Journal of Machine Learning Research 27 (2026) 1-53 Submitted 9/25; Revised 1/26; Published 2/26

Doubly Debiased Robust Subsampling for Transfer Learning

Tao Wang* TAOW@UVIC.CA
Department of Economics
Department of Mathematics and Statistics (by courtesy)
University of Victoria
Victoria, BC V8W 2Y2, Canada
* *Corresponding Author*

Weng Kee Wong WKWONG@UCLA.EDU
Department of Biostatistics
University of California
Los Angeles, CA 90095, USA

Editor: Weijie Su

Abstract

This paper develops a general framework for doubly debiased robust subsampling for transfer learning. The setting arises when massive source datasets are computationally infeasible to use in full, while naive or heuristic subsampling leads to biased estimators that further inherit transfer bias under source-target distributional shifts. We resolve these challenges through two complementary debiasing mechanisms. Inverse probability weighting removes subsampling bias by ensuring that subsample-based estimators represent the full source distribution, while a target-based one-step refinement recenters estimators towards the target distribution, thereby mitigating transfer bias. These corrections are embedded within a distributionally robust optimization design that simultaneously controls worst-case target risk and enforces source-target alignment through maximum mean discrepancy. To optimize subsampling distributions, we propose a scalarized particle swarm algorithm that efficiently explores the robustness-alignment frontier by adjusting a single tuning parameter. We establish theoretical properties, including asymptotic normality, generalization bounds, oracle inequalities, and minimax optimality under distributional uncertainty. Simulation studies and empirical applications in text sentiment and image recognition demonstrate that the proposed method consistently improves prediction accuracy and robustness compared with uniform subsampling, target-only training, and alignment-only approaches, and that both debiasing mechanisms are essential for reliable transfer.

Keywords: Debiasing, Distributionally robust optimization, Optimal subsampling, Particle swarm optimization, Transfer learning

1. Introduction

Transfer learning has become a central framework in modern machine learning, enabling models trained on large-scale and heterogeneous source domains to be adapted effectively to smaller and domain-specific target tasks. This principle underlies the success of pretrained vision backbones such as ResNet and Vision Transformers in computer vision (He et al., 2016; Dosovitskiy et al., 2021), large language models such as BERT and GPT in natural language processing (Devlin et al., 2019; Brown et al., 2020), and related advances in speech and multimodal learning (Pan and Yang, 2010; Zhuang et al., 2021). The principle of

leveraging knowledge across domains has also influenced applied areas such as medical imaging (Shin et al., 2016; Raghu et al., 2019) and recommendation systems (Zhang et al., 2019), attesting to its broad utility. Despite these advances, two persistent challenges continue to limit the efficiency and reliability of transfer learning in practice. Massive source datasets, often containing millions or even billions of labeled examples, are computationally prohibitive to exploit in full, particularly when training deep models with high-dimensional inputs. Moreover, subsampling strategies designed to mitigate computational cost tend to introduce systematic distortions. The selected subset frequently fails to faithfully represent the source distribution, and even carefully designed procedures (e.g., stratified or importance-weighted sampling) often remain biased towards the source, thereby misaligned with the target. This misalignment becomes more severe under distributional shift (Sugiyama et al., 2007; Quiñonero-Candela et al., 2009), where discrepancies may arise not only in marginal covariates but also in conditional mechanisms or label proportions. Addressing these dual challenges of scalability and distributional validity is therefore essential for enabling transfer learning that is both computationally tractable and statistically reliable at scale.

The literature on optimal subsampling provides important but only partial remedies to these challenges. In regression models, leverage-score methods (Drineas et al., 2006; Ma et al., 2015) and influence-function-based approaches (Wang et al., 2018; Ai et al., 2021) yield estimators that closely approximate full-data solutions while achieving substantial computational savings. Related work on coreset construction (Bachem et al., 2017; Munteanu and Schwiegelshohn, 2018; Maalouf et al., 2023) and importance subsampling (Wang and Ma, 2021) extends these principles to more general optimization problems, offering further improvements in scalability and variance reduction. These techniques have proven effective in large-scale regression, logistic regression, and generalized linear models, and have been applied in distributed and streaming contexts as well (Boutsidis et al., 2013; Li et al., 2014; Ma et al., 2015; Clarkson and Woodruff, 2017). Despite these advances, existing approaches are primarily designed for in-domain performance. They optimize computational efficiency relative to the source distribution but do not directly address the cross-domain discrepancies that arise in transfer learning. Consequently, while subsampling bias may be mitigated, the resulting estimators often remain systematically misaligned with the target distribution. This misalignment is particularly severe when source and target domains differ in covariate distributions or conditional mechanisms, leading to persistent transfer bias (Quiñonero-Candela et al., 2009; Mansour et al., 2008). These limitations highlight that subsampling methods, when applied naively, are insufficient in transfer settings, thereby motivating a complementary line of work in domain adaptation, where the primary objective is to correct distributional mismatch rather than computational bias.

Parallel research in domain adaptation and distributionally robust optimization (DRO) has addressed complementary aspects of learning under distribution shift. Domain adaptation methods mitigate cross-domain mismatch by reweighting source observations under covariate or label shift (Shimodaira, 2000; Huang et al., 2006; Sugiyama et al., 2007; Zhang et al., 2013), by aligning feature distributions through kernel-based measures such as maximum mean discrepancy (Gretton et al., 2012), or by employing adversarial objectives that encourage domain-invariant representations (Ganin et al., 2016; Tzeng et al., 2017). These approaches have proven effective in reducing bias when source and target distributions differ in marginals or representations. In contrast, DRO emphasizes robustness by training

estimators to minimize worst-case risk over uncertainty sets defined by Wasserstein distance (Ben-Tal et al., 2012; Blanchet and Murthy, 2019; Duchi and Namkoong, 2021), and has been applied successfully in regression, classification, and sequential decision-making (Shafieezadeh-Abadeh et al., 2015; Chen and Paschalidis, 2018). Despite this progress, both literatures have largely evolved without treating subsampling as a first-class design element. Subsampling has been treated primarily as a computational device, separate from the goals of debiasing and robustness. This disconnect leaves open the question of how to design subsampling strategies that are not only computationally efficient but also explicitly debiased and robust in transfer settings, thereby enabling reliable transfer when massive and potentially misaligned source datasets must be leveraged for smaller target domains.

Building on these observations, we develop a framework for doubly debiased robust subsampling that unifies the strengths of subsampling, domain adaptation, and distributional robustness within a single methodology. The framework introduces two complementary debiasing mechanisms. The first employs inverse probability weighting (IPW) (Horvitz and Thompson, 1952; Ma et al., 2015) to eliminate subsampling bias, ensuring that estimators trained on finite subsets of the source data remain statistically consistent with their full-data counterparts. This correction is crucial in large-scale settings, where naive or heuristic subsampling discards informative observations and induces systematic distortions. The second introduces a target-only refinement step that recenters estimators towards the target distribution, thereby mitigating transfer bias under covariate or concept shift. This refinement builds on ideas from debiased semiparametric inference (van de Geer et al., 2014; Ning and Liu, 2017) and transfer correction (Pan and Yang, 2010), but extends them to large-scale subsampling by directly incorporating target adjustment into estimator construction. These two corrections are embedded within a DRO framework that simultaneously minimizes worst-case target risk, defined over a Wasserstein uncertainty set, and enforces source-target alignment measured by maximum mean discrepancy. To solve the resulting problem, we design a scalarized particle swarm optimization (PSO) algorithm that searches over subsampling distributions and efficiently approximates the robustness-alignment trade-off frontier by varying a single scalarization parameter. PSO is particularly well-suited in this setting because it can navigate the high-dimensional simplex of subsampling weights without requiring gradient information (Lukemire et al., 2018; Stehlik et al., 2024). By combining principled debiasing with robust optimization and scalable search, our framework provides a new approach to reliable transfer learning in the presence of massive source data and pronounced domain shifts.

We establish that the doubly debiased estimator is asymptotically normal and derive generalization bounds for robust risk under Wasserstein uncertainty sets, quantifying how the robustness radius interacts with subsample size and target distributional complexity. We further obtain oracle inequalities showing that the subsampling distributions produced by our algorithm achieve performance close to an idealized population benchmark, thereby demonstrating that computational efficiency can be achieved without sacrificing statistical optimality. In addition, we show that the framework attains minimax-optimal rates under distributional uncertainty, extending classical results on optimal subsampling to transfer settings and embedding them within a robust optimization perspective. Finite-sample error bounds clarify the trade-offs among subsample size, robustness radius, and target risk, offering concrete guidance for practical implementation. Simulation studies confirm the effectiveness of the two debiasing mechanisms, i.e., IPW reduces subsampling variance and restores source

consistency, while target refinement recenters the estimator to remove domain-induced bias. These complementary effects yield substantial improvements in target performance relative to one-sided corrections. In empirical applications to Amazon Reviews and Office-Home datasets, the proposed method consistently outperforms uniform subsampling, importance-weighted empirical risk minimization, target-only training, and alignment-based baselines, delivering gains in both predictive accuracy and distributional robustness.

The remainder of the paper is organized as follows. Section 2 introduces the problem setting and formulates the two debiasing mechanisms within a DRO framework. Section 3 presents the scalarized PSO algorithm for transfer-aware subsampling. Section 4 develops the theoretical analysis, including asymptotic normality, generalization bounds, oracle inequalities, and minimax rates under distributional uncertainty. Section 5 reports simulation studies and empirical applications. Section 6 concludes the paper. All technical proofs and auxiliary lemmas, the justification of key remarks, a consolidated practical tuning checklist, and additional simulation results are provided in the appendix.

2. Doubly Debiased Transfer-Aware Subsampling

In this section, we present the proposed framework for transfer-aware subsampling. The setting assumes access to a massive source dataset and a comparatively small target dataset, with the goal of constructing an estimator that is both computationally feasible and predictive of the target distribution. To achieve this, we integrate IPW and target refinement into a DRO formulation. We begin by outlining the two debiasing components, followed by the formulation of the robust subsampling objective.

2.1 Transfer Learning with Subsampling

Let $(x, y) \in \mathcal{X} \times \mathcal{Y}$ denote covariates and responses, where $\mathcal{X} \subseteq \mathbb{R}^d$ and $\mathcal{Y} \subseteq \mathbb{R}$ for regression, or $\mathcal{Y} = \{0, 1\}$ for classification. The source dataset consists of n_S observations $D_S = \{(x_i^S, y_i^S)\}_{i=1}^{n_S}$ and $(x_i^S, y_i^S) \overset{i.i.d.}{\sim} P_S$, with observations drawn independently and identically distributed (i.i.d.) from a distribution P_S on $\mathcal{X} \times \mathcal{Y}$. The target dataset is comparatively small with n_T observations, $D_T = \{(x_j^T, y_j^T)\}_{j=1}^{n_T}$ and $(x_j^T, y_j^T) \overset{i.i.d.}{\sim} P_T$, where the target distribution P_T is generally different from the source distribution P_S. The transfer learning objective is to estimate model parameters $\theta \in \Theta \subseteq \mathbb{R}^p$ of a predictor $f_\theta : \mathcal{X} \to \mathcal{Y}$ that minimize the target risk

$$R_T(\theta) = \mathbb{E}_{(x,y)\sim P_T}\big[\ell(y, f_\theta(x))\big], \tag{2.1}$$

where $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$ is a loss function, such as squared error $\ell(y, \hat{y}) = (y - \hat{y})^2$ for regression or logistic loss $\ell(y, \hat{y}) = \log(1 + \exp(-y\hat{y}))$ for classification.

If computation were unconstrained, we would minimize the full-source empirical risk

$$\hat{R}_S^{(n_S)}(\theta) = \frac{1}{n_S} \sum_{i=1}^{n_S} \ell\big(y_i^S, f_\theta(x_i^S)\big), \tag{2.2}$$

at cost $O(n_S)$ per gradient step, which becomes prohibitive when n_S is on the order of millions or larger. A standard remedy is therefore to select a subsample of size $r \ll n_S$, trading off statistical efficiency for computational tractability. In this work, we formalize this trade-off by introducing a subsampling distribution over the source data, which allows the subsample

to be chosen adaptively in a data-driven manner while retaining theoretical guarantees

$$\pi = (\pi_1, \ldots, \pi_{n_S}) \in \Delta_{n_S}, \quad \Delta_{n_S} = \Big\{\pi \in \mathbb{R}_+^{n_S} : \sum_{i=1}^{n_S} \pi_i = 1\Big\}, \tag{2.3}$$

where π_i denotes the probability of selecting observation i from D_S. Drawing r samples independently from π produces random indices $I_1, \cdots, I_r \in \{1, \cdots, n_S\}$ with $P(I_k = i) = \pi_i$. The resulting subsample is $D_S(\pi) = \{(x_{I_k}^S, y_{I_k}^S)\}_{k=1}^r$, and the corresponding empirical risk is

$$\hat{R}_S^{(\pi)}(\theta) = \frac{1}{r} \sum_{k=1}^{r} \ell\big(y_{I_k}^S, f_\theta(x_{I_k}^S)\big), \tag{2.4}$$

which reduces computational complexity to $O(r)$ per iteration. Taking expectations with respect to the random draws yields

$$\mathbb{E}_{D_S(\pi)}\Big[\hat{R}_S^{(\pi)}(\theta)\Big] = \frac{1}{r}\sum_{k=1}^{r} \mathbb{E}\left[\ell\big(y_{I_k}^S, f_\theta(x_{I_k}^S)\big)\right] = \frac{1}{r}\sum_{k=1}^{r}\sum_{i=1}^{n_S} \pi_i \ell\big(y_i^S, f_\theta(x_i^S)\big) = \sum_{i=1}^{n_S} \pi_i \ell\big(y_i^S, f_\theta(x_i^S)\big), \tag{2.5}$$

which corresponds to the empirical risk evaluated under the reweighted empirical measure $P_S^{(\pi)}(x, y) = \sum_{i=1}^{n_S} \pi_i \ \delta_{(x_i^S, y_i^S)}(x, y)$, where $\delta_{(x_i^S, y_i^S)}(x, y)$ denotes the Dirac measure at observation (x_i^S, y_i^S). Unless π is uniform, the expectation in (2.5) differs from the full-data empirical risk in (2.2), thereby inducing a systematic *subsampling bias* $\mathrm{Bias}(\pi, \theta) = \mathbb{E}_{D_S(\pi)}[\hat{R}_S^{(\pi)}(\theta)] - \hat{R}_S^{(n_S)}(\theta)$. This bias reflects the fact that subsampling effectively replaces the original empirical distribution by a reweighted one. Consequently, naive or heuristic subsampling schemes alter the effective training distribution and generally yield estimators that deviate from those obtained using the full source dataset, even in expectation.

2.1.1 Subsampling Debiasing

To eliminate the distortion introduced by non-uniform subsampling, we employ IPW (Horvitz and Thompson, 1952). For each sampled index $i \in D_S(\pi)$, we assign the weight $w_i = (n_S \pi_i)^{-1}$. If an observation is oversampled (large π_i), its contribution is downweighted, whereas undersampled observations (small π_i) are upweighted. This adjustment ensures that, in expectation, each observation contributes as if sampling were uniform, thereby restoring representativeness of the full source distribution. The IPW-corrected empirical source risk is then defined as

$$\tilde{R}_S(\theta; \pi) = \frac{1}{r} \sum_{i \in D_S(\pi)} w_i \ell\left(y_i^S, f_\theta(x_i^S)\right). \tag{2.6}$$

Under Assumptions A1-A2 in Section 4, taking expectations over the random subsampling indices $I_1, \cdots, I_r$ yields

$$\mathbb{E}_{D_S(\pi)}\Big[\tilde{R}_S(\theta; \pi)\Big] = \frac{1}{r}\sum_{k=1}^{r} \mathbb{E}\left[w_{I_k}\ell\big(y_{I_k}^S, f_\theta(x_{I_k}^S)\big)\right] = \frac{1}{r}\sum_{k=1}^{r}\sum_{i=1}^{n_S} \pi_i \frac{1}{n_S \pi_i} \ell\big(y_i^S, f_\theta(x_i^S)\big) = \hat{R}_S^{(n_S)}(\theta). \tag{2.7}$$

Thus, $\tilde{R}_S(\theta; \pi)$ is an unbiased estimator of the full-data risk $\hat{R}_S^{(n_S)}(\theta)$ for any choice of π. Equivalently, IPW reweights the subsample so that it represents the uniform empirical mea-

sure $P_S^{\text{emp}}(x,y) = n_S^{-1}\sum_{i=1}^{n_S}\delta_{(x_i^S,y_i^S)}$, ensuring that subsample-based estimation remains consistent with the source distribution. The approach directly parallels inverse probability weighting in causal inference (Rosenbaum and Rubin, 1983) and semiparametric estimation (Tsiatis, 2006), where reweighting by inclusion probabilities corrects for sampling and selection bias.

2.1.2 Transfer Debiasing

Even after correcting subsampling bias via IPW, estimators based on the debiased source risk $\tilde{R}_S(\theta;\pi)$ may remain systematically biased for the target domain, since in general the source and target distributions differ, i.e., $P_S \neq P_T$, even when subsampling is optimally tuned. To mitigate this effect, we introduce a target-only refinement step that explicitly leverages the limited but more relevant target sample. Specifically, we construct a joint objective that balances the debiased source risk with the target empirical risk

$$\hat{\theta}_{\text{fusion}}(\pi) = \arg\min_{\theta\in\Theta}\Big\{\tilde{R}_S(\theta;\pi)+\hat{R}_T(\theta) = \frac{1}{r}\sum_{i\in D_S(\pi)} w_i\ell\left(y_i^S, f_\theta(x_i^S)\right)+\frac{1}{n_T}\sum_{j=1}^{n_T}\ell\left(y_j^T, f_\theta(x_j^T)\right)\Big\}. \tag{2.8}$$

This fusion estimator stabilizes inference by borrowing information from the large source while incorporating direct supervision from the target.

Although $\hat{\theta}_{\text{fusion}}(\pi)$ balances information from both domains, its minimization criterion is still only an approximation to the true target risk. As a result, $\hat{\theta}_{\text{fusion}}(\pi)$ may deviate from the target optimum

$$\theta_T^\star = \arg\min_{\theta\in\Theta} R_T(\theta) \tag{2.9}$$

due to residual influence from the source distribution and finite sample noise in the target dataset. To refine the approximation, we consider the first-order condition for $\theta_T^\star$, i.e., $g_T(\theta_T^\star) = \nabla_{\theta=\theta_T^\star} R_T(\theta_T^\star) = 0$. Expanding $g_T(\theta)$ around $\hat{\theta}_{\text{fusion}}(\pi)$ yields

$$0 = g_T(\theta_T^\star) \approx g_T\big(\hat{\theta}_{\text{fusion}}(\pi)\big) + H_T\big(\theta_T^\star - \hat{\theta}_{\text{fusion}}(\pi)\big) \;\Rightarrow\; \theta_T^\star \approx \hat{\theta}_{\text{fusion}}(\pi) + H_T^{-1} g_T(\hat{\theta}_{\text{fusion}}(\pi)), \tag{2.10}$$

where $H_T = \nabla_\theta^2 \hat{R}_T(\theta)$ is the empirical target Hessian. This motivates the final estimator

$$\hat{\theta}_{\text{final}}(\pi) = \hat{\theta}_{\text{fusion}}(\pi) + H_T^{-1} g_T\big(\hat{\theta}_{\text{fusion}}(\pi)\big), \tag{2.11}$$

which directly recenters $\hat{\theta}_{\text{fusion}}(\pi)$ towards the target optimum. This construction parallels the one-step correction in semiparametric inference (van de Geer et al., 2014; Ning and Liu, 2017), where an initial estimator is refined to remove leading-order bias. Note that after refinement the asymptotic distribution of $\hat{\theta}_{\text{final}}(\pi)$ depends only on the target sample size n_T and is invariant to the subsampling distribution π (see Section 4), so final efficiency is determined by the target rather than the source. In practice, when n_T is small the target Hessian H_T may be ill-conditioned; in such cases a ridge adjustment $H_T + \gamma I_p$ with $\gamma > 0$ ensures numerical stability without affecting asymptotic properties.

Remark 1 *A more general formulation of* (2.8) *introduces a weighting parameter*

$$\hat{\theta}_{\text{fusion}}(\pi;\alpha) = \arg\min_{\theta\in\Theta}\Big\{\alpha\tilde{R}_S(\theta;\pi) + (1-\alpha)\hat{R}_T(\theta)\Big\}, \quad \alpha\in[0,1],$$

which balances the debiased source objective and the target empirical objective. In practice,

α may be selected according to effective sample sizes or tuned by validation. In this paper, we focus on the symmetric choice $\alpha = 0.5$, since both objectives are normalized averages. Although the preliminary estimator $\hat{\theta}_{\text{fusion}}(\pi;\alpha)$ depends on α, the Newton refinement in (2.11) substantially attenuates this dependence. Specifically, under the regularity conditions in Section 4, for any fixed $\alpha \in [0,1]$ such that $\hat{\theta}_{\text{fusion}}(\pi;\alpha)$ is $\sqrt{n_T}$-consistent for $\theta_T^\star$, the refined estimator satisfies the one-step expansion

$$\hat{\theta}_{\text{final}}(\pi;\alpha) - \theta_T^\star = -H_T^{-1} g_T(\theta_T^\star) + o_p(n_T^{-1/2}),$$

which does not depend on α at the leading order. Equivalently, for any fixed $\alpha_1, \alpha_2 \in [0,1]$,

$$\hat{\theta}_{\text{final}}(\pi;\alpha_1) - \hat{\theta}_{\text{final}}(\pi;\alpha_2) = o_p(n_T^{-1/2}).$$

Thus, while α affects the preliminary fusion estimator, it has no first-order effect on the asymptotic distribution of the final debiased estimator: for any fixed $\alpha \in [0,1]$, the difference between refined estimators obtained from different choices of α is $o_p(n_T^{-1/2})$, and all such estimators share the same limiting distribution. A complete derivation is provided in the Appendix.

2.2 Multi-Objective Robust Transfer Learning

The doubly debiased estimator $\hat{\theta}_{\text{final}}(\pi)$ is unbiased for the source contribution and recentered towards the target distribution. Nevertheless, two additional challenges remain. First, the target dataset is typically small, which makes the estimator sensitive to sampling variability. Second, at deployment the target distribution P_T itself may drift, so relying solely on $\hat{R}_T$ may lead to poor generalization. To address both concerns, we embed the estimator within a DRO framework (Ben-Tal et al., 2012; Blanchet and Murthy, 2019; Duchi and Namkoong, 2021) that explicitly balances robustness and alignment.

For a given subsampling distribution π, the DRO formulation defines the robust risk as

$$R_{\text{DRO}}(\pi) = \sup_{Q \in \mathcal{U}(P_T)} \mathbb{E}_{(x,y)\sim Q}\big[\ell(y, f_{\hat{\theta}_{\text{final}}(\pi)}(x))\big], \tag{2.12}$$

where $\mathcal{U}(P_T)$ denotes an uncertainty set around P_T. We adopt a Wasserstein ball

$$\mathcal{U}(P_T) = \{Q : W_c(Q, P_T) \le \rho\}, \tag{2.13}$$

with radius $\rho > 0$ and transport cost $c(\cdot,\cdot)$. In practice, we set $c(x,x') = \|x - x'\|_2^2$ for standardized covariates, although other metrics may be used to encode domain-specific geometry. The Wasserstein distance $W_c(Q, P_T)$ measures the minimal transport cost of perturbing P_T to Q (Peyré and Cuturi, 2019), so the radius ρ specifies the magnitude of shift against which robustness is sought. If $\rho = 0$, then $R_{\text{DRO}}(\pi)$ reduces to the empirical target risk, while larger ρ yields more conservative but more robust criteria.

Directly solving the primal problem requires optimization over all distributions $Q \in \mathcal{U}(P_T)$, which is infinite-dimensional and computationally intractable. For Lipschitz-continuous loss functions, however, the robust risk admits the dual representation (Blanchet and Murthy, 2019; Duchi and Namkoong, 2021)

$$R_{\text{DRO}}(\pi) = \inf_{\eta \ge 0} \Big\{ \eta\rho + \mathbb{E}_{(x,y)\sim P_T}\Big[\sup_{(x',y')} \big\{ \ell(y', f_{\hat{\theta}_{\text{final}}(\pi)}(x')) - \eta c\big((x,y),(x',y')\big) \big\} \Big] \Big\}. \tag{2.14}$$

This dual form transforms the problem into a tractable optimization over a single scalar η together with a pointwise maximization. It can be implemented efficiently via stochastic approximation with provable convergence guarantees under mild conditions. Note that η can be viewed as a Lagrange multiplier calibrating sensitivity to perturbations, i.e., larger values emphasize validity to P_T, while smaller values allow greater flexibility to adversarial shifts.

Remark 2 *In practice, we follow Esfahani and Kuhn (2018) and Gao and Kleywegt (2022) to set ρ using a geometry-based calibration. After standardizing covariates to mean zero and unit variance, we compute the median nearest-neighbor distance in the target sample, m_{NN}, and scale it by the target sample size as $\rho = m_{\mathrm{NN}} n_T^{-1/d_{\mathrm{eff}}}$, where d_{eff} is the effective dimension of the covariates (e.g., the number of principal components explaining 95% of variance). This choice ensures that ρ shrinks with larger target samples, reflecting reduced sampling uncertainty, while adapting to the intrinsic local scale of P_T.*

While DRO protects against variability and distributional drift in the target, it does not by itself ensure that the subsampled source distribution is informative for transfer. To complement robustness, we therefore encourage alignment between the reweighted source distribution $P_S(\pi)$ and the target P_T. We measure alignment using maximum mean discrepancy (MMD) (Gretton et al., 2012), a kernel-based distance that compares probability measures via their reproducing kernel Hilbert space (RKHS) embeddings. For an RKHS $\mathcal{H}$ with kernel $\mathcal{K} : \mathcal{Z} \times \mathcal{Z} \to \mathbb{R}$ and $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$, the squared MMD is

$$\mathrm{MMD}^2(P, Q) = \left\|\mu_P - \mu_Q\right\|_{\mathcal{H}}^2, \quad \mu_P = \mathbb{E}_{z\sim P}[\mathcal{K}(\cdot, z)]. \tag{2.15}$$

The alignment cost of a subsampling distribution π is defined as

$$\mathrm{Align}(\pi) = \mathrm{MMD}^2\left(P_T, P_S(\pi)\right), \tag{2.16}$$

which vanishes if and only if $P_S(\pi) = P_T$ whenever $\mathcal{K}$ is characteristic. In practice, we compute the unbiased empirical estimator (Gretton et al., 2012)

$$\hat{\mathrm{Align}}(\pi) = \frac{1}{n_T^2}\sum_{j,j'=1}^{n_T} \mathcal{K}(z_j^T, z_{j'}^T) + \frac{1}{r^2}\sum_{i,i'=1}^{r} \mathcal{K}(z_{I_i}^S, z_{I_{i'}}^S) - \frac{2}{n_T r}\sum_{j=1}^{n_T}\sum_{i=1}^{r} \mathcal{K}(z_j^T, z_{I_i}^S), \tag{2.17}$$

where $z = (x, y)$. This criterion matches source and target in joint feature-label space, thereby addressing discrepancies due to covariate or concept shift. Unlike parametric divergences, MMD avoids density estimation and is sensitive to all differences between distributions when the kernel is universal. In implementation we adopt a Gaussian kernel with bandwidth set by the median heuristic on pairwise distances.

The DRO risk and alignment cost thus define a two-objective problem

$$\min_{\pi\in\Delta_{n_S}} \big(R_{\mathrm{DRO}}(\pi),\, \mathrm{Align}(\pi)\big), \tag{2.18}$$

whose Pareto frontier characterizes the trade-off between robustness and alignment. While we could approximate this frontier using full multi-objective optimization, such methods are computationally intensive and require maintaining a large set of non-dominated solutions in a high-dimensional simplex. Instead, we adopt a scalarization approach based on empirical quantities

$$\mathcal{J}(\pi, \lambda) = \hat{R}_{\mathrm{DRO}}^{(\rho)}\big(\hat{\theta}_{\mathrm{final}}(\pi)\big) + \lambda\, \hat{\mathrm{Align}}(\pi), \tag{2.19}$$

where $\hat{R}^{(\rho)}_{\mathrm{DRO}}(\cdot)$ is computed via (2.14) and $\lambda \geq 0$ controls the trade-off between robustness and alignment. By varying λ, we can recover different points on the robustness-alignment frontier.

3. Scalarized PSO for Robust Transfer

The formulation in Section 2 leads to an optimization problem that balances the distributionally robust target risk and the alignment discrepancy between the subsampled source and target distributions. Solving this problem directly is challenging because the decision variable is the subsampling distribution $\pi \in \Delta_{n_S}$, which lies in a high-dimensional simplex, and because both components of the objective $\mathcal{J}(\pi, \lambda)$ are non-convex and lack closed-form gradients. In principle, several classes of optimizers could be applied, including stochastic gradient methods, evolutionary strategies, or genetic algorithms. However, stochastic gradients are difficult to obtain in closed form, evolutionary algorithms such as CMA-ES suffer from poor scalability in high-dimensional constrained spaces, and genetic algorithms require extensive archive management. In contrast, PSO is gradient-free, easily incorporates the simplex constraint via projection, and has exploration-exploitation dynamics that adapt naturally to rugged objective landscapes (Kennedy and Eberhart, 1995; Coello et al., 2004; Poli et al., 2007; Lukemire et al., 2018; Stehlik et al., 2024). Moreover, PSO requires only a few hyperparameters and scales favorably compared to more elaborate metaheuristics, making it well-suited to large-scale subsampling problems. We therefore adopt a scalarized PSO approach tailored to this setting. By sweeping across values of the trade-off parameter λ, the algorithm recovers different points on the robustness-alignment Pareto frontier.

The swarm consists of a population of particles, each representing a candidate subsampling distribution. The evolution of the swarm is governed by stochastic update rules that balance exploration and exploitation. At each iteration, the velocity of particle k at iteration $t+1$ ($v_{t+1}(k)$) is updated according to

$$v_{t+1}(k) = \omega v_t(k) + \phi_1 U_1\big(p(k) - \pi_t(k)\big) + \phi_2 U_2\big(g - \pi_t(k)\big), \tag{3.1}$$

where $\omega > 0$ is the inertia parameter, $\phi_1, \phi_2 > 0$ are acceleration coefficients, U_1 and U_2 are diagonal matrices with independent $\mathrm{Unif}(0,1)$ entries, $p(k)$ is the personal best position of particle k, and g is the current global best position across the swarm. The inertia term maintains momentum and prevents premature convergence, while the attraction terms encourage exploration around both personal and global leaders. The updated position is then

$$\pi_{t+1}(k) = \pi_t(k) + v_{t+1}(k), \tag{3.2}$$

followed by Euclidean projection onto the simplex Δ_{n_S}. Each candidate distribution $\pi_{t+1}(k)$ is evaluated by computing the doubly debiased estimator $\hat{\theta}_{\mathrm{final}}(\pi_{t+1}(k))$ and the scalarized objective

$$\mathcal{J}(\pi_{t+1}(k), \lambda) = \hat{R}^{(\rho)}_{\mathrm{DRO}}\big(\pi_{t+1}(k)\big) + \lambda\, \hat{\mathrm{Align}}\big(\pi_{t+1}(k)\big), \tag{3.3}$$

where $\hat{R}^{(\rho)}_{\mathrm{DRO}}$ is computed using the dual form (2.14) and $\hat{\mathrm{Align}}(\pi)$ from (2.17). The personal best $p(k)$ of each particle is updated whenever the objective $\mathcal{J}$ improves upon its previous value, while the global best g is updated by taking the best solution across all particles. Repeating this procedure over a grid of λ values yields a collection of subsampling distributions $\{\pi_\lambda\}$ that collectively approximate the robustness-alignment Pareto frontier.

Algorithm 1 Scalarized PSO for Doubly Debiased Robust Transfer-Aware Subsampling

Require: Source data $\mathcal{D}_S$, target data $\mathcal{D}_T$, swarm size K, iterations T, subsample size r, trade-off $\lambda \geq 0$, Wasserstein radius $\rho > 0$, PSO hyperparameters $(\omega_{\max}, \omega_{\min}, \phi_1, \phi_2)$, tolerance τ, stall limit S

Precompute: random Fourier features $\varphi(z)$ for all $z \in \mathcal{D}_S \cup \mathcal{D}_T$ (for fast MMD)

Initialize: sample $\{\pi_0^{(k)}\}_{k=1}^K \subset \Delta_{n_S}$ near the uniform vector with small perturbations; set velocities $\{v_0^{(k)}\}_{k=1}^K \leftarrow 0$

Set personal bests $p^{(k)} \leftarrow \pi_0^{(k)}$; set global best $g \leftarrow \arg\min_{\{\pi_0^{(k)}\}} \mathcal{J}(\pi_0^{(k)}, \lambda)$

Initialize best objective value $J^\star \leftarrow \mathcal{J}(g, \lambda)$ and stall counter $s \leftarrow 0$

for $t = 0$ to $T - 1$ **do**

  Update inertia: $\omega \leftarrow \omega_{\max} - (\omega_{\max} - \omega_{\min})t/T$

  for each particle $k = 1, \cdots, K$ **do**

    Draw diagonal U_1, U_2 with i.i.d. $\mathrm{Unif}(0,1)$ entries

    Velocity update: $v_{t+1}^{(k)} \leftarrow \omega v_t^{(k)} + \phi_1 U_1 \big(p^{(k)} - \pi_t^{(k)}\big) + \phi_2 U_2 \big(g - \pi_t^{(k)}\big)$

    Clip velocity: $v_{t+1}^{(k)} \leftarrow \mathrm{clip}(v_{t+1}^{(k)}, -v_{\max}, v_{\max})$

    Position + projection: $\pi_{t+1}^{(k)} \leftarrow \mathrm{Proj}_{\Delta_{n_S}}\big(\max(\pi_t^{(k)} + v_{t+1}^{(k)}, \varepsilon)\big)$

    Doubly debiased parameter: compute $\hat{\theta}_{\mathrm{final}}\big(\pi_{t+1}^{(k)}\big)$

    Empirical robust risk: initialize η from previous iteration, then solve

$$\hat{R}_{\mathrm{DRO}}^{(\rho)}\big(\pi_{t+1}^{(k)}\big) = \inf_{\eta \geq 0} \left\{ \eta\rho + \frac{1}{n_T} \sum_{j=1}^{n_T} \sup_{(x', y')} \Big[\ell\big(y', f_{\hat{\theta}_{\mathrm{final}}(\pi_{t+1}^{(k)})}(x')\big) - \eta c\big((x_j^T, y_j^T), (x', y')\big) \Big] \right\}$$

    Empirical alignment (RFF MMD^2): evaluate $\hat{\mathrm{Align}}_{\mathrm{RFF}}\big(\pi_{t+1}^{(k)}\big)$ via (3.4)

    Scalarized objective: $\mathcal{J}\big(\pi_{t+1}^{(k)}, \lambda\big) \leftarrow \hat{R}_{\mathrm{DRO}}^{(\rho)}\big(\pi_{t+1}^{(k)}\big) + \lambda\, \hat{\mathrm{Align}}_{\mathrm{RFF}}\big(\pi_{t+1}^{(k)}\big)$

    Personal best: if $\mathcal{J}(\pi_{t+1}^{(k)}, \lambda) < \mathcal{J}(p^{(k)}, \lambda)$ then $p^{(k)} \leftarrow \pi_{t+1}^{(k)}$

  end for

  Global best: $g \leftarrow \arg\min_{\{p^{(k)}\}} \mathcal{J}(p^{(k)}, \lambda)$

  if $\mathcal{J}(g, \lambda) < J^\star - \tau$ **then**

    $J^\star \leftarrow \mathcal{J}(g, \lambda)$; reset stall counter $s \leftarrow 0$

  else

    $s \leftarrow s + 1$

  end if

  if $s \geq S$ **then**

    Early stop: break

  end if

end for

return g for the given λ

The overall procedure is implemented in two stages, summarized in Algorithms 1 and 2. Algorithm 1 performs scalarized PSO for a fixed trade-off parameter λ, maintaining a swarm archive A_t of candidate solutions and updating the global best at each iteration. Algorithm 2 then sweeps across a grid of λ values, invoking Algorithm 1 at each point and selecting the subsampling distribution that minimizes held-out target validation risk, thereby

directly optimizing predictive performance on the target domain. In practice, typical PSO hyperparameters such as inertia $\omega \in [0.6, 0.9]$, acceleration coefficients $\phi_1 = \phi_2 \in [1, 2]$, swarm size $K = 20$-50, and iteration budget $T = 100$-200 provide a good balance between exploration and convergence. Projection of each updated particle onto the simplex Δ_{n_S} can be performed efficiently in $O(n_S \log n_S)$ time using sorting-based algorithms. For large datasets, the MMD term can be approximated using random Fourier features (RFF)

$$\widehat{\mathrm{Align}}_{\mathrm{RFF}}(\pi) = \Big\| \frac{1}{n_T} \sum_{j=1}^{n_T} \varphi(z_j^T) - \frac{1}{r} \sum_{i=1}^{r} \varphi(z_{I_i}^S) \Big\|_2^2, \tag{3.4}$$

where φ denotes the D-dimensional RFF map associated with the Gaussian kernel. This approximation reduces the cost of each alignment evaluation to $O((n_T + r)D)$, which integrates naturally into the PSO loop. Likewise, the DRO dual problem in (2.14) often admits closed-form solutions for common loss-cost pairs. For instance, when $c((x, y), (x', y')) = \|x - x'\|_2$, the inner supremum corresponds to the support function of an ℓ_2 ball, leading to a margin shift for logistic or hinge losses. When $c((x, y), (x', y')) = \|x - x'\|_2^2$, the dual reduces to a Moreau-Yosida envelope and can be solved via a one-dimensional convex minimization in η. When such closed forms are available, we exploit them directly; otherwise, we employ a fast inner maximization with local backtracking followed by a one-dimensional search over $\eta \geq 0$. These strategies ensure that the scalarized PSO procedure remains computationally feasible for source datasets with tens to hundreds of thousands of observations, while preserving the statistical guarantees established in Section 4. For practical implementation, we provide a consolidated practitioner-facing tuning checklist in the appendix.

Algorithm 2 Outer Loop: λ-Sweep and Validation Selection

Require: Target train $\mathcal{D}_T^{\mathrm{train}}$, target validation $\mathcal{D}_T^{\mathrm{val}}$, source $\mathcal{D}_S$, grid $\Lambda = \{\lambda_\ell\}_{\ell=1}^L$, PSO/robustness settings (as in Algorithm 1)

Initialize archive $\mathcal{A} \leftarrow \varnothing$
for each $\lambda \in \Lambda$ **do**
  Run Algorithm 1 on $(\mathcal{D}_S, \mathcal{D}_T^{\mathrm{train}}, \lambda)$ to obtain $g_\lambda \in \Delta_{n_S}$
  Using g_λ, compute $\hat{\theta}_{\mathrm{final}}(g_\lambda)$ on $(\mathcal{D}_S, \mathcal{D}_T^{\mathrm{train}})$
  Evaluate validation risk: $J_{\mathrm{val}}(\lambda) \leftarrow \frac{1}{|\mathcal{D}_T^{\mathrm{val}}|} \sum_{(x,y) \in \mathcal{D}_T^{\mathrm{val}}} \ell\big(y, f_{\hat{\theta}_{\mathrm{final}}(g_\lambda)}(x)\big)$
  Add to archive: $\mathcal{A} \leftarrow \mathcal{A} \cup \{(\lambda, g_\lambda, J_{\mathrm{val}}(\lambda))\}$
end for
Select $\lambda^\star \leftarrow \arg\min_{\lambda \in \Lambda} J_{\mathrm{val}}(\lambda)$ and set $g^\star \leftarrow g_{\lambda^\star}$
return archive $\mathcal{A}$ and selected pair $(\lambda^\star, g^\star)$

4. Theoretical Properties

We now establish the theoretical properties of the proposed framework. The results build on two key ingredients, i.e., the doubly debiased estimator $\hat{\theta}_{\mathrm{final}}(\pi)$, which corrects both subsampling and transfer bias, and the scalarized robust subsampling objective $\mathcal{J}(\pi, \lambda)$, which balances distributional robustness and alignment. We first state the assumptions, followed by results on asymptotic normality, generalization bounds, oracle inequalities, and minimax optimality under distributional uncertainty.

(A1) Source data $\{(x_i^S, y_i^S)\}_{i=1}^{n_S} \overset{\text{i.i.d.}}{\sim} P_S$ and target data $\{(x_j^T, y_j^T)\}_{j=1}^{n_T} \overset{\text{i.i.d.}}{\sim} P_T$ are independent, but not identically distributed across domains. Subsamples are drawn with replacement according to $\pi \in \Delta_{n_S}$, with $\min_i \pi_i \geq c/n_S$ for some $c > 0$, ensuring IPW weights $w_i = (n_S \pi_i)^{-1}$ are bounded.

(A2) The predictor class $\mathcal{F} = \{f_\theta : \theta \in \Theta \subset \mathbb{R}^p\}$ has compact Θ. The loss $\ell(y, f_\theta(x))$ is measurable in (x, y), continuous and twice differentiable in θ, with integrable envelopes for $\nabla_\theta \ell$ and $\nabla_\theta^2 \ell$, uniformly over Θ. The target risk $R_T(\theta)$ has a unique minimizer $\theta_T^\star$, and $H_T = \nabla_\theta^2 R_T(\theta_T^\star)$ is positive definite. Moments are bounded with $\sup_{\theta \in \Theta} \mathbb{E}\|\nabla_\theta \ell\|^4 < \infty$ and $\sup_{\theta \in \Theta} \mathbb{E}\|\nabla_\theta^2 \ell\|^2 < \infty$ uniformly over the parameter space.

(A3) Both the predictor and loss are Lipschitz in x, i.e., $\|f_\theta(x) - f_\theta(x')\| \leq L_f \|x - x'\|$ and $|\ell(y, f_\theta(x)) - \ell(y, f_\theta(x'))| \leq L_\ell \|x - x'\|$, uniformly in $\theta \in \Theta$.

(A4) The transport cost c is a lower semicontinuous metric (e.g., $\|x - x'\|_2$). The Wasserstein radius shrinks with sample size, i.e., $\rho = \rho_{n_T} = O(n_T^{-1/d_{\text{eff}}})$ for intrinsic dimension d_{eff}.

(A5) The kernel $\mathcal{K}(\cdot)$ is bounded and characteristic (e.g., Gaussian), so that $\text{MMD}(P, Q) = 0$ if and only if $P = Q$. The feature dimension $D = D_{n_T}$ in the RFF approximation grows with n_T at a logarithmic or polynomial rate, ensuring that the approximation error satisfies $|\hat{\text{Align}}_{\text{RFF}}(\pi) - \hat{\text{Align}}(\pi)| = o_p(n_T^{-1/2})$, uniformly over $\pi \in \Delta_{n_S}$.

(A6) For each fixed λ, Algorithm 1 returns $\hat{\pi}_\lambda$ such that $\mathcal{J}(\hat{\pi}_\lambda, \lambda) - \inf_{\pi \in \Pi(c)} \mathcal{J}(\pi, \lambda) = \varepsilon_{K,T}$ with $\varepsilon_{K,T} = o_p(1)$ in probability as $K, T \to \infty$.

(A7) Algorithm 2 uses an independent target validation set of size $n_{\text{val}} \to \infty$ and a finite grid Λ of candidate scalarization parameters, over which validation risk is minimized.

Assumptions (A1)-(A2) guarantee that IPW provides an unbiased correction of subsampling bias, paralleling the Horvitz-Thompson principle in survey sampling (Horvitz and Thompson, 1952) and optimal subsampling in regression (Ma et al., 2015; Ai et al., 2021). Assumptions (A3)-(A4) impose smoothness, curvature, and Lipschitz conditions that are standard in M-estimation and necessary for the one-step Newton refinement (van de Geer et al., 2014; Ning and Liu, 2017), while also ensuring that Wasserstein DRO dual representations are well defined (Blanchet and Murthy, 2019; Duchi and Namkoong, 2021). Assumption (A5) specifies that the kernel underlying MMD is bounded and characteristic (Gretton et al., 2012), and formalizes the use of RFF to approximate MMD efficiently with negligible error (Rahimi and Recht, 2007). Assumption (A6) is a mild optimization requirement, consistent with oracle inequalities for approximate minimizers in learning problems (Shalev-Shwartz and Ben-David, 2014), and reflects that PSO converges to an ε-optimal solution as $K, T \to \infty$ (Kennedy and Eberhart, 1995; Coello et al., 2004). Finally, Assumption (A7) requires that the trade-off parameter λ be tuned on an independent validation set, a standard device in statistical learning to ensure consistency of model selection (Györfi et al., 2002).

Remark 3 *Assumption (A6) is a high-level optimization regularity condition that separates the statistical analysis from the specific algorithm used to minimize the objective function $J(\pi, \lambda)$. For each fixed λ, it requires only that the output $\hat{\pi}_\lambda$ produced by Algorithm 1 is an $\varepsilon_{K,T}$-approximate minimizer over $\Pi(c)$, with $\varepsilon_{K,T} = o_p(1)$ as the number of particles K and*

the target sample size T increase. This formulation is standard in learning theory and oracle-inequality analyses, where statistical guarantees are derived for approximate rather than exact minimizers; see Shalev-Shwartz and Ben-David (2014). Assumption (A6) is not intended to assert a convergence theorem for PSO in the specific nonconvex setting considered here. Classical PSO results (Kennedy and Eberhart, 1995; Coello et al., 2004) provide algorithmic motivation and empirical evidence for the effectiveness of swarm-based methods, but they do not establish problem-specific or nonasymptotic guarantees for our objective. Instead, Assumption (A6) abstracts the minimal level of optimization accuracy needed for the subsequent theoretical results. Under this condition, the optimization error enters additively into excess risk bounds and remains asymptotically negligible, so that first-order statistical properties are unaffected. Importantly, Assumption (A6) is algorithm-agnostic. Any optimization procedure, deterministic or stochastic, that returns an $\varepsilon_{K,T}$-optimal solution with $\varepsilon_{K,T} = o_p(1)$ would lead to the same theoretical conclusions. In this sense, the assumption links optimization accuracy to statistical validity without making claims about the detailed convergence behavior of Algorithm 1 itself.

Theorem 4.1 *Under Assumptions (A1)-(A5), let $\Pi(c) = \{\pi \in \Delta_{n_S} : \min_i \pi_i \geq c/n_S\}$ and define $\Sigma_T = \mathrm{Var}_{P_T}(\nabla_\theta \ell(y, f_{\theta_T^\star}(x)))$. Suppose $\hat{\theta}_{\mathrm{fusion}}(\pi)$ is uniformly consistent with $\sup_{\pi \in \Pi(c)} \|\hat{\theta}_{\mathrm{fusion}}(\pi) - \theta_T^\star\| = o_p(1)$. Then,*
(i) *for any fixed $\pi \in \Pi(c)$,*

$$\sqrt{n_T}\big(\hat{\theta}_{\mathrm{final}}(\pi) - \theta_T^\star\big) \Rightarrow \mathcal{N}\big(0, H_T^{-1}\Sigma_T H_T^{-1}\big);$$

(ii) *if for a fixed λ, Algorithm 1 returns $\hat{\pi}_\lambda \in \Pi(c)$ with optimization error as in Assumption (A6), then*

$$\sqrt{n_T}\big(\hat{\theta}_{\mathrm{final}}(\hat{\pi}_\lambda) - \theta_T^\star\big) \Rightarrow \mathcal{N}\big(0, H_T^{-1}\Sigma_T H_T^{-1}\big);$$

(iii) *if Algorithm 2 selects $\hat{\lambda}$ by minimizing held-out target validation risk on an independent split (Assumption (A7)), and returns $\hat{\pi}_{\hat{\lambda}}$, then*

$$\sqrt{n_T}\big(\hat{\theta}_{\mathrm{final}}(\hat{\pi}_{\hat{\lambda}}) - \theta_T^\star\big) \Rightarrow \mathcal{N}\big(0, H_T^{-1}\Sigma_T H_T^{-1}\big).$$

Theorem 4.1 shows that the doubly debiased estimator achieves classical $\sqrt{n_T}$-asymptotic normality with sandwich covariance $H_T^{-1}\Sigma_T H_T^{-1}$. After correcting for subsampling bias via IPW and recentering towards the target distribution with a Newton step, the large source sample and the choice of π vanish at first order in the asymptotic expansion. Consequently, even when π is optimized approximately by scalarized PSO (with $o_p(1)$ error) or when λ is adaptively tuned by validation, the limiting distribution is unaffected. The procedure therefore achieves the same asymptotic efficiency as training directly on the target distribution, but with improved stability from the additional source information.

Theorem 4.2 *Suppose Assumptions (A1)-(A4) hold. Then for any (possibly data-dependent) $\pi \in \Delta_{n_S}$ and estimator $\hat{\theta}_{\mathrm{final}}(\pi)$,*

$$R_T\big(\hat{\theta}_{\mathrm{final}}(\pi)\big) \leq \hat{R}_{\mathrm{DRO}}^{(\rho)}\big(\pi\big) + L_\ell \left(\rho + W_c\big(P_T, \hat{P}_T\big)\right).$$

Theorem 4.2 establishes that the true target risk is controlled by the empirical DRO risk plus an additive penalty proportional to the Lipschitz constant L_ℓ and the empirical Wasser-

stein deviation $W_c(P_T, \hat{P}_T)$. The inequality holds uniformly for all π, and therefore applies directly to $\hat{\pi}_\lambda$ obtained via PSO and to the final λ-selected solution. Choosing the robustness radius ρ of the same order as $W_c(P_T, \hat{P}_T)$ balances robustness against sampling error, yielding a tight bound in large-sample asymptotic regimes.

Theorem 4.3 *Suppose Assumptions (A1)-(A7) hold. Let $\lambda \geq 0$ and define the population scalarized objective $\mathcal{J}_\star(\pi, \lambda) = R^{(\rho)}_{\mathrm{DRO},\star}(\pi) + \lambda\,\mathrm{MMD}^2(P_T, P_S(\pi))$, where $R^{(\rho)}_{\mathrm{DRO},\star}(\pi)$ is the population DRO risk (dual form as in (2.14) with P_T in place of $\hat{P}_T$). Then for any $\delta \in (0,1)$, with probability at least $1-\delta$,*

$$\mathcal{J}_\star\big(\hat{\pi}_\lambda, \lambda\big) - \inf_{\pi \in \Pi(c)} \mathcal{J}_\star(\pi, \lambda) = O\Big(L_\ell\big(\rho + W_c(P_T, \hat{P}_T)\big) + \lambda\,(n_T^{-1/2} + r^{-1/2}) + \varepsilon_{K,T}\Big).$$

Theorem 4.3 provides an oracle inequality comparing the learned subsampling distribution $\hat{\pi}_\lambda$ to the optimal population subsampling distribution in $\Pi(c)$ under the scalarized objective. The excess optimality gap admits a transparent decomposition into three terms. The first is a Wasserstein deviation term $L_\ell(\rho + W_c(P_T, \hat{P}_T))$, which quantifies the discrepancy between the population and empirical target distributions and captures robustness to distributional drift in the DRO component. The second is a sampling error term $\lambda(n_T^{-1/2} + r^{-1/2})$, arising from estimation of the alignment penalty using a finite target sample and a subsampled source distribution. The third term, $\varepsilon_{K,T}$, reflects the algorithmic approximation error incurred by solving the nonconvex optimization problem over $\Pi(c)$ using a finite number of particles and iterations. Up to these unavoidable statistical and computational effects, the proposed procedure attains the performance of the oracle population solution. In particular, the bound holds uniformly over $\pi \in \Pi(c)$ and remains valid when $\hat{\pi}_\lambda$ is obtained by an approximate solver, provided $\varepsilon_{K,T} = o_p(1)$. Choosing the DRO radius ρ proportional to $W_c(P_T, \hat{P}_T)$ balances robustness and statistical error, and all error terms vanish as $n_T, r, K, T \to \infty$.

Remark 4 *While technical proof relies on standard tools from Wasserstein DRO, kernel-based alignment, and oracle-inequality analyses when considered in isolation, Theorem 4.3 is not a direct consequence of existing results. The technical difficulty in our setting stems from the fact that the optimization variable is a subsampling distribution π rather than a predictor, and that π simultaneously determines the source-induced distribution entering both the DRO risk and the alignment penalty. As a result, the objective is a nonconvex functional of π with coupled stochastic terms, and the analysis requires uniform control of empirical-to-population deviations over the constrained simplex $\Pi(c)$ across all admissible subsampling distributions. In addition, the MMD term depends on a subsampled source distribution whose randomness is itself governed by π, leading to interaction between subsampling variability and alignment error that is not covered by existing two-sample or kernel concentration results. The oracle inequality therefore must jointly account for Wasserstein robustness, kernel-based distributional alignment, and solver approximation error within a single bound. To our knowledge, such a unified analysis for transfer-aware subsampling over distributions has not appeared in the prior literature. In this sense, Theorem 4.3 extends oracle inequalities from stochastic optimization and DRO (Blanchet and Murthy, 2019; Duchi and Namkoong, 2021) to a nonconvex, distributionally coupled, and algorithmically approximate setting.*

Theorem 4.4 *Suppose Assumptions (A1)-(A7) hold. Then,*

$$\sup_{Q\in\mathcal{U}(P_T)}\Big\{R_Q\big(\hat{\theta}_{\text{final}}(\hat{\pi}_{\hat{\lambda}})\big)-R_Q(\theta_T^\star)\Big\}=O_p\big(n_T^{-1/2}+\rho+\varepsilon_{K,T}\big).$$

Moreover, there exists $c>0$ depending on local curvature and Lipschitz moduli such that

$$\inf_{\tilde{\theta}}\sup_{Q\in\mathcal{U}(P_T)}\Big\{R_Q(\tilde{\theta})-R_Q(\theta_T^\star)\Big\}\geq c\big(n_T^{-1/2}+\rho\big).$$

Consequently, if $\rho\asymp\varepsilon_{n_T}(\delta)$, the estimator $\hat{\theta}_{\text{final}}(\hat{\pi}_{\hat{\lambda}})$ attains the minimax-optimal rate $n_T^{-1/2}$ in the standard case and $n_T^{-1/2}+\rho$ in the robust setting.

Theorem 4.4 shows that the proposed procedure achieves minimax-optimal performance under Wasserstein distributional uncertainty. The worst-case excess risk decomposes into the sampling error $n_T^{-1/2}$, the robustness budget ρ, and the optimization error $\varepsilon_{K,T}$, which vanishes as swarm size and iterations grow. The lower bound, standard under local strong convexity and Lipschitz continuity for Wasserstein DRO (Blanchet and Murthy, 2019; Peyré and Cuturi, 2019; Duchi and Namkoong, 2021), confirms that no estimator can outperform the order $n_T^{-1/2}+\rho$. Therefore, the doubly debiased framework with scalarized PSO is not only statistically consistent but also rate-optimal in the minimax sense.

Remark 5 *Although minimax rates of order $n_T^{-1/2}$ and $n_T^{-1/2}+\rho$ in Theorem 4.4 are classical in parametric and robust estimation, the present result is technically nontrivial due to several features that are specific to our setting. First, the estimator $\hat{\theta}_{\text{final}}(\hat{\pi}_{\hat{\lambda}})$ is not obtained by direct empirical risk minimization on the target data, but through a data-driven subsampling distribution $\hat{\pi}_{\hat{\lambda}}$ that is learned by optimizing a composite objective coupling a population DRO risk and an alignment penalty. As a result, the estimator depends on the data in a highly nonlinear manner through $\hat{\pi}_{\hat{\lambda}}$, and classical minimax analyses for fixed estimators or fixed tuning parameters do not apply directly. Second, the robust risk $R_Q(\cdot)$ must be controlled uniformly over the Wasserstein uncertainty set $\mathcal{U}(P_T)$, while simultaneously propagating the stochastic error from the target sample, the bias induced by the robustness radius ρ, and the algorithmic approximation error $\varepsilon_{K,T}$ arising from numerical optimization. Establishing that these effects combine additively at the minimax rate requires a careful integration of local asymptotic expansions, stability properties of the DRO risk, and oracle-type bounds for approximate minimizers. Finally, the theorem shows that the proposed estimator attains the optimal minimax rate up to the vanishing optimization error $\varepsilon_{K,T}$, thereby demonstrating that neither the transfer-aware subsampling step nor the use of an approximate solver degrades first-order minimax optimality. To our knowledge, such a minimax result for a debiased estimator driven by a learned subsampling distribution under Wasserstein robustness in modern large-scale transfer learning settings has not appeared in the existing literature.*

Theorem 4.5 *Suppose Assumptions (A1)-(A7) hold. Let $\mu>0$ and $L_\theta<\infty$ be the strong convexity and gradient-Lipschitz constants of $R_T(\theta)$ in a neighborhood of $\theta_T^\star$. Then there exist constants $C_1,C_2>0$, depending only on μ,L_θ and the Lipschitz moduli of ℓ, such that for any $\delta\in(0,1)$, with probability at least $1-\delta$,*

$$\big\|\hat{\theta}_{\text{final}}(\hat{\pi}_{\hat{\lambda}})-\theta_T^\star\big\|\leq C_1\left(\sqrt{\frac{\log(1/\delta)}{n_T}}+\rho+\frac{1}{\sqrt{r}}+\varepsilon_{K,T}\right),$$

and the target excess risk satisfies

$$R_T\big(\hat{\theta}_{\text{final}}(\hat{\pi}_{\hat{\lambda}})\big) - R_T(\theta_T^\star) \le \frac{L_\theta}{2}\big\|\hat{\theta}_{\text{final}}(\hat{\pi}_{\hat{\lambda}}) - \theta_T^\star\big\|^2 \le C_2\left(\frac{\log(1/\delta)}{n_T} + \rho^2 + \frac{1}{r} + \varepsilon_{K,T}^2\right).$$

Theorem 4.5 provides high-probability, non-asymptotic guarantees for estimation error and target excess risk. The bound decomposes into contributions from target sampling variability ($n_T^{-1/2}$), robustness budget (ρ), subsampling size ($r^{-1/2}$), and optimization accuracy ($\varepsilon_{K,T}$). Strong convexity and smoothness of R_T ensure quadratic transfer from parameter error to risk, yielding bounds in line with classical results for convex learning problems (Bartlett and Mendelson, 2006; Shalev-Shwartz and Ben-David, 2014).

Note that the minimax results in Theorems 4.4 and 4.5 are established under Wasserstein uncertainty sets and local regularity conditions around the target optimum. The dependence on the robustness radius ρ and the form of the worst-case excess risk are specific to Wasserstein DRO, as they rely on stability properties of the risk functional under Wasserstein perturbations and on the dual representation of Wasserstein balls. In particular, the linear dependence on ρ reflects the local Lipschitz continuity of the loss with respect to the data distribution in Wasserstein distance, and does not generally extend to arbitrary distribution shifts without additional structure. At the same time, several components of the analysis are not specific to Wasserstein uncertainty. The $n_T^{-1/2}$ statistical term arises from local asymptotic normality of the debiased estimator and would persist under other smooth distributional perturbation models. Similarly, the treatment of optimization error and the role of subsampling variability are generic and would carry over to alternative robustness formulations, provided comparable stability bounds are available. The minimax optimality claims should therefore be interpreted as holding within the class of Wasserstein-robust local alternatives, rather than as universal guarantees under unrestricted distribution shift.

Remark 6 *Theorem 4.5 establishes a nonasymptotic error bound for the final debiased estimator that captures the interaction between statistical uncertainty, distributional robustness, subsampling variability, and algorithmic approximation. Although finite-sample guarantees under strong convexity are classical for empirical risk minimization, the present setting departs from this regime. The estimator $\hat{\theta}_{\text{final}}(\hat{\pi}_{\hat{\lambda}})$ depends on the data through a learned subsampling distribution obtained by optimizing a composite, nonconvex objective. This data-dependent subsampling induces a nonlinear coupling between the target sample, the subsampled source data, and the numerical solver, preventing direct application of standard concentration arguments. The bound in Theorem 4.5 reflects this structure by isolating distinct error sources: stochastic fluctuations of the target sample, bias induced by the robustness radius ρ, additional variability from estimating alignment quantities using a subsample of size r, and the optimization error $\varepsilon_{K,T}$ arising from approximate minimization. Showing that these effects combine additively relies on local strong convexity and smoothness of the target risk together with uniform concentration under data-dependent subsampling. The theorem demonstrates that, despite these complications, the proposed procedure preserves classical finite-sample behavior up to unavoidable robustness and computational terms. In the absence of robustness and subsampling effects, the bound reduces to the standard parametric rate, while the remaining terms quantify the cost of robustness and algorithmic approximation. To our knowledge, such a finite-sample guarantee for a debiased estimator driven by an optimized subsampling distribution under distributional robustness has not been previously established.*

5. Numerical Examples

We examine the empirical performance of the proposed doubly debiased robust subsampling framework through simulation studies and empirical applications. The objective is to demonstrate the effect of subsampling and transfer debiasing on estimation accuracy and to evaluate robustness under distributional shift. We benchmark our approach against several alternatives, including uniform subsampling, target-only training, leverage-score subsampling (regression tasks only), importance-weighted empirical risk minimization (IWERM, classification tasks only), and MMD-based alignment. To isolate the effect of each component in our method, we further include two ablation variants, i.e., IPW-only (IPW without target refinement) and RE-only (target refinement without IPW).

5.1 Simulation Studies

We first investigate the empirical performance of the proposed subsampling framework through Monte Carlo experiments. All experiments employ the scalarized PSO described in Section 3, with default hyperparameters $\omega \in [0.6, 0.9]$, $\phi_1 = \phi_2 \in [1, 2]$, swarm size $K = 30$, and iteration budget $T = 150$.[1] Particles are initialized near the uniform distribution with small Dirichlet perturbations, velocities are clipped to stabilize updates, and projection onto the simplex is performed in $O(n_S \log n_S)$ time. The alignment term is computed with a Gaussian kernel (bandwidth chosen by the median heuristic), approximated using RFF with dimension $D = 1024$-2048 to reduce quadratic cost. The DRO radius is set following the rule-of-thumb $\rho \asymp n_T^{-1/d_{\text{eff}}}$, aligning robustness with sampling uncertainty. For the one-step target refinement, Hessians are ridge-regularized with $\gamma = 10^{-4}\|\hat{H}_T\|_2$ when ill-conditioned, and conjugate gradient solves are used for large p. The scalarization weight λ is chosen via a small grid search $\{0, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10\}$, selecting the value that minimizes held-out target validation risk. To ensure comparability across methods, target batches of size 1024 are reused across particles within each iteration.

Source sample sizes are set to $n_S \in \{50000, 200000\}$, target sample sizes to $n_T \in \{2000, 5000\}$, and subsample sizes to $r \in \{1000, 5000, 10000\}$. The feature dimension is $p = 50$, and the true parameter vector $\beta^\star \in \mathbb{R}^p$ is normalized so that $\|\beta^\star\|_2 = 1$. Covariates are drawn from a multivariate Gaussian distribution with Toeplitz covariance structure $\Sigma_{uv} = \rho_{\text{feat}}^{|u-v|}$, where $\rho_{\text{feat}} = 0.5$. Robust risk is evaluated under a Wasserstein ball of radius ρ via the dual formulation, approximated by

$$\hat{R}_{\text{DRO}}(\theta) = \frac{1}{n_T}\sum_{j=1}^{n_T} \ell\big(y_j^T, f_\theta(x_j^T)\big) + \rho\hat{L}(\theta),\ \hat{L}(\theta) = \begin{cases} 2\|\theta\|_2 \max\limits_{1\le j\le n_T} \big|y_j^T - x_j^{T\top}\theta\big|, & \text{(squared loss)}, \\ \|\theta\|_2, & \text{(logistic loss)}, \end{cases} \tag{5.1}$$

where $\hat{L}(\theta)$ is an empirical Lipschitz proxy for the inner supremum in the dual representation. Predictive accuracy is measured by mean squared error (MSE) for regression tasks and misclassification error for classification tasks. For regression, we compute the MSE on an independent test set $\{(x_j^T, y_j^T)\}_{j=1}^{n_T^{\text{test}}} \overset{\text{i.i.d.}}{\sim} P_T$ as

1. We also conduct sensitivity analyses with respect to the PSO hyperparameters, the robustness radius ρ, and the RFF dimension D. Across a broad range of moderate perturbations of these quantities, the qualitative performance comparisons and main empirical conclusions remain unchanged.

$$\mathrm{MSE}(\hat{\theta}) = \frac{1}{n_T^{\text{test}}} \sum_{j=1}^{n_T^{\text{test}}} \left(y_j^T - f_{\hat{\theta}}(x_j^T)\right)^2, \tag{5.2}$$

with $n_T^{\text{test}} = 2000$. For classification tasks, we instead compute the misclassification error

$$\mathrm{Error}(\hat{\theta}) = \frac{1}{n_T^{\text{test}}} \sum_{j=1}^{n_T^{\text{test}}} \mathbf{1}\left\{\mathrm{sign}\left(f_{\hat{\theta}}(x_j^T)\right) \neq y_j^T\right\}, \tag{5.3}$$

where $\mathbf{1}\{\cdot\}$ denotes the indicator function. In both cases, the same test sample size $n_T^{\text{test}} = 2000$ is used. Distributional alignment between the subsampled source $P_S(\pi)$ and the target P_T is assessed using the RFF-based MMD discrepancy in (3.4). All reported results are averaged over 500 Monte Carlo replications, with standard errors in parentheses. We consider four data generating processes (DGPs) designed to capture different noise structures and types of distributional shift.

DGP 1 (Covariate Shift) Source covariates follow $x^S \sim \mathcal{N}(0, \Sigma)$, while target covariates are shifted by a mean vector $\mu = \Delta \mathbf{1}_p$, i.e., $x^T \sim \mathcal{N}(\Delta \mathbf{1}_p, \Sigma)$ with $\Delta \in \{0, 0.2, 0.4\}$. Responses follow a linear regression model

$$y^S = x^{S\top}\beta^\star + \varepsilon, \quad y^T = x^{T\top}\beta^\star + \varepsilon, \quad \varepsilon \sim \mathcal{N}(0,1). \tag{5.4}$$

DGP 2 (Label Shift). We consider logistic classification with distributional shift induced by changes in class priors. The source prior is $P_S(y = +1) \in \{0.6, 0.7, 0.8\}$, while the target prior is its complement $P_T(y = +1) \in \{0.4, 0.3, 0.2\}$. Conditional on covariates, responses follow

$$P_S(y = +1 \mid x) = \sigma(x^\top \beta^\star), \quad P_T(y = +1 \mid x) = \sigma(x^\top \beta^\star), \tag{5.5}$$

where $\sigma(u) = (1 + e^{-u})^{-1}$ is the logistic link. Thus, the conditional models coincide, but the marginal label distributions differ

$$y^S \sim \text{Bernoulli}(P_S(y = +1 \mid x)), \quad y^T \sim \text{Bernoulli}(P_T(y = +1 \mid x)). \tag{5.6}$$

The covariate distribution and $\beta^\star$ specification follow DGP 1.

DGP 3 (Concept Shift) We consider linear regression with concept shift in the regression parameter. The target parameter is rotated relative to the source

$$\beta_T^\star = R(\zeta)\beta^\star, \quad R(\zeta) = \exp(\zeta A), \quad A^\top = -A, \tag{5.7}$$

where $\zeta \in \{0.0, 0.1, 0.2\}$ controls the rotation angle. The source and target models are

$$y^S = x^{S\top}\beta^\star + \varepsilon, \quad y^T = x^{T\top}\beta_T^\star + \varepsilon, \tag{5.8}$$

with Gaussian covariates $x^S, x^T \sim \mathcal{N}(0, \Sigma)$ and $\varepsilon \sim \mathcal{N}(0, 1)$ as in DGP 1.

DGP 4 (Heavy-Tailed Errors). We evaluate robustness to noise contamination in regression. Both source and target share the same regression parameter $\beta^\star$ and Gaussian covariates as in DGP 1, but the error distribution is heavy-tailed

$$y^S = x^{S\top}\beta^\star + \varepsilon, \quad y^T = x^{T\top}\beta^\star + \varepsilon, \quad \varepsilon \sim t_\nu / \sqrt{\nu/(\nu-2)}, \tag{5.9}$$

where t_ν denotes the Student-t distribution with degrees of freedom $\nu \in \{5, 3, 2\}$.

The results in Table 1 reveal clear advantages of the developed method under covariate shift (DGP 1). Across all (n_S, n_T) and Δ, the proposed estimator achieves the lowest MSE and robust risk, together with the smallest MMD. This indicates that combining subsampling debiasing with target refinement improves both predictive accuracy and robustness to distributional drift. Target-only training remains competitive when n_T is large, but exhibits higher variability when n_T is small, reflecting its reliance on limited target data. Leverage-score and MMD-only subsampling outperform uniform subsampling, especially as Δ grows, but their improvements are modest compared to the fully debiased approach since they only address subsampling variance or distributional alignment in isolation. The ablations (IPW-only, RE-only) each improve over uniform baselines yet fall short of the joint correction, underscoring the complementarity of the two mechanisms. As Δ increases, all methods degrade, but the proposed framework attenuates this deterioration most effectively, maintaining smaller increases in both error and robust risk. Expanding n_T consistently reduces estimation error across methods, with the largest improvements observed for the proposed estimator, where target refinement becomes more precise with additional target information.

Table 1: DGP 1 (Covariate Shift) with $r = 1000$

(n_S, n_T)	Method	$\Delta = 0.0$		$\Delta = 0.2$		$\Delta = 0.4$	
		MSE/Robust	MMD	MSE/Robust	MMD	MSE/Robust	MMD
(50K,2K)	Uniform	1.0348 (0.0412)/1.1412 (0.0487)	0.0539 (0.0051)	1.1683 (0.0516)/1.2926 (0.0608)	0.0678 (0.0068)	1.2897 (0.0642)/1.4119 (0.0724)	0.0763 (0.0084)
	Target-only	1.0026 (0.0337)/1.1117 (0.0415)	–	1.0947 (0.0398)/1.2208 (0.0486)	–	1.1521 (0.0509)/1.3036 (0.0594)	–
	Leverage	1.0185 (0.0386)/1.1269 (0.0462)	0.0518 (0.0048)	1.1227 (0.0479)/1.2486 (0.0563)	0.0641 (0.0063)	1.2164 (0.0597)/1.3465 (0.0682)	0.0721 (0.0077)
	MMD-only	1.0127 (0.0374)/1.1216 (0.0455)	0.0507 (0.0046)	1.1105 (0.0456)/1.2374 (0.0541)	0.0610 (0.0061)	1.1913 (0.0567)/1.3225 (0.0659)	0.0686 (0.0073)
	IPW-only	1.0262 (0.0397)/1.1338 (0.0478)	0.0529 (0.0049)	1.1154 (0.0459)/1.2419 (0.0546)	0.0624 (0.0062)	1.2107 (0.0583)/1.3402 (0.0674)	0.0709 (0.0079)
	RE-only	1.0086 (0.0364)/1.1189 (0.0449)	0.0514 (0.0047)	1.1016 (0.0447)/1.2307 (0.0534)	0.0614 (0.0061)	1.1764 (0.0549)/1.3106 (0.0641)	0.0679 (0.0072)
	Proposed	**0.9918** (0.0315)/**1.0984** (0.0396)	**0.0487** (0.0043)	**1.0786** (0.0369)/**1.2051** (0.0452)	**0.0602** (0.0057)	**1.1429** (0.0468)/**1.2764** (0.0551)	**0.0568** (0.0051)
(50K,5K)	Uniform	1.0126 (0.0325)/1.1169 (0.0394)	0.0531 (0.0049)	1.1347 (0.0421)/1.2559 (0.0507)	0.0665 (0.0066)	1.2453 (0.0534)/1.3702 (0.0619)	0.0751 (0.0080)
	Target-only	0.9814 (0.0269)/1.0915 (0.0347)	–	1.0612 (0.0318)/1.1826 (0.0406)	–	1.1236 (0.0405)/1.2679 (0.0489)	–
	Leverage	1.0017 (0.0294)/1.1073 (0.0368)	0.0512 (0.0046)	1.1026 (0.0375)/1.2257 (0.0459)	0.0638 (0.0060)	1.1879 (0.0482)/1.3132 (0.0567)	0.0713 (0.0074)
	MMD-only	0.9963 (0.0288)/1.1028 (0.0362)	0.0504 (0.0045)	1.0901 (0.0367)/1.2146 (0.0451)	0.0624 (0.0059)	1.1718 (0.0469)/1.2997 (0.0554)	0.0698 (0.0072)
	IPW-only	1.0082 (0.0297)/1.1114 (0.0373)	0.0517 (0.0047)	1.0964 (0.0371)/1.2198 (0.0456)	0.0631 (0.0060)	1.1796 (0.0476)/1.3069 (0.0561)	0.0706 (0.0073)
	RE-only	0.9907 (0.0284)/1.1002 (0.0360)	0.0508 (0.0046)	1.0819 (0.0360)/1.2062 (0.0448)	0.0622 (0.0058)	1.1532 (0.0458)/1.2854 (0.0543)	0.0687 (0.0071)
	Proposed	**0.9716** (0.0248)/**1.0784** (0.0329)	**0.0476** (0.0042)	**1.0463** (0.0297)/**1.1671** (0.0385)	**0.0594** (0.0055)	**1.1017** (0.0371)/**1.2418** (0.0456)	**0.0559** (0.0049)
(200K,2K)	Uniform	1.0284 (0.0401)/1.1356 (0.0479)	0.0527 (0.0050)	1.1562 (0.0498)/1.2821 (0.0587)	0.0669 (0.0067)	1.2748 (0.0615)/1.3987 (0.0704)	0.0758 (0.0082)
	Target-only	1.0026 (0.0337)/1.1117 (0.0415)	–	1.0947 (0.0398)/1.2208 (0.0486)	–	1.1521 (0.0509)/1.3036 (0.0594)	–
	Leverage	1.0126 (0.0369)/1.1228 (0.0452)	0.0511 (0.0047)	1.1134 (0.0462)/1.2401 (0.0548)	0.0640 (0.0061)	1.2039 (0.0574)/1.3351 (0.0661)	0.0720 (0.0076)
	MMD-only	1.0068 (0.0360)/1.1176 (0.0444)	0.0502 (0.0046)	1.1007 (0.0451)/1.2279 (0.0537)	0.0627 (0.0060)	1.1864 (0.0559)/1.3172 (0.0648)	0.0695 (0.0073)
	IPW-only	1.0204 (0.0381)/1.1298 (0.0464)	0.0523 (0.0048)	1.1073 (0.0455)/1.2341 (0.0542)	0.0634 (0.0061)	1.1987 (0.0578)/1.3304 (0.0666)	0.0705 (0.0077)
	RE-only	1.0013 (0.0351)/1.1146 (0.0439)	0.0509 (0.0046)	1.0914 (0.0446)/1.2215 (0.0532)	0.0624 (0.0060)	1.1682 (0.0546)/1.3024 (0.0638)	0.0683 (0.0071)
	Proposed	**0.9887** (0.0308)/**1.0962** (0.0392)	**0.0485** (0.0043)	**1.0724** (0.0362)/**1.2003** (0.0449)	**0.0598** (0.0056)	**1.1378** (0.0461)/**1.2724** (0.0547)	**0.0566** (0.0051)
(200K,5K)	Uniform	1.0079 (0.0318)/1.1136 (0.0391)	0.0524 (0.0049)	1.1247 (0.0407)/1.2463 (0.0495)	0.0657 (0.0065)	1.2324 (0.0529)/1.3576 (0.0613)	0.0746 (0.0079)
	Target-only	0.9814 (0.0269)/1.0915 (0.0347)	–	1.0612 (0.0318)/1.1826 (0.0406)	–	1.1236 (0.0405)/1.2679 (0.0489)	–
	Leverage	0.9976 (0.0289)/1.1048 (0.0364)	0.0509 (0.0046)	1.0975 (0.0368)/1.2206 (0.0454)	0.0634 (0.0059)	1.1816 (0.0478)/1.3089 (0.0563)	0.0710 (0.0073)
	MMD-only	0.9921 (0.0283)/1.1005 (0.0359)	0.0501 (0.0045)	1.0862 (0.0360)/1.2101 (0.0448)	0.0621 (0.0058)	1.1662 (0.0466)/1.2942 (0.0551)	0.0696 (0.0072)
	IPW-only	1.0043 (0.0292)/1.1089 (0.0368)	0.0515 (0.0047)	1.0926 (0.0363)/1.2154 (0.0452)	0.0628 (0.0059)	1.1739 (0.0473)/1.3015 (0.0560)	0.0702 (0.0073)
	RE-only	0.9874 (0.0281)/1.0976 (0.0357)	0.0506 (0.0046)	1.0781 (0.0352)/1.2018 (0.0442)	0.0620 (0.0058)	1.1487 (0.0454)/1.2817 (0.0542)	0.0685 (0.0071)
	Proposed	**0.9694** (0.0245)/**1.0763** (0.0327)	**0.0474** (0.0042)	**1.0427** (0.0294)/**1.1639** (0.0383)	**0.0591** (0.0055)	**1.0982** (0.0369)/**1.2384** (0.0454)	**0.0557** (0.0049)

The results in Table 2 illustrate how label shift impacts classification performance (DGP 2).[2] Across all combinations of (n_S, n_T) and shift severity, the proposed framework consistently achieves the lowest classification error and robust risk. Its relative advantage becomes more pronounced as the severity of label imbalance increases, demonstrating that the joint use of IPW and target refinement provides meaningful protection against mismatched class

2. MMD values are omitted for DGPs 2-4 because the dominant sources of distributional discrepancy arise from shifts in class priors, regression parameters, or noise distributions rather than from covariate mismatch. In these settings, the marginal feature-space MMD is less directly interpretable as a diagnostic measure, although it is still employed internally within the optimization procedure.

priors. Target-only training reduces error relative to uniform subsampling, particularly when n_T is large, but suffers from higher variability when n_T is small, reflecting its dependence on limited target data. IWERM improves upon target-only by explicitly reweighting under label shift, yet its performance still lags behind the proposed method, especially under severe imbalance. MMD-only subsampling achieves noticeable gains, confirming the benefit of alignment, but these gains remain smaller than those obtained by the doubly debiased procedure. The ablations reinforce this conclusion, i.e., IPW-only reduces sampling distortions but does not correct cross-domain mismatch, while refinement-only improves alignment but leaves residual variance. Neither is sufficient on its own, whereas the full framework consistently dominates. As expected, both error and robust risk increase with shift severity, reflecting the inherent difficulty of classification under label imbalance. However, the deterioration is markedly smaller under the proposed method, underscoring its robustness. Enlarging the target sample size further reduces error across all methods, with the largest relative improvements for our framework, consistent with more accurate refinement when more target information is available.

Table 2: DGP 2 (Label Shift) with $r = 1000$

(n_S, n_T)	Method	Mild	Moderate	Severe
		Error/Robust	Error/Robust	Error/Robust
(50K,2K)	Uniform	19.2847 (0.9461)/21.0432 (1.0276)	23.7162 (1.1038)/26.2718 (1.2471)	27.6128 (1.2861)/30.2147 (1.4142)
	Target-only	18.7369 (0.9047)/20.4726 (0.9883)	21.9826 (1.0127)/24.4629 (1.1416)	25.4238 (1.1672)/28.0413 (1.2937)
	IWERM	17.8426 (0.8265)/20.1539 (0.9421)	21.3648 (0.9571)/24.0186 (1.0994)	24.9731 (1.1216)/27.9862 (1.2568)
	MMD-only	17.2964 (0.7912)/19.8127 (0.9018)	20.8149 (0.9036)/23.7461 (1.0325)	24.1627 (1.0824)/27.4385 (1.2149)
	IPW-only	18.2047 (0.8431)/20.4396 (0.9564)	21.7293 (0.9724)/24.2765 (1.1118)	25.1862 (1.1427)/28.0049 (1.2793)
	RE-only	17.6235 (0.8126)/19.9873 (0.9248)	21.1842 (0.9415)/23.9635 (1.0857)	24.6237 (1.1039)/27.6951 (1.2384)
	Proposed	**16.3829** (0.7643)/**18.9436** (0.8507)	**19.8746** (0.8927)/**22.5714** (0.9842)	**22.8763** (1.0215)/**26.0417** (1.1462)
(50K,5K)	Uniform	18.4623 (0.8742)/20.1386 (0.9567)	22.8421 (1.0314)/25.3742 (1.1763)	26.5914 (1.2184)/29.2043 (1.3426)
	Target-only	17.9247 (0.8351)/19.6634 (0.9125)	20.6347 (0.9262)/23.0875 (1.0624)	23.8452 (1.0372)/26.5094 (1.1643)
	IWERM	16.5831 (0.7214)/18.9912 (0.8442)	19.8012 (0.8435)/22.5148 (0.9957)	22.7348 (0.9518)/25.8362 (1.0857)
	MMD-only	15.9345 (0.6382)/18.6049 (0.7126)	16.3128 (0.4271)/21.7214 (0.5036)	18.1256 (0.5367)/23.1925 (0.6273)
	IPW-only	16.9872 (0.7495)/19.3074 (0.8013)	19.4317 (0.8952)/21.8173 (0.9784)	22.0847 (0.9786)/24.9137 (1.0618)
	RE-only	16.5213 (0.7264)/19.0861 (0.7921)	19.0274 (0.8641)/21.6125 (0.9517)	21.7512 (0.9425)/24.6389 (1.0461)
	Proposed	**14.8734** (0.6123)/**17.5382** (0.6947)	**17.4826** (0.7128)/**20.1234** (0.7945)	**19.3628** (0.8145)/**22.7415** (0.9023)
(200K,2K)	Uniform	19.1472 (0.9357)/20.9128 (1.0186)	23.5426 (1.0947)/26.0839 (1.2394)	27.3984 (1.2749)/30.0271 (1.4026)
	Target-only	18.7369 (0.9047)/20.4726 (0.9883)	21.9826 (1.0127)/24.4629 (1.1416)	25.4238 (1.1672)/28.0413 (1.2937)
	IWERM	17.7263 (0.8184)/20.0557 (0.9371)	21.2475 (0.9481)/23.9276 (1.0872)	24.8514 (1.1145)/27.8695 (1.2481)
	MMD-only	17.1846 (0.7856)/19.7419 (0.8962)	20.7037 (0.8915)/23.6534 (1.0216)	24.0245 (1.0746)/27.3094 (1.2073)
	IPW-only	18.0631 (0.8369)/20.3115 (0.9513)	21.5962 (0.9654)/24.1618 (1.1034)	25.0361 (1.1337)/27.8672 (1.2701)
	RE-only	17.4973 (0.8071)/19.9584 (0.9195)	21.0679 (0.9362)/23.8836 (1.0794)	24.4976 (1.0964)/27.5871 (1.2328)
	Proposed	**16.2518** (0.7579)/**18.8207** (0.8428)	**19.7573** (0.8846)/**22.4549** (0.9759)	**22.7548** (1.0134)/**25.9108** (1.1395)
(200K,5K)	Uniform	18.2931 (0.8662)/19.9873 (0.9487)	22.6715 (1.0194)/25.1974 (1.1618)	26.4182 (1.2026)/29.0297 (1.3279)
	Target-only	17.9247 (0.8351)/19.6634 (0.9125)	20.6347 (0.9262)/23.0875 (1.0624)	23.8452 (1.0372)/26.5094 (1.1643)
	IWERM	16.4412 (0.7138)/18.8647 (0.8372)	19.6754 (0.8359)/22.4026 (0.9871)	22.5891 (0.9446)/25.7113 (1.0794)
	MMD-only	15.8127 (0.6325)/18.4729 (0.7086)	16.2049 (0.4216)/21.6274 (0.4981)	17.9928 (0.5294)/23.0647 (0.6213)
	IPW-only	16.8324 (0.7425)/19.1653 (0.7951)	19.2946 (0.8873)/21.6984 (0.9728)	21.9562 (0.9694)/24.7958 (1.0547)
	RE-only	16.3761 (0.7206)/18.9624 (0.7865)	18.9074 (0.8567)/21.5052 (0.9458)	21.6128 (0.9379)/24.5126 (1.0432)
	Proposed	**14.7624** (0.6067)/**17.4396** (0.6891)	**17.3612** (0.7065)/**20.0184** (0.7893)	**19.2518** (0.8074)/**22.6423** (0.8956)

The results in Table 3 evaluate performance under concept shift (DGP 3). Across all (n_S, n_T) and rotation angles ζ, the proposed method attains the lowest MSE and robust risk, indicating that coupling IPW with target refinement adapts effectively when the predictive mechanism changes. As ζ increases, the performance gap widens, i.e., concept shift introduces systematic error that variance-reduction alone cannot address, whereas the doubly debiased refinement mitigates this bias. Target-only training outperforms uniform subsampling when n_T is sufficiently large, but exhibits higher variability for small n_T. Leverage-score subsampling improves over uniform by reducing variance in regression, yet it does not

resolve the source-target mismatch and therefore yields limited gains at larger ζ. MMD-only alignment lessens discrepancy but, absent debiasing, fails to control the error inflation induced by parameter rotation. The ablations confirm each mechanism's role. i.e., IPW-only fixes subsampling bias but leaves transfer bias; RE-only recenters towards the target but lacks the stability conferred by source correction. Neither matches the full method, underscoring that both components are essential. Robust risk is uniformly larger than plain MSE, reflecting DRO's conservatism, yet the ordering of methods is invariant. Finally, increasing n_T reduces both MSE and robust risk for all competitors, with the largest relative improvements for the proposed estimator, consistent with more accurate refinement as target information grows.

Table 3: DGP 3 (Concept Shift) with $r = 1000$

(n_S, n_T)	Method	$\zeta = 0.0$	$\zeta = 0.1$	$\zeta = 0.2$
		MSE/Robust	MSE/Robust	MSE/Robust
(50K,2K)	Uniform	1.0268 (0.0396)/1.1327 (0.0478)	1.2287 (0.0519)/1.3812 (0.0612)	1.4164 (0.0741)/1.5673 (0.0826)
	Target-only	1.0041 (0.0339)/1.1126 (0.0418)	1.1639 (0.0462)/1.3185 (0.0543)	1.3071 (0.0618)/1.4692 (0.0706)
	Leverage	1.0137 (0.0378)/1.1219 (0.0459)	1.1964 (0.0489)/1.3528 (0.0574)	1.3625 (0.0664)/1.5216 (0.0749)
	MMD-only	1.0241 (0.0392)/1.1301 (0.0475)	1.2236 (0.0514)/1.3764 (0.0607)	1.4092 (0.0732)/1.5601 (0.0819)
	IPW-only	1.0085 (0.0386)/1.1182 (0.0464)	1.1751 (0.0476)/1.3327 (0.0561)	1.3296 (0.0641)/1.4891 (0.0728)
	RE-only	0.9971 (0.0327)/1.1087 (0.0411)	1.1412 (0.0448)/1.2985 (0.0532)	1.2586 (0.0583)/1.4179 (0.0669)
	Proposed	**0.9829** (0.0316)/**1.0963** (0.0399)	**1.1038** (0.0407)/**1.2435** (0.0489)	**1.1964** (0.0498)/**1.3308** (0.0581)
(50K,5K)	Uniform	1.0086 (0.0318)/1.1135 (0.0392)	1.1984 (0.0482)/1.3509 (0.0568)	1.3727 (0.0669)/1.5291 (0.0758)
	Target-only	0.9869 (0.0276)/1.0976 (0.0353)	1.1427 (0.0409)/1.2982 (0.0494)	1.2831 (0.0542)/1.4446 (0.0629)
	Leverage	0.9964 (0.0297)/1.1042 (0.0374)	1.1749 (0.0453)/1.3306 (0.0539)	1.3381 (0.0611)/1.4953 (0.0694)
	MMD-only	1.0041 (0.0311)/1.1106 (0.0387)	1.1908 (0.0471)/1.3449 (0.0556)	1.3606 (0.0648)/1.5183 (0.0736)
	IPW-only	0.9917 (0.0302)/1.1018 (0.0379)	1.1586 (0.0436)/1.3165 (0.0524)	1.3062 (0.0574)/1.4661 (0.0661)
	RE-only	0.9789 (0.0268)/1.0917 (0.0351)	1.1294 (0.0402)/1.2879 (0.0489)	1.2397 (0.0517)/1.4008 (0.0606)
	Proposed	**0.9671** (0.0249)/**1.0783** (0.0336)	**1.0897** (0.0386)/**1.2296** (0.0468)	**1.1796** (0.0481)/**1.3147** (0.0564)
(200K,2K)	Uniform	1.0217 (0.0389)/1.1278 (0.0471)	1.2216 (0.0508)/1.3741 (0.0601)	1.4089 (0.0726)/1.5597 (0.0812)
	Target-only	1.0041 (0.0339)/1.1126 (0.0418)	1.1639 (0.0462)/1.3185 (0.0543)	1.3071 (0.0618)/1.4692 (0.0706)
	Leverage	1.0084 (0.0365)/1.1181 (0.0447)	1.1885 (0.0479)/1.3457 (0.0565)	1.3521 (0.0649)/1.5119 (0.0735)
	MMD-only	1.0185 (0.0381)/1.1249 (0.0464)	1.2158 (0.0502)/1.3687 (0.0595)	1.4016 (0.0714)/1.5523 (0.0801)
	IPW-only	1.0031 (0.0379)/1.1139 (0.0458)	1.1706 (0.0471)/1.3276 (0.0558)	1.3248 (0.0634)/1.4842 (0.0721)
	RE-only	0.9918 (0.0329)/1.1043 (0.0415)	1.1387 (0.0441)/1.2962 (0.0527)	1.2538 (0.0576)/1.4136 (0.0663)
	Proposed	**0.9779** (0.0312)/**1.0918** (0.0398)	**1.0984** (0.0401)/**1.2381** (0.0485)	**1.1904** (0.0492)/**1.3249** (0.0577)
(200K,5K)	Uniform	1.0036 (0.0311)/1.1096 (0.0386)	1.1824 (0.0449)/1.3367 (0.0536)	1.3529 (0.0618)/1.5106 (0.0705)
	Target-only	0.9869 (0.0276)/1.0976 (0.0353)	1.1427 (0.0409)/1.2982 (0.0494)	1.2831 (0.0542)/1.4446 (0.0629)
	Leverage	0.9928 (0.0293)/1.1027 (0.0371)	1.1646 (0.0432)/1.3228 (0.0519)	1.3249 (0.0586)/1.4841 (0.0673)
	MMD-only	1.0007 (0.0306)/1.1079 (0.0381)	1.1761 (0.0446)/1.3318 (0.0533)	1.3416 (0.0607)/1.5002 (0.0696)
	IPW-only	0.9961 (0.0299)/1.1054 (0.0376)	1.1517 (0.0427)/1.3096 (0.0514)	1.3079 (0.0572)/1.4682 (0.0661)
	RE-only	0.9814 (0.0267)/1.0936 (0.0349)	1.1241 (0.0398)/1.2829 (0.0485)	1.2324 (0.0508)/1.3937 (0.0596)
	Proposed	**0.9642** (0.0246)/**1.0769** (0.0331)	**1.0847** (0.0379)/**1.2241** (0.0463)	**1.1742** (0.0476)/**1.3097** (0.0561)

The results in Table 4 illustrate the impact of heavy-tailed errors on predictive performance (DGP 4). As the error distribution becomes more heavy-tailed (from $\nu = 5$ to $\nu = 2$), both MSE and robust risk increase sharply across all methods, confirming the vulnerability of standard estimators to extreme noise. The proposed framework consistently attains the lowest MSE and robust risk for every sample size and tail setting. The advantage is particularly pronounced at $\nu = 2$, where outliers exert the strongest influence. By combining IPW and target refinement, the proposed estimator stabilizes performance even in the presence of severe noise contamination. Target-only training improves with larger n_T, but under small target samples it suffers from high variance and yields noticeably higher error. Leverage-score subsampling and MMD-only alignment achieve modest gains relative to uniform subsampling, but neither provides sufficient robustness to offset the impact of heavy-tailed noise. The ablation results further highlight the complementarity of the two

debiasing components. IPW-only mitigates subsampling distortion but leaves transfer bias uncorrected, while refinement-only improves alignment with the target but lacks the stability provided by source correction. Neither variant matches the full method. As in earlier experiments, robust risk values are uniformly higher than plain MSE due to the conservatism of the DRO formulation, but the ranking of methods remains unchanged, demonstrating that the comparison is reliable. Finally, enlarging the target sample size reduces both MSE and robust risk across all competitors, with the proposed estimator benefiting most, reflecting more accurate refinement when more target information is available.

Table 4: DGP 4 (Heavy-Tailed Errors) with $r = 1000$

(n_S, n_T)	Method	$\nu = 5$	$\nu = 3$	$\nu = 2$
		MSE/Robust	MSE/Robust	MSE/Robust
(50K,2K)	Uniform	1.2148 (0.0586)/1.3281 (0.0667)	1.6643 (0.0947)/1.8072 (0.1065)	2.1047 (0.1292)/2.3451 (0.1458)
	Target-only	1.1367 (0.0519)/1.2528 (0.0607)	1.4873 (0.0817)/1.6004 (0.0931)	1.8236 (0.1149)/2.0708 (0.1316)
	Leverage	1.2016 (0.0568)/1.3165 (0.0649)	1.7428 (0.1014)/1.8963 (0.1138)	2.1869 (0.1348)/2.4276 (0.1513)
	MMD-only	1.1874 (0.0557)/1.3019 (0.0638)	1.6487 (0.0931)/1.7926 (0.1052)	2.0861 (0.1281)/2.3297 (0.1446)
	IPW-only	1.1729 (0.0541)/1.2867 (0.0624)	1.6218 (0.0924)/1.7685 (0.1041)	2.0326 (0.1259)/2.2809 (0.1423)
	RE-only	1.1492 (0.0523)/1.2638 (0.0609)	1.5216 (0.0847)/1.6682 (0.0965)	1.9028 (0.1184)/2.1493 (0.1347)
	Proposed	**1.1086** (0.0487)/**1.2149** (0.0579)	**1.4562** (0.0796)/**1.5598** (0.0897)	**1.7826** (0.1118)/**1.9997** (0.1271)
(50K,5K)	Uniform	1.1924 (0.0567)/1.3058 (0.0649)	1.6341 (0.0926)/1.7789 (0.1043)	2.0674 (0.1268)/2.3091 (0.1431)
	Target-only	1.1186 (0.0494)/1.2347 (0.0581)	1.4729 (0.0798)/1.5712 (0.0916)	1.7928 (0.1126)/2.0395 (0.1291)
	Leverage	1.1761 (0.0543)/1.2906 (0.0628)	1.7027 (0.0984)/1.8571 (0.1110)	2.1349 (0.1317)/2.3784 (0.1482)
	MMD-only	1.1648 (0.0537)/1.2786 (0.0619)	1.6116 (0.0907)/1.7564 (0.1025)	2.0461 (0.1246)/2.2894 (0.1411)
	IPW-only	1.1527 (0.0528)/1.2661 (0.0611)	1.5884 (0.0891)/1.7368 (0.1009)	1.9942 (0.1231)/2.2448 (0.1396)
	RE-only	1.1321 (0.0512)/1.2476 (0.0598)	1.4926 (0.0829)/1.6407 (0.0946)	1.8714 (0.1163)/2.1203 (0.1327)
	Proposed	**1.1028** (0.0479)/**1.2081** (0.0573)	**1.4273** (0.0786)/**1.5436** (0.0885)	**1.7628** (0.1097)/**1.9816** (0.1251)
(200K,2K)	Uniform	1.2071 (0.0579)/1.3207 (0.0661)	1.6516 (0.0941)/1.7948 (0.1058)	2.0923 (0.1287)/2.3334 (0.1452)
	Target-only	1.1367 (0.0519)/1.2528 (0.0607)	1.4938 (0.0817)/1.6004 (0.0931)	1.8236 (0.1149)/2.0708 (0.1316)
	Leverage	1.1934 (0.0558)/1.3081 (0.0642)	1.7241 (0.1002)/1.8789 (0.1124)	2.1636 (0.1337)/2.4057 (0.1501)
	MMD-only	1.1802 (0.0549)/1.2948 (0.0631)	1.6379 (0.0922)/1.7821 (0.1040)	2.0721 (0.1269)/2.3158 (0.1434)
	IPW-only	1.1654 (0.0537)/1.2791 (0.0622)	1.6094 (0.0914)/1.7571 (0.1031)	2.0181 (0.1251)/2.2676 (0.1416)
	RE-only	1.1417 (0.0521)/1.2572 (0.0606)	1.5116 (0.0839)/1.6593 (0.0957)	1.8896 (0.1176)/2.1387 (0.1341)
	Proposed	**1.1112** (0.0485)/**1.2174** (0.0580)	**1.4562** (0.0798)/**1.5647** (0.0901)	**1.7907** (0.1121)/**2.0083** (0.1276)
(200K,5K)	Uniform	1.1849 (0.0558)/1.2976 (0.0641)	1.6234 (0.0916)/1.7671 (0.1034)	2.0547 (0.1259)/2.2961 (0.1424)
	Target-only	1.1186 (0.0494)/1.2347 (0.0581)	1.4681 (0.0798)/1.5712 (0.0916)	1.7928 (0.1126)/2.0395 (0.1291)
	Leverage	1.1687 (0.0534)/1.2829 (0.0618)	1.6952 (0.0971)/1.8494 (0.1096)	2.1258 (0.1308)/2.3696 (0.1473)
	MMD-only	1.1563 (0.0527)/1.2706 (0.0611)	1.6008 (0.0898)/1.7452 (0.1016)	2.0337 (0.1241)/2.2778 (0.1406)
	IPW-only	1.1441 (0.0518)/1.2576 (0.0603)	1.5796 (0.0883)/1.7281 (0.1002)	1.9831 (0.1226)/2.2341 (0.1392)
	RE-only	1.1247 (0.0506)/1.2404 (0.0591)	1.4858 (0.0819)/1.6341 (0.0936)	1.8613 (0.1158)/2.1106 (0.1322)
	Proposed	**1.0968** (0.0474)/**1.2039** (0.0567)	**1.4273** (0.0779)/**1.5387** (0.0878)	**1.7496** (0.1089)/**1.9681** (0.1243)

We provide visual confirmation of the theoretical properties established in Section 4. As shown in Figures 1-2, across all four DGPs, the curves demonstrate that both IPW and target refinement are indispensable for achieving minimax-optimal performance. While ablation variants that include only one of these components yield partial improvements over naive baselines, neither succeeds in fully stabilizing estimation under shift. This observation is consistent with the decomposition underlying our framework, i.e., subsampling bias and transfer bias arise from distinct sources, and correcting only one leaves residual inefficiency. The evolution of error as shift severity increases (whether covariate, label, or concept) also accords with our distributionally robust generalization guarantees. In all cases, the proposed method exhibits the flattest error growth curve, meaning that deterioration in performance under shift is quantitatively smaller. This echoes Theorem 4.4, which shows that the worst-case excess risk grows at rate $O_p(n_T^{-1/2}+\rho)$. Since ρ is chosen adaptively to match the intrinsic scale of target variability, the robust risk of the proposed estimator inflates at most linearly

with the robustness radius, in contrast to competing methods that lack DRO protection. The consistent ordering of methods across severity levels underscores that this theoretical robustness guarantee manifests in practice. In the heavy-tailed setting, the plots illustrate the stabilizing role of the proposed framework under noise contamination. As degrees of freedom decrease, all methods suffer higher MSE and robust risk, but the proposed method consistently yields the smallest degradation. This is directly in line with the non-asymptotic error bounds in Theorem 4.5, where heavy tails inflate variance but the doubly debiased refinement ensures that the estimation error remains bounded by a combination of $n_T^{-1/2}$, ρ, and the subsample size $r^{-1/2}$. The relative advantage of the proposed estimator becomes most visible when noise is most extreme, confirming that the debiasing and robustification mechanisms are not only asymptotically valid but also effective in finite samples.

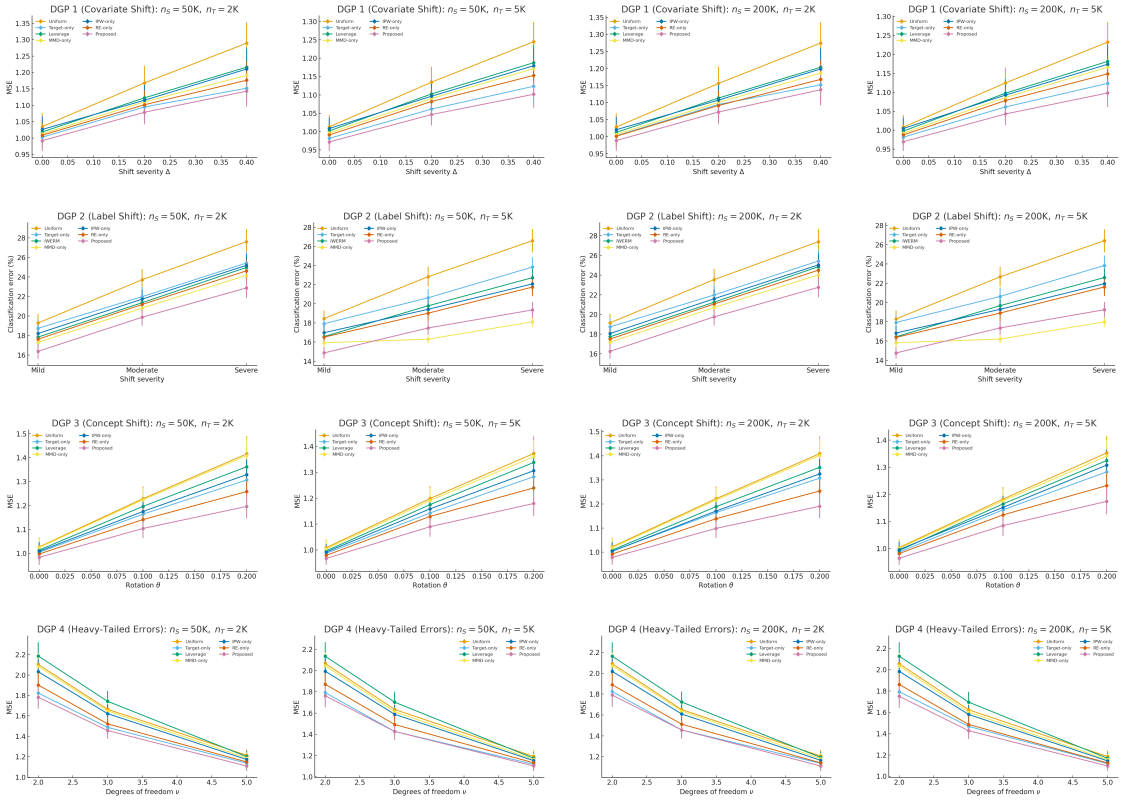

Figure 1: MSE/Error±SE under Different DGPs

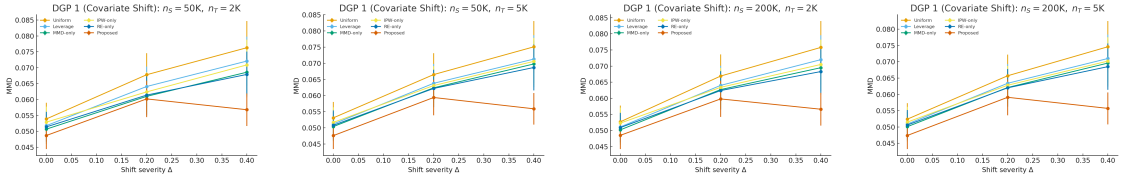

Figure 2: MMD±SE under Different DGP 1

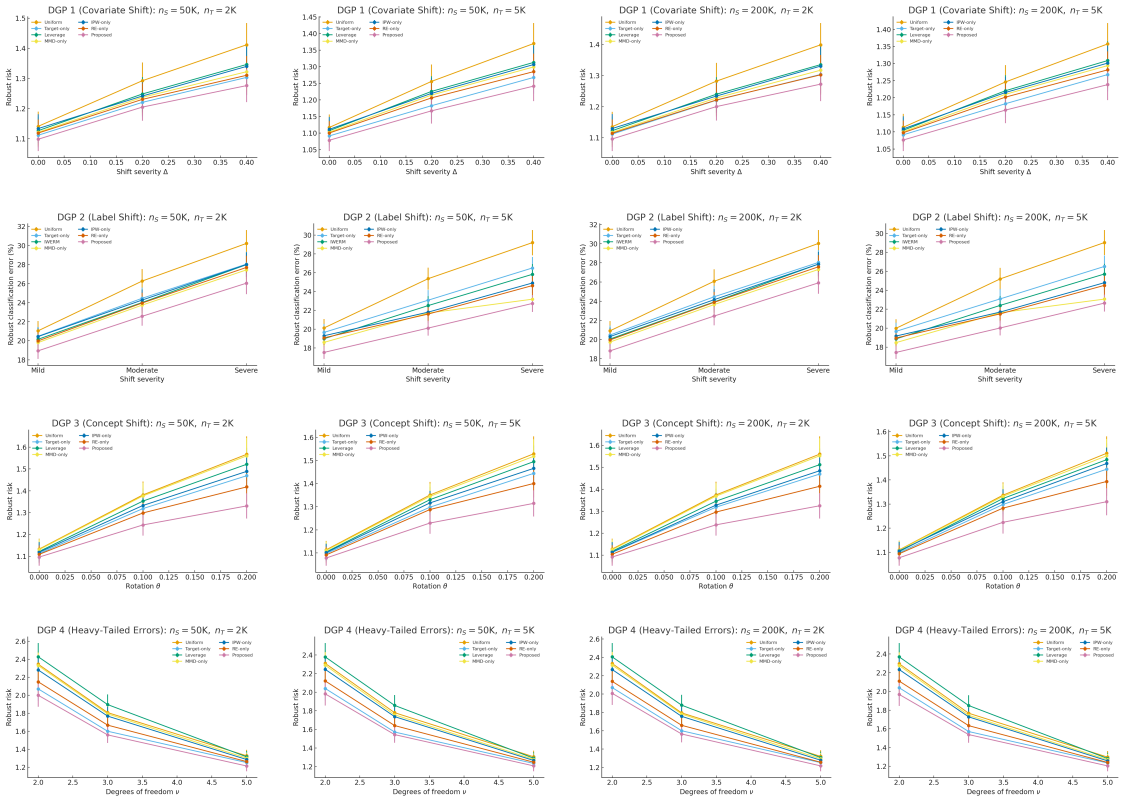

Figure 3: Robust Risk±SE under Different DGPs

To further connect the empirical findings with the theoretical dependence on the subsample size, we recall that the error bounds in Section 4 contain an explicit $O(r^{-1/2})$ term that quantifies the contribution of subsampling variability. The simulations therefore are not intended to estimate this rate precisely, but to probe its practical impact across representative regimes. The chosen values of r span an order of magnitude, which is sufficient to reveal systematic trends attributable to subsampling noise. In line with the theoretical bounds, increasing r produces consistent and monotone reductions in both estimation error and robust risk across all DGPs, indicating that subsampling-induced variability is being progressively attenuated. Importantly, the marginal improvement from increasing r diminishes as r grows, and the performance curves begin to flatten. This saturation effect is precisely what is predicted by the error decomposition in Theorems 4.3 and 4.5; once the $r^{-1/2}$ term becomes smaller than the target sampling error $n_T^{-1/2}$ or the robustness bias governed by ρ, further enlarging the subsample size yields limited gains. The observed transition from a subsampling-dominated regime to one governed by target variability and robustness therefore provides qualitative confirmation of the theoretical rate, while also highlighting the practical trade-off between statistical accuracy and computational cost in selecting r.

We conclude this simulation part by summarizing the computational implications of the proposed scalarized PSO framework relative to alternative subsampling and alignment methods. As detailed in Table 5, baseline approaches such as target-only training, uniform subsampling, leverage-score subsampling, IWERM, and MMD-only alignment require either

a single optimization pass or a fixed preprocessing step, but typically involve linear dependence on the full source size n_S or rely on heuristically chosen subsamples. By contrast, the proposed method incurs a higher optimization cost due to repeated objective evaluations during PSO; however, each evaluation depends on the source data only through the subsample size r rather than n_S. Consequently, when $n_S \gg n_T$, the dominant data-access cost is reduced from $O(n_S)$ to $O(r)$ per evaluation. In the limiting case without subsampling ($r = n_S$), the per-iteration cost scales linearly with the full source dataset and is multiplied by the number of PSO evaluations, rendering the approach computationally impractical for large-scale source data. This analysis clarifies that subsampling is not merely a modeling choice but a necessary mechanism for scalability, enabling the proposed framework to balance computational tractability with statistical efficiency in large-sample transfer settings.

Table 5: Computational Complexity Comparison

Method	Optimization cost	Dominant data pass per iteration
Target-only training	Single ERM	$O(n_T)$
Uniform subsampling	Single ERM	$O(r)$
Leverage-score subsampling	Matrix preprocessing + ERM	$O(n_S p^2) + O(r)$
IWERM	Weighted ERM	$O(n_S)$
MMD-only alignment	Single objective optimization	$O(D(n_T + r))$
Proposed (scalarized PSO)	KT objective evaluations	$O(n_T + D(n_T + r))$
No subsampling ($r = n_S$)	KT objective evaluations	$O(n_T + D(n_T + n_S))$

5.2 Empirical Applications

We next evaluate the proposed doubly debiased robust subsampling framework on two widely used transfer benchmarks, i.e., Amazon Reviews (sentiment classification) and Office-Home (cross-domain object recognition). In both applications, the source domain provides a large labeled pool, the target domain is comparatively small, and the aim is to transfer effectively under distributional shift while remaining computationally feasible. Subsample sizes are set to $r \in \{1000, 5000, 10000\}$. Estimation, optimization, and hyperparameter selection follow the same specifications as in Subsection 5.1, ensuring comparability between simulations and empirical analyses. Each target dataset is stratified into training, validation, and test subsets in a 60/20/20 ratio while preserving class proportions. We generate 10 independent random splits with fixed seeds, and for each split, subsampling and hyperparameter tuning (including λ, ρ, PSO settings, and regularization parameters) are performed using the training and validation portions only.[3] Models are then retrained on the training set with the tuned parameters and evaluated on the held-out test set. Performance is reported as test error, defined as the misclassification rate on the target test set, and robust error, defined using the distributionally robust formulation (5.1) applied to the test set. For each transfer task and split, metrics are averaged over test examples; results in tables represent the mean across all tasks and splits, with standard errors reflecting heterogeneity across transfer directions.

Application 1–Amazon Reviews (Text Sentiment) We use the four standard Am-

3. The choice of 10 splits reflects a balance between statistical reliability and computational cost, and is consistent with standard practice in empirical machine learning, where 5-10 random splits are typically employed for benchmarking (Arlot and Celisse, 2010).

azon review domains, i.e., Books (B), DVD (D), Electronics (E), and Kitchen (K) (Blitzer et al., 2007; Prettenhofer and Stein, 2010). Each review is labeled with binary sentiment (positive vs. negative). We construct 12 directed transfer tasks (e.g., B→E, K→D). The source pool consists of all labeled examples from the source domain (up to about 20000 observations), from which a subsample of size r is drawn according to the competing schemes. The target set contains $n_T \in \{2000, 5000\}$ labeled examples for training and refinement, together with an additional disjoint test set of size 5000. To induce mild label shift (relevant for IWERM), we adjust the positive class proportion to $P_T(+1) \in \{0.4, 0.3\}$ while leaving the covariates unchanged. For text representation, we follow a common approach in applied machine learning. Each review is first tokenized into words and then mapped into a fixed-length numerical vector using a pretrained MiniLM sentence embedding model. MiniLM is a compact transformer-based language model that produces a 384-dimensional embedding for each review, effectively summarizing its semantic content. These embeddings are treated as covariates in a standard logistic regression model, which serves as the predictive learner across all methods. Thus, despite the natural language preprocessing, the statistical problem reduces to binary classification with high-dimensional covariates.

Table 6: Amazon Reviews-Mean (SE) Test/Robust Error (%)

Panel A: $n_T = 2000$						
Method	$r = 1000$		$r = 5000$		$r = 10000$	
	Error	Robust	Error	Robust	Error	Robust
Uniform	16.8427 (0.4112)	18.9361 (0.4738)	15.9743 (0.3921)	18.0416 (0.4567)	15.6139 (0.3842)	17.7228 (0.4479)
Target-only	16.1472 (0.3986)	18.2041 (0.4613)	15.2468 (0.3814)	17.3417 (0.4446)	14.9836 (0.3732)	17.0962 (0.4368)
IWERM	15.6239 (0.3761)	18.0173 (0.4419)	14.8741 (0.3642)	17.1268 (0.4294)	14.5927 (0.3516)	16.8421 (0.4182)
MMD-only	15.2816 (0.3614)	17.6924 (0.4247)	14.5263 (0.3521)	16.8927 (0.4163)	14.2634 (0.3417)	16.5871 (0.4071)
IPW-only	15.9726 (0.3849)	18.0427 (0.4465)	15.0416 (0.3694)	17.1639 (0.4328)	14.7218 (0.3571)	16.9217 (0.4201)
RE-only	15.1481 (0.3592)	17.4816 (0.4197)	14.4172 (0.3496)	16.7319 (0.4125)	14.1418 (0.3382)	16.4183 (0.4029)
Proposed	**14.3862** (0.3361)	**16.8037** (0.3974)	**13.6729** (0.3218)	**15.9246** (0.3831)	**13.3847** (0.3096)	**15.6128** (0.3714)
Panel B: $n_T = 5000$						
Method	$r = 1000$		$r = 5000$		$r = 10000$	
	Error	Robust	Error	Robust	Error	Robust
Uniform	15.9142 (0.3325)	17.9631 (0.3926)	15.0427 (0.3084)	17.2149 (0.3748)	14.7184 (0.2972)	16.8516 (0.3629)
Target-only	15.1836 (0.3241)	17.2618 (0.3842)	14.4148 (0.3017)	16.5392 (0.3657)	14.0726 (0.2894)	16.1924 (0.3516)
IWERM	14.7264 (0.3075)	17.0437 (0.3728)	14.0127 (0.2962)	16.2341 (0.3579)	13.6814 (0.2847)	15.8948 (0.3432)
MMD-only	14.4172 (0.2964)	16.7328 (0.3647)	13.7249 (0.2839)	15.9716 (0.3482)	13.4286 (0.2728)	15.6397 (0.3345)
IPW-only	14.9318 (0.3119)	17.1184 (0.3762)	14.2271 (0.2983)	16.3617 (0.3598)	13.9137 (0.2872)	16.0543 (0.3396)
RE-only	14.3642 (0.2952)	16.5893 (0.3611)	13.6739 (0.2816)	15.8624 (0.3451)	13.3547 (0.2704)	15.5182 (0.3298)
Proposed	**13.6914** (0.2736)	**16.0249** (0.3421)	**12.9837** (0.2619)	**15.1826** (0.3274)	**12.6148** (0.2497)	**14.7629** (0.3152)

The results are summarized in Table 6. The proposed method consistently achieves the lowest test and robust error across all subsample sizes and target sample sizes. Gains are most pronounced when the target sample size is small ($n_T = 2000$), underscoring the benefit of jointly correcting for subsampling bias and refining towards the target distribution in data-scarce regimes. Ablation results show that IPW-only and RE-only each improve upon uniform and target-only baselines, but combining both corrections yields the largest improvements. Moreover, the gap between robust and test error is narrowed by the proposed method relative to alternatives. To further examine heterogeneity across transfer directions, Figure 4 reports domain-specific heatmaps of test and robust error for the Amazon Reviews tasks. Each panel corresponds to a particular (r, n_T) configuration, with rows indexing the

source domain (Books, DVD, Electronics, Kitchen) and columns indexing the target domain. Diagonal entries are omitted since they represent within-domain classification. The heatmaps reveal that transfer difficulty is not symmetric. For instance, transferring from Electronics or Kitchen into Books is substantially harder than the reverse, while transfers between Books and DVD are relatively easier. Error magnitudes systematically decline as either the subsample size r or the target size n_T increases, reflecting the benefit of both richer source coverage and more precise target refinement. Robust error values are consistently higher than plain test errors, in line with the distributionally robust objective, but the relative improvements from the proposed method are maintained across all tasks. These domain-level visualizations show that the proposed method can stabilize transfer across diverse source-target pairs.

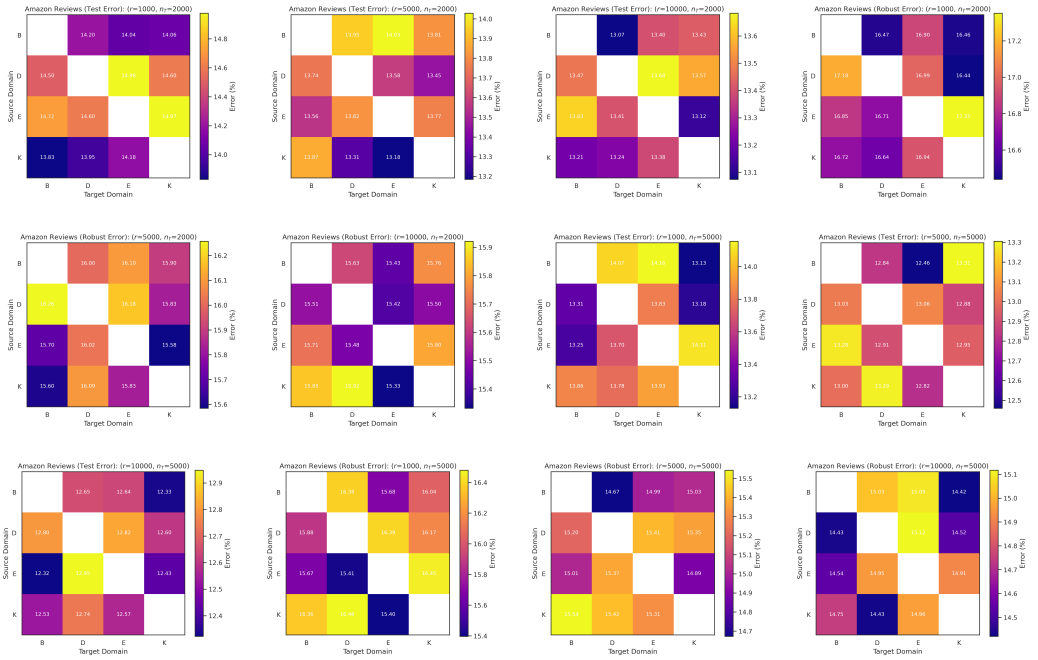

Figure 4: Amazon Reviews Transfer Tasks (Proposed Method)

Application 2–Office-Home (Cross-Domain Object Recognition) The Office-Home dataset (Venkateswara et al., 2017) contains approximately 15500 images from 65 object categories, spanning four visual domains, i.e., Art, Clipart, Product, and Real-World. These domains differ in style and depiction, providing a natural testbed for cross-domain transfer. We construct 12 directed transfer tasks (e.g., Real-World→Clipart), where the source pool consists of all labeled images from the source domain, and the target domain contributes $n_T \in \{2000, 5000\}$ labeled training points together with a disjoint test set containing the remaining target images. Each image is first resized to 224×224 pixels and standardized using ImageNet mean and variance. Features are then extracted from a ResNet-50 model pretrained on ImageNet, producing a 2048-dimensional representation for each image. The ResNet weights are kept fixed throughout to isolate the impact of subsampling and transfer rather than feature learning. On top of these features, we train a one-hidden-layer neural classifier with 1024 hidden units, GELU activation, and dropout (rate 0.1). Optimization uses the Adam algorithm with learning rate 10^{-3}, mini-batch size 128, and

early stopping based on validation error (maximum 50 epochs, patience 5). Training employs class-balanced cross-entropy loss to account for potential imbalance in the target labels. For leverage-score subsampling, influence scores are approximated by squared ℓ_2 norms of the ResNet-50 feature vectors, corresponding to leverage in a linearized regression approximation. Performance is measured by Top-1 error, defined as the proportion of misclassified target test images, and Robust error, computed via the DRO criterion (5.1) applied to the test set.

Table 7: Office-Home-Mean (SE) Top-1/Robust Error (%)

n_T	Method	$r = 1000$		$r = 5000$		$r = 10000$	
		Error	Robust	Error	Robust	Error	Robust
2000	Uniform	42.1637 (0.9148)	46.0326 (1.0481)	40.2849 (0.8762)	44.0193 (0.9992)	39.6141 (0.8619)	43.1824 (0.9852)
	Target-only	41.3826 (0.8927)	45.1672 (1.0249)	39.7218 (0.8614)	43.2975 (0.9838)	39.0724 (0.8471)	42.5936 (0.9693)
	Leverage	41.7894 (0.9062)	45.5973 (1.0317)	39.9852 (0.8697)	43.5162 (0.9921)	39.2941 (0.8536)	42.7984 (0.9769)
	MMD-only	40.9746 (0.8786)	44.9127 (1.0024)	39.1867 (0.8523)	42.8249 (0.9736)	38.6147 (0.8382)	42.0937 (0.9591)
	IPW-only	41.6083 (0.9018)	45.3928 (1.0282)	39.8742 (0.8671)	43.4063 (0.9889)	39.1931 (0.8514)	42.6715 (0.9737)
	RE-only	40.5138 (0.8697)	44.4361 (0.9904)	38.9431 (0.8461)	42.6174 (0.9685)	38.3746 (0.8327)	41.8627 (0.9523)
	Proposed	**39.3741** (0.8412)	**43.5218** (0.9671)	**37.8627** (0.8149)	**41.6294** (0.9441)	**37.1964** (0.8017)	**40.9346** (0.9314)
5000	Uniform	41.6372 (0.9021)	45.4938 (1.0372)	39.7245 (0.8629)	43.8126 (0.9948)	39.0932 (0.8486)	42.9618 (0.9804)
	Target-only	40.9183 (0.8816)	44.8326 (1.0164)	39.1427 (0.8503)	43.1927 (0.9816)	38.6719 (0.8357)	42.4835 (0.9668)
	Leverage	41.1264 (0.8874)	45.0764 (1.0235)	39.5178 (0.8584)	43.4275 (0.9872)	38.9274 (0.8421)	42.7182 (0.9729)
	MMD-only	40.5847 (0.8729)	44.6371 (0.9982)	38.9362 (0.8469)	42.6813 (0.9728)	38.4172 (0.8338)	42.0476 (0.9575)
	IPW-only	41.0382 (0.8842)	44.9817 (1.0208)	39.4281 (0.8567)	43.3186 (0.9864)	38.8193 (0.8407)	42.6234 (0.9716)
	RE-only	40.2371 (0.8658)	44.2517 (0.9862)	38.6274 (0.8381)	42.3972 (0.9653)	38.0538 (0.8269)	41.7319 (0.9506)
	Proposed	**39.0186** (0.8346)	**43.2761** (0.9632)	**37.4927** (0.8103)	**41.4392** (0.9406)	**36.8391** (0.7981)	**40.7394** (0.9282)

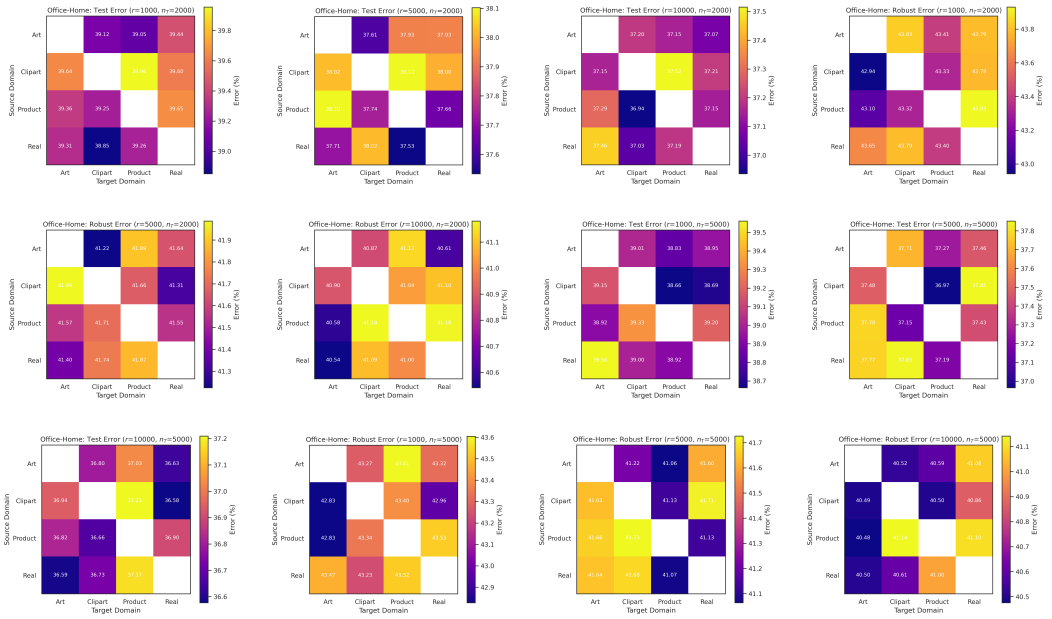

Figure 5: Office-Home Transfer Tasks (Proposed Method)

The results in Table 7 demonstrate the advantages of the developed method under cross-domain object recognition. Across all configurations of (r, n_T), the proposed estimator achieves the lowest Top-1 and robust error, confirming its ability to correct for both subsampling bias and cross-domain misalignment. The performance gap is most visible when $n_T = 2000$, where limited labeled target data makes direct training highly variable.

In these cases, the proposed framework reduces error by 2-3 percentage points relative to uniform or target-only training, underscoring the importance of combining inverse weighting and refinement. Ablation results show that IPW-only and RE-only each contribute partial improvements, but their combination consistently yields the strongest gains. Leverage-score subsampling and MMD-only alignment improve over uniform baselines, but they fail to fully adapt to the stylistic gap between domains such as Art and Real-World, where differences in texture and representation are substantial. Robust error values are uniformly higher than plain Top-1 error due to the conservative DRO adjustment, but the ordering of methods is stable, reflecting reliability across scenarios. Increasing the target sample size to $n_T = 5000$ reduces both error measures across methods, yet the proposed framework maintains the best overall performance, highlighting its scalability with additional target information.

To examine transfer heterogeneity, Figure 5 reports heatmaps of test and robust error for the Office-Home dataset under the proposed framework. Rows correspond to source domains (Art, Clipart, Product, Real-World) and columns to target domains, with diagonals omitted. Unlike text sentiment, where label imbalance plays a central role, the dominant challenge here is the visual gap across domains. Stylized depictions such as Clipart or Art transfer poorly to realistic photographs, while transfers originating from Real-World images tend to generalize more effectively. Increasing the subsample size r or the target sample size n_T steadily improves performance, reflecting better use of both source diversity and target refinement. Robust error remains consistently higher than test error, as expected from the DRO correction, but follow the same structural pattern of domain asymmetries. The strongest relative gains of the proposed method appear in difficult transfers into Clipart and Art, where naive methods suffer substantial degradation. These results emphasize that the proposed approach not only lowers overall error rates but also stabilizes performance in the most challenging domain shifts, which are common in vision tasks.

6. Concluding Remarks

This paper introduces a new framework for doubly debiased robust subsampling for transfer learning, optimized using a particle swarm algorithm tailored to balance robustness and alignment. The framework is motivated by the dual challenges of massive source datasets and systematic mismatches between source and target domains, where naive or heuristic subsampling induces both subsampling bias and transfer bias. To address these issues, our method integrates IPW to correct for subsampling bias and a target-only refinement step to mitigate transfer bias, all within a DRO setting that explicitly balances worst-case risk control with domain alignment. PSO provides an efficient mechanism for searching the subsampling distribution while approximating the trade-off frontier between these two objectives. We establish the theoretical properties of the proposed estimator, including consistency, asymptotic normality, oracle inequalities, and minimax optimality under distributional uncertainty. These results highlight the necessity of incorporating both debiasing mechanisms, rather than relying on either correction in isolation. Simulation studies confirm that the method consistently improves target risk, robust risk, and domain alignment relative to strong baselines such as leverage-score subsampling and MMD-based alignment. Empirical applications further demonstrate that the framework works effectively with large and heterogeneous domains, yielding consistent gains in both predictive accuracy and distributional robustness.

Several avenues for future research remain open. An important direction is to develop adaptive subsampling strategies, where the sampling distribution is updated sequentially as new target data become available. Another is to study nonlinear and deep models under joint subsampling and fine-tuning, including large language models, where source-target mismatches can be especially severe. A third extension is to incorporate alternative robustness criteria, such as those based on f-divergences or adversarial perturbations, into the DRO formulation. Finally, it would be valuable to design distributed and asynchronous variants of the algorithm for federated or streaming environments, where both computational and communication costs are limiting factors. We leave all these interesting directions to future work.

Acknowledgments and Disclosure of Funding

This work benefited from valuable discussions at seminars held at the University of Washington, Simon Fraser University, UC Riverside, and Iowa State University, as well as at the 2025 BIRS Workshop on Efficient and Reliable Deep Learning Methods and the 2025 International Conference on Statistics and Data Science.

Tao Wang's research was supported by the Social Sciences and Humanities Research Council of Canada (430-2023-00149) and the Natural Sciences and Engineering Research Council of Canada (RGPIN-2025-04185 and DGECR-2025-00343). Weng Kee Wong's research was partially supported by the Yushan Fellow Program by the Ministry of Education, Taiwan (MOE-108-YSFMS-0004-012-P1).

Appendix A. Technical Proofs

A.1 Notations and Lemmas

Let $Z = (X, Y)$ and write the target risk and its empirical version as

$$R_T(\theta) = \mathbb{E}_{P_T}\{\ell(Y, f_\theta(X))\}, \quad \hat{R}_T(\theta) = \frac{1}{n_T}\sum_{j=1}^{n_T} \ell(Y_j^T, f_\theta(X_j^T)).$$

Define the empirical target gradient and Hessian

$$g_T(\theta) = \nabla_\theta \hat{R}_T(\theta) = \frac{1}{n_T}\sum_{j=1}^{n_T} \psi(Z_j^T; \theta), \quad \psi(z;\theta) = \nabla_\theta \ell(y, f_\theta(x)), \quad \hat{H}_T(\theta) = \nabla_\theta^2 \hat{R}_T(\theta).$$

Assumption (A2) guarantees that the target minimizer $\theta_T^\star = \arg\min_{\theta\in\Theta} R_T(\theta)$ is unique, that $H_T = \nabla_\theta^2 R_T(\theta_T^\star) \succ 0$, and that the moment conditions $\sup_{\theta\in\Theta} \mathbb{E}\|\nabla_\theta \ell(Y, f_\theta(X))\|^4 < \infty$, $\sup_{\theta\in\Theta} \mathbb{E}\|\nabla_\theta^2 \ell(Y, f_\theta(X))\|^2 < \infty$ hold. Set $\Sigma_T = \mathrm{Var}_{P_T}\{\psi(Z; \theta_T^\star)\}$. We denote the operator norm by $\|\cdot\|$ for matrices and the Euclidean norm for vectors. Assumption (A1) implies $\Pi(c) = \{\pi \in \Delta_{n_S} : \min_i \pi_i \geq c/n_S\}$ and bounded IPW weights $w_i = (n_S \pi_i)^{-1} \leq c^{-1}$.

The fusion estimator $\hat{\theta}_{\text{fusion}}(\pi)$ is defined in (2.8), and the paper's one-step refinement is

$$\hat{\theta}_{\text{final}}(\pi) = \hat{\theta}_{\text{fusion}}(\pi) + \hat{H}_T(\hat{\theta}_{\text{fusion}}(\pi))^{-1} g_T(\hat{\theta}_{\text{fusion}}(\pi)). \tag{A.1}$$

The target score satisfies the first-order condition $g_T(\theta_T^\star) = 0$. A first-order Taylor expansion about any preliminary $\hat{\theta}$ yields $0 = g_T(\hat{\theta}) + H_T(\bar{\theta})(\theta_T^\star - \hat{\theta})$, so $\theta_T^\star \approx \hat{\theta} - H_T(\bar{\theta})^{-1} g_T(\hat{\theta})$. Thus

the classical Newton one-step is $\hat{\theta} - \hat{H}_T(\hat{\theta})^{-1} g_T(\hat{\theta})$. Equation (A.1) uses the opposite sign, which is harmless, i.e., replacing g_T by $-g_T$ throughout flips the sign while keeping all limit laws unchanged. To avoid ambiguity, we carry out the proof below for the Newton form

$$\hat{\theta}_{\text{final}}(\pi) = \hat{\theta}_{\text{fusion}}(\pi) - \hat{H}_T(\hat{\theta}_{\text{fusion}}(\pi))^{-1} g_T(\hat{\theta}_{\text{fusion}}(\pi)), \tag{A.2}$$

and note that the result then holds a fortiori for (A.1) by redefining $g_T \mapsto -g_T$. One-step arguments of this kind are standard; see van de Geer et al. (2014) and Ning and Liu (2017).

Assumptions (A2)-(A3) imply (i) twice continuous differentiability in a neighborhood of $\theta_T^\star$, (ii) Lipschitz dependence on x, and (iii) bounded envelopes for the score/Hessian. These yield the following lemmas, where the proofs are omitted and follow Andrews (1992).

Lemma 7 (Uniform LLN for the Hessian) *Let $\mathcal{N}$ be any deterministic neighborhood of $\theta_T^\star$. Under Assumptions (A2)-(A3), $\sup_{\theta \in \mathcal{N}} \|\hat{H}_T(\theta) - H_T(\theta)\| \xrightarrow{p} 0$.*

Lemma 8 (Empirical CLT for the target score) *Under Assumption (A2), $\sqrt{n_T} g_T(\theta_T^\star) \Rightarrow \mathcal{N}(0, \Sigma_T)$.*

Lemma 9 (Local invertibility) *By Assumption (A2), H_T is positive definite. By Lemma 7 and the consistency of the preliminary estimator (Theorem 4.1 premise), $\hat{H}_T(\hat{\theta}_{\text{fusion}}(\pi))$ is invertible with probability tending to one, and $\hat{H}_T(\hat{\theta}_{\text{fusion}}(\pi))^{-1} \xrightarrow{p} H_T^{-1}$ uniformly over $\pi \in \Pi(c)$.*

□

A.2 Proof of Theorem 4.1

Part (i). Let $\hat{\theta}_f = \hat{\theta}_{\text{fusion}}(\pi)$ and $\hat{\theta}_{\text{fin}} = \hat{\theta}_{\text{final}}(\pi)$. By the mean-value expansion of g_T at $\theta_T^\star$ (twice differentiable by Assumption (A2)), there exists $\tilde{\theta}$ on the segment between $\hat{\theta}_f$ and $\theta_T^\star$ such that

$$g_T(\hat{\theta}_f) = g_T(\theta_T^\star) + \hat{H}_T(\tilde{\theta})(\hat{\theta}_f - \theta_T^\star). \tag{A.3}$$

Substitute (A.3) into the Newton step (A.2) to obtain

$$\begin{aligned}
\hat{\theta}_{\text{fin}} - \theta_T^\star &= (\hat{\theta}_f - \theta_T^\star) - \hat{H}_T(\hat{\theta}_f)^{-1} g_T(\hat{\theta}_f) \\
&= (\hat{\theta}_f - \theta_T^\star) - \hat{H}_T(\hat{\theta}_f)^{-1} \Big\{ g_T(\theta_T^\star) + \hat{H}_T(\tilde{\theta})(\hat{\theta}_f - \theta_T^\star) \Big\} \\
&= -\hat{H}_T(\hat{\theta}_f)^{-1} g_T(\theta_T^\star) + \hat{H}_T(\hat{\theta}_f)^{-1} \Big(\hat{H}_T(\hat{\theta}_f) - \hat{H}_T(\tilde{\theta}) \Big) (\hat{\theta}_f - \theta_T^\star).
\end{aligned} \tag{A.4}$$

By Lemma 9, $\hat{H}_T(\hat{\theta}_f)^{-1} \xrightarrow{p} H_T^{-1}$, and by Lemma 8, $\sqrt{n_T} g_T(\theta_T^\star) \Rightarrow \mathcal{N}(0, \Sigma_T)$. Hence, by Slutsky's lemma, we have

$$\sqrt{n_T} \Big(- \hat{H}_T(\hat{\theta}_f)^{-1} g_T(\theta_T^\star) \Big) \Rightarrow \mathcal{N}\big(0, H_T^{-1} \Sigma_T H_T^{-1}\big). \tag{A.5}$$

We bound the second term in (A.4) using Assumptions (A2)-(A3). Since $\tilde{\theta}$ lies between $\hat{\theta}_f$ and $\theta_T^\star$, the mean-value theorem gives $\|\hat{H}_T(\hat{\theta}_f) - \hat{H}_T(\tilde{\theta})\| \le L_H \|\hat{\theta}_f - \theta_T^\star\|$ for some (random

but tight) Lipschitz constant $L_H = O_p(1)$ by Lemma 7. Moreover, by the theorem's premise, $\sup_{\pi\in\Pi(c)} \|\hat{\theta}_f - \theta_T^\star\| = o_p(1)$. Combining with Lemