delta_n \log\left(\frac{1-\varepsilon}{\varepsilon}\right).$$

The analogous relation holds to the right of the interval. For points inside the interval, the error satisfies

$$1 - \Psi_{n,j}^{\text{logistic}}(x) \ge 1 - \sigma\left(\frac{u_n(x) - \ell_{n,j}}{\delta_n}\right)$$

near the left endpoint, with an analogous inequality near the right endpoint. Therefore,

$$\left\{\left|B_{n,j}(x) - \Psi_{n,j}^{\text{logistic}}(x)\right| > \varepsilon\right\} \subseteq \left\{d_{n,j}(u_n(x)) < a_{n,j}^{\text{logistic}}(\varepsilon)\right\},$$

where one may take

$$a_{n,j}^{\text{logistic}}(\varepsilon) = \delta_n \log\left(\frac{1-\varepsilon}{\varepsilon}\right). \tag{C.6}$$

This is a boundary-neighborhood upper bound. When the subclass width satisfies

$$\frac{r_{n,j} - \ell_{n,j}}{\delta_n} \longrightarrow \infty,$$

the two endpoint transitions become asymptotically separated, and the bound describes the actual transition width at each endpoint up to a vanishing interaction between the two transitions.

Two-sided hyperbolic-tangent construction. For the two-sided hyperbolic-tangent construction,

$$\Psi_{n,j}^{\text{two-tanh}}(x) = \frac{1}{2}\left[\tanh\left(\frac{u_n(x)-\ell_{n,j}}{\delta_n}\right) - \tanh\left(\frac{u_n(x)-r_{n,j}}{\delta_n}\right)\right].$$

Near the left endpoint, the contribution of the left transition is controlled by

$$\frac{1}{2}\left[1+\tanh\left(\frac{u_n(x)-\ell_{n,j}}{\delta_n}\right)\right],$$

and the right endpoint is treated analogously. Using

$$\frac{1}{2}[1+\tanh(v)] = \varepsilon \quad \Longleftrightarrow \quad v = -\operatorname{arctanh}(1-2\varepsilon),$$

one obtains the boundary-neighborhood bound

$$\left\{\left|B_{n,j}(x) - \Psi_{n,j}^{\text{two-tanh}}(x)\right| > \varepsilon\right\} \subseteq \left\{d_{n,j}(u_n(x)) < a_{n,j}^{\text{two-tanh}}(\varepsilon)\right\},$$

where

$$a_{n,j}^{\text{two-tanh}}(\varepsilon) = \delta_n \operatorname{arctanh}(1-2\varepsilon). \tag{C.7}$$

As with the two-sided logistic construction, this is a uniform boundary-neighborhood bound. If the subclass width is large relative to δ_n, the two endpoint transitions are asymptotically separated.

Smoothstep construction. Let

$$S(v) = v^2(3-2v), \qquad v \in [0,1],$$

denote the cubic smoothstep function. The smoothstep construction is

$$\Psi_{n,j}^{\text{smoothstep}}(x) = S\left(\frac{u_n(x)-\ell_{n,j}+\delta_n}{2\delta_n}\right) S\left(\frac{r_{n,j}-u_n(x)+\delta_n}{2\delta_n}\right).$$

Let

$$v_\varepsilon^{\text{in}} = S^{-1}\left(\sqrt{1-\varepsilon}\right), \qquad v_\varepsilon^{\text{out}} = S^{-1}(\varepsilon),$$

where S^{-1} denotes the inverse of S on $[0,1]$. Define

$$a_{n,j}^{\text{smoothstep}}(\varepsilon) = \delta_n \max\left\{2v_\varepsilon^{\text{in}} - 1,\, 1 - 2v_\varepsilon^{\text{out}}\right\}. \tag{C.8}$$

For an interior point, if its distance from both endpoints is at least

$$\delta_n(2v_\varepsilon^{\text{in}} - 1),$$

then each of the two smoothstep factors is at least $\sqrt{1-\varepsilon}$, and therefore their product is at least $1-\varepsilon$. For an exterior point, if its distance from the nearest endpoint is at least

$$\delta_n(1 - 2v_\varepsilon^{\text{out}}),$$

then the corresponding transition factor is at most ε, so the smooth weight is at most ε. Consequently,

$$\left\{\left|B_{n,j}(x) - \Psi_{n,j}^{\text{smoothstep}}(x)\right| > \varepsilon\right\} \subseteq \left\{d_{n,j}(u_n(x)) < a_{n,j}^{\text{smoothstep}}(\varepsilon)\right\}. \tag{C.9}$$

Since $v_\varepsilon^{\text{in}}$ and $v_\varepsilon^{\text{out}}$ depend only on ε, we again have

$$a_{n,j}^{\text{smoothstep}}(\varepsilon) = O(\delta_n).$$

Summary. For all six smooth constructions, there exists a constant $A^{(k)}(\varepsilon) < \infty$, depending only on the construction and on ε, such that

$$a_{n,j}^{(k)}(\varepsilon) \le A^{(k)}(\varepsilon)\delta_n.$$

Therefore,

$$\left\{\left|B_{n,j}(x) - \Psi_{n,j}^{(k)}(x)\right| > \varepsilon\right\} \subseteq \left\{d_{n,j}(u_n(x)) \le A^{(k)}(\varepsilon)\delta_n\right\}. \tag{C.10}$$

Equivalently, the set on which the hard and smooth subclassification weights differ by more than ε is contained in the union of two intervals around the subclass endpoints:

$$\begin{aligned}\left\{\left|B_{n,j}(x) - \Psi_{n,j}^{(k)}(x)\right| > \varepsilon\right\} \subseteq\ & \left\{u_n(x) \in [\ell_{n,j} - A^{(k)}(\varepsilon)\delta_n, \ell_{n,j} + A^{(k)}(\varepsilon)\delta_n]\right\} \\ & \cup \left\{u_n(x) \in [r_{n,j} - A^{(k)}(\varepsilon)\delta_n, r_{n,j} + A^{(k)}(\varepsilon)\delta_n]\right\}.\end{aligned} \tag{C.11}$$

The total length of these two boundary neighborhoods is therefore at most

$$4A^{(k)}(\varepsilon)\delta_n,$$

unless the neighborhoods overlap or extend beyond the propensity-score domain $[0,1]$. This calculation shows that the relevant localization scale is δ_n, rather than a fixed positive margin.

C.3 Discussion of Assumption 3.3

The boundary-localization result in Section C.2 gives a direct interpretation of Assumption 3.3. For each smooth construction

$$k \in \{\text{sigmoid}, \arctan, \tanh, \text{logistic}, \text{two-tanh}, \text{smoothstep}\}$$

and every fixed $\varepsilon \in (0, 1/2)$, there exists a finite constant $A^{(k)}(\varepsilon)$ such that

$$\left\{\left|B_{n,j}(X_i) - \Psi_{n,j}^{(k)}(X_i)\right| > \varepsilon\right\} \subseteq \left\{d_{n,j}\big(u_n(X_i)\big) \le A^{(k)}(\varepsilon)\delta_n\right\},$$

where

$$u_n(X_i) = \widehat{e}(\psi(X_i))$$

is the fitted propensity score and

$$d_{n,j}(u) = \text{dist}\left(u, \{\ell_{n,j}, r_{n,j}\}\right)$$

denotes the distance from u to the nearest boundary of the jth subclass. Therefore,

$$\begin{aligned}&\frac{1}{n}\sum_{i=1}^{n}\Pr\left(\left|B_{n,j}(X_i) - \Psi_{n,j}^{(k)}(X_i)\right| > \varepsilon\right)\\ &\qquad\le \frac{1}{n}\sum_{i=1}^{n}\Pr\left(d_{n,j}\big(u_n(X_i)\big) \le A^{(k)}(\varepsilon)\delta_n\right).\end{aligned}$$

Hence, under the smooth constructions considered in Section C.2, Assumption 3.3 is implied by the condition

$$\frac{1}{n}\sum_{i=1}^{n}\Pr\left(d_{n,j}\big(u_n(X_i)\big) \le A^{(k)}(\varepsilon)\delta_n\right) \longrightarrow 0.$$

In other words, for a fixed tolerance ε, a substantial discrepancy between the hard and smooth subclassification weights can occur only when the fitted propensity score lies within an $A^{(k)}(\varepsilon)\delta_n$-neighborhood of a subclass boundary.

The condition above is an additional non-verifiable regularity condition in the present framework. The fitted propensity scores depend on the estimated neural network, while the subclass boundaries are selected by the dynamic-programming procedure from the same sample. Consequently, both the propensity-score distribution and the subclass boundaries are random and data-dependent. Establishing the vanishing-boundary-mass condition from primitive assumptions would require a joint analysis of these two data-adaptive components. The algorithm does not automatically guarantee the condition, and it cannot be verified solely from the definitions of the six smooth functions. We therefore retain Assumption 3.3 explicitly as an additional regularity condition.

Nevertheless, the transformed condition is plausible under the following considerations.

(1) Continuous propensity-score distributions typically do not concentrate at a single boundary. Suppose that the distribution of the fitted propensity score is locally regular around the subclass boundaries and does not assign a point mass to any individual boundary. For example, if the fitted propensity-score distribution has a locally bounded density, then the probability mass of a neighborhood with radius r_n around a boundary is typically of order $O(r_n)$. Taking

$$r_n = A^{(k)}(\varepsilon)\delta_n$$

and using $\delta_n \to 0$ suggests that

$$\Pr\left(d_{n,j}\big(u_n(X_i)\big) \le A^{(k)}(\varepsilon)\delta_n\right) \longrightarrow 0.$$

Thus, only a vanishing fraction of observations is expected to fall into the region where the hard and smooth weights differ by more than ε. This argument is motivational rather than a formal proof, because the fitted propensity-score distribution and the subclass boundaries are both sample-dependent.

(2) Dynamic-programming boundaries are selected between ordered observations. The dynamic-programming procedure first sorts the observations according to their fitted propensity scores and restricts each subclass to be a contiguous interval in this ordering. Internal boundaries are selected at legal gaps between adjacent ordered propensity-score values. In addition, observations with identical propensity scores are not separated. Therefore, the boundaries are constructed between observed score values rather than being intentionally placed through the middle of an observed tied group.

This construction supports the plausibility of the vanishing-boundary-mass condition. Section C.2 shows that smoothing can alter the hard membership decision only within a neighborhood whose width is proportional to δ_n. If the local spacing and concentration of the ordered fitted propensity scores are sufficiently regular, the number of observations falling within the $A^{(k)}(\varepsilon)\delta_n$-neighborhoods of the selected boundaries should be negligible relative to n.

This reasoning does not provide a formal guarantee. Since the dynamic-programming boundaries are selected adaptively, the procedure may in principle choose a boundary in a region with unusually high local concentration of fitted propensity scores. Controlling this data-adaptive behavior would require additional assumptions and a separate analysis of the partition-selection procedure.

In summary, Section C.2 reduces Assumption 3.3 to a vanishing-mass requirement for neighborhoods of the subclass boundaries. The relevant neighborhood width is of order

$$A^{(k)}(\varepsilon)\delta_n.$$

The condition is statistically plausible when the fitted propensity-score distribution does not concentrate at individual boundaries and when the ordered score gaps selected by dynamic programming are sufficiently regular. However, the condition is neither automatically implied by the algorithm nor currently verifiable under the proposed framework. It is therefore retained as an explicit additional regularity condition for the asymptotic equivalence result.

C.4 Discussion of the Remaining Assumptions

Assumption 3.1. This assumption is generally not directly verifiable from the observed data, since its validity is determined mainly by the data-generation and sampling procedures. The observations are required to be mutually independent, but they need not be identically distributed. This flexibility allows the framework to handle pooled data from different populations, sites, cohorts, or time periods, provided that the individual observations remain independent.

Assumption 3.2. This moment condition is also generally not directly verifiable from a finite sample. Nevertheless, it is satisfied by many common distributions, including multivariate normal, bounded, uniform, beta, sub-Gaussian, and sub-exponential distributions, when their parameters are uniformly controlled. It can also hold for certain heavy-tailed distributions; for example, a Student's t distribution with degrees of freedom ν satisfies the condition whenever

$$\nu > 2 + \eta.$$

Thus, the assumption does not require bounded or normally distributed covariates; it only requires a uniformly bounded moment of order strictly greater than two.

Assumption 3.4. This condition can be guaranteed directly by the algorithm's hyperparameter.

Assumption 3.5. This condition is supported by the minimum treated and control sample-size constraints used in the dynamic-programming procedure. In particular, suppose there exists a constant $\delta > 0$ such that, for all sufficiently large n,

$$\frac{m_{1,n}}{n} \geq \delta_1, \qquad \frac{m_{0,n}}{n} \geq \delta_0.$$

Then every selected subclass is required to contain a non-negligible fraction of both treated and control observations.

Assumption 3.6. Assumption 3.6 imposes a uniform lower bound on the within-subclass variance used to standardize the covariate mean differences. Its primary role is to prevent the denominator of the within-subclass standardized mean difference from becoming arbitrarily small. Without this condition, a covariate may have nearly zero within-subclass variation in one or both treatment groups, causing the standardized mean difference to become unstable or poorly defined even when the corresponding raw mean difference is small. Thus, Assumption 3.6 should be interpreted as

a non-degeneracy condition for the SMD normalization rather than as a condition required for the dynamic-programming algorithm itself. In applications where a covariate is nearly constant within a subclass, alternative normalization or preprocessing strategies may be more appropriate.
Overall, Assumptions 3.1 and 3.2 are sampling and distributional regularity conditions, Assumption 3.4 can be enforced by fixing K, Assumption 3.5 is practically supported by requiring the minimum treated and control counts to remain a positive fraction of the sample size, and Assumption 3.6 is a necessary regular condition for using our model.

D Detailed Experimental Results

The box plots mentioned in Section 4.2 are presented in Figure 2–5.
The hyperparameter search described in Section 4.3 is presented in Table 4–6. Furthermore, we conduct experiments to evaluate the choice of smoothing functions in our model. The results show that the sigmoid smoothing function yields the best performance for BCAUSSWithSubclassification, whereas the Smoothstep smoothing function performs best for Dragonnet.

Table 4: Block-search results for BCAUSS under different values of K and δ_n. Each entry reports the mean $\pm$ standard deviation.

K	$\delta_n = n^{-1/3}$	$\delta_n = n^{-2/3}$	$\delta_n = -1$
5	ATE bias: 0.145933 ± 0.131654 PEHE: 1.987641 ± 1.002353	ATE bias: 0.186877 ± 0.148319 PEHE: 2.149408 ± 1.340202	ATE bias: 0.504771 ± 0.608452 PEHE: 2.240223 ± 1.032355
7	ATE bias: 0.485132 ± 0.996010 PEHE: 2.244580 ± 1.096177	ATE bias: 0.212654 ± 0.193597 PEHE: 1.899649 ± 0.862774	ATE bias: 0.335226 ± 0.292757 PEHE: 2.119144 ± 1.156637
10	ATE bias: 0.257223 ± 0.207985 PEHE: 2.029834 ± 0.780217	ATE bias: 0.211387 ± 0.135928 PEHE: 2.195603 ± 1.353895	ATE bias: 0.199254 ± 0.220745 PEHE: 2.204774 ± 1.131516

Table 5: Block-search results for Dragonnet under different values of K and δ_n. Each entry reports the mean $\pm$ standard deviation.

K	$\delta_n = n^{-1/3}$	$\delta_n = n^{-2/3}$	$\delta_n = -1$
5	ATE bias: 0.260353 ± 0.201454 PEHE: 1.747837 ± 0.337163	ATE bias: 0.232004 ± 0.149098 PEHE: 1.927381 ± 0.839913	ATE bias: 0.200235 ± 0.201813 PEHE: 2.030889 ± 1.153945
7	ATE bias: 0.219734 ± 0.240842 PEHE: 1.745533 ± 0.267933	ATE bias: 0.168655 ± 0.133973 PEHE: 2.000582 ± 1.059514	ATE bias: 0.215854 ± 0.174949 PEHE: 2.082777 ± 0.944272
10	ATE bias: 0.157893 ± 0.107611 PEHE: 1.529875 ± 0.028363	ATE bias: 0.273964 ± 0.189994 PEHE: 1.663035 ± 0.116128	ATE bias: 0.282512 ± 0.160411 PEHE: 2.058292 ± 0.947559

Table 6: Functional search results for smoothing functions. BCUS and DGWS are abbreviations for BCAUSSWithSubclassification and DragonnetWithSubclassification, respectively. Each entry reports the mean $\pm$ standard deviation.

(a)

Function	BCUS ATE bias	BCUS PEHE
arctan	0.306946 ± 0.209097	2.051886 ± 0.969362
tanh	0.247905 ± 0.248032	2.042645 ± 0.888173
sigmoid	0.163909 ± 0.175992	1.938759 ± 0.623262
logistic	0.225739 ± 0.179666	2.016229 ± 0.883591
two_tanh	0.252645 ± 0.241781	2.009226 ± 0.931307
smoothstep	0.275462 ± 0.196939	1.909316 ± 0.755951

(b)

Function	DGWS ATE bias	DGWS PEHE
arctan	0.335817 ± 0.277178	2.021522 ± 1.044047
tanh	0.332459 ± 0.503890	1.991534 ± 0.994272
sigmoid	0.327349 ± 0.325152	2.038872 ± 0.887086
logistic	0.297685 ± 0.339553	1.736123 ± 0.409928
two_tanh	0.297196 ± 0.253762	2.185803 ± 1.146882
smoothstep	0.092287 ± 0.082184	1.678614 ± 0.177396

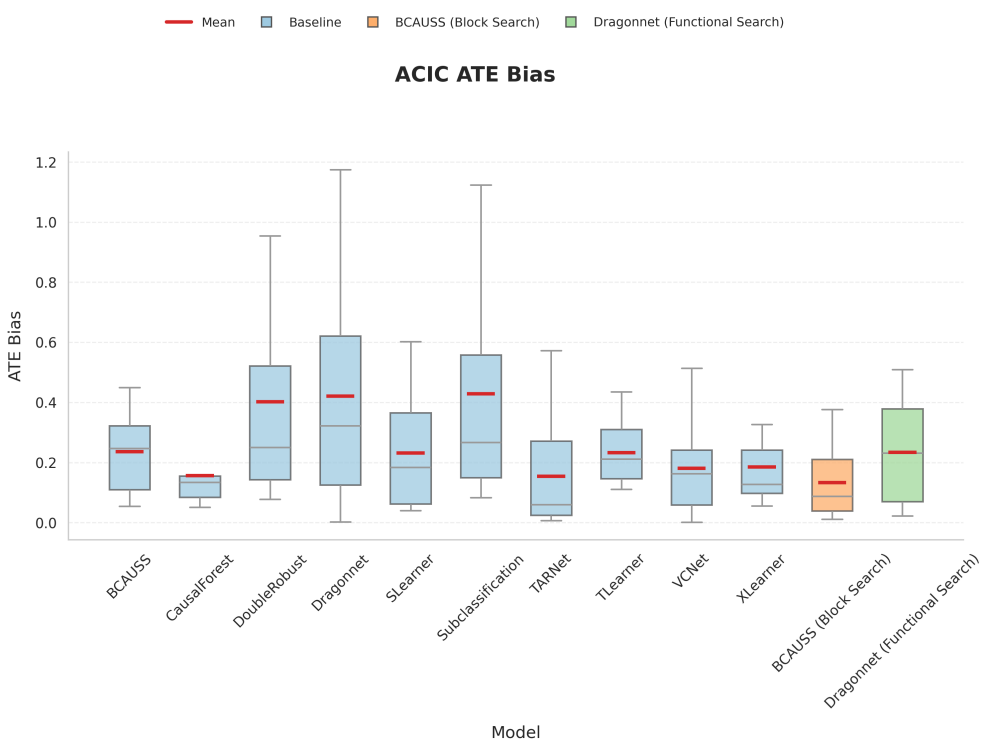


Figure 2: ACIC ATE bias box plots across experimental runs.

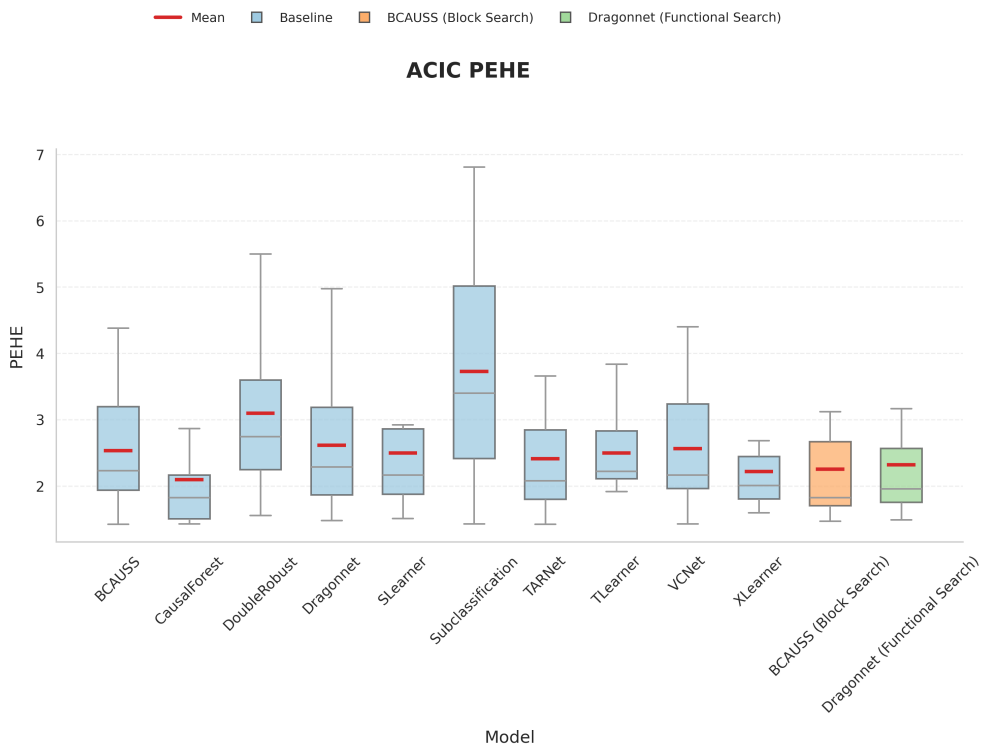


Figure 3: ACIC PEHE box plots across experimental runs.

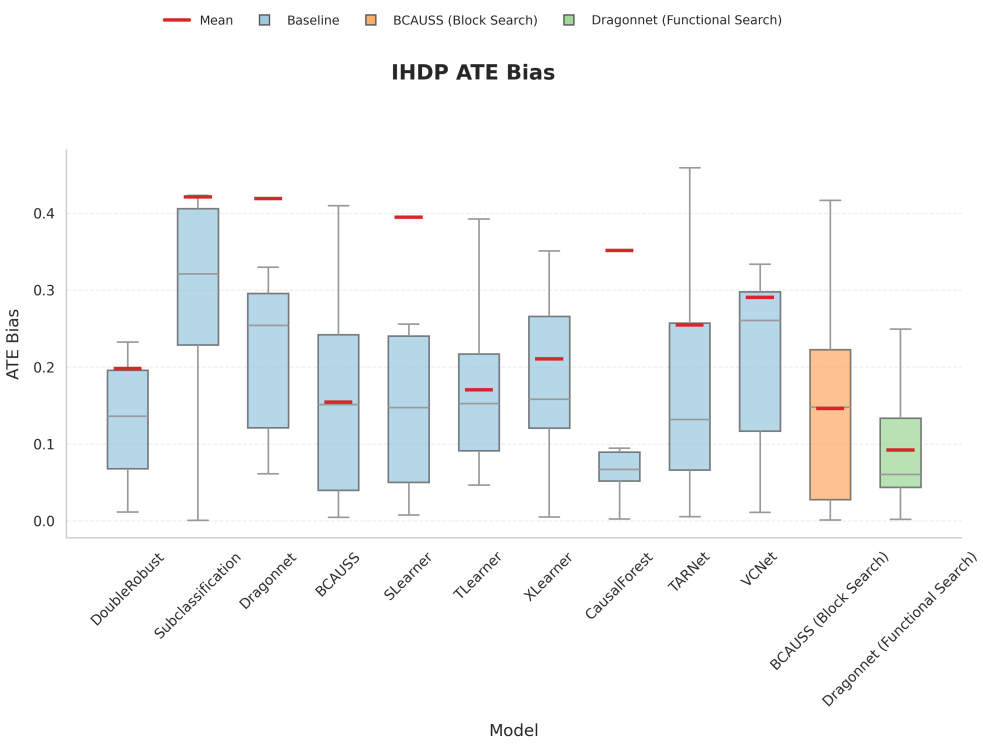


Figure 4: IHDP ATE bias box plots across experimental runs.

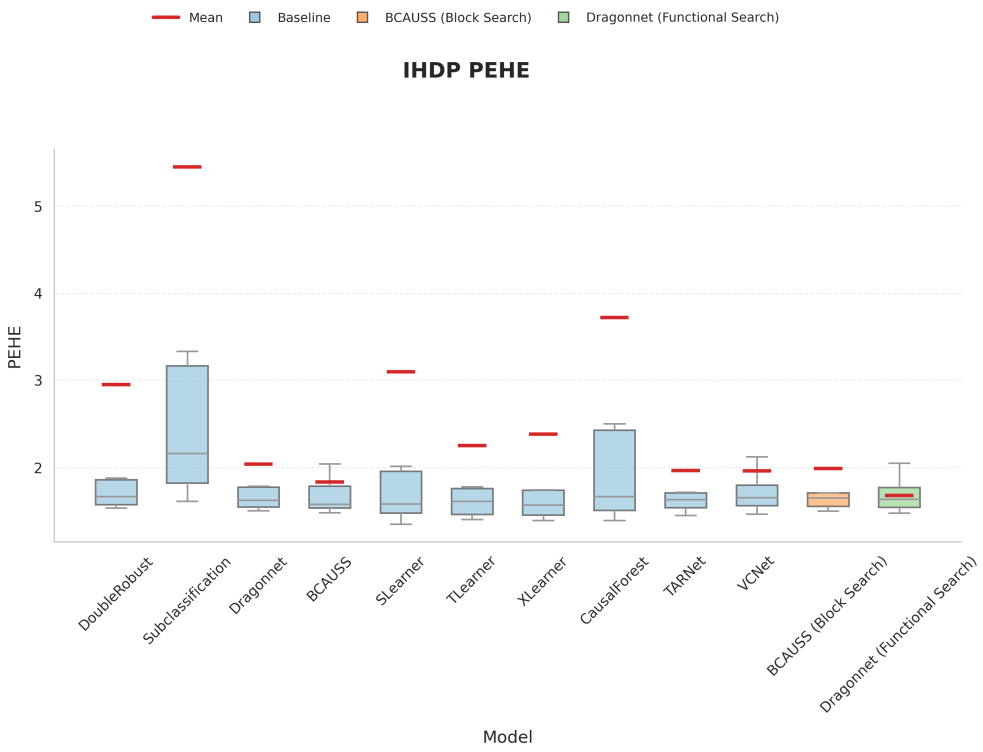


Figure 5: IHDP PEHE box plots across experimental runs.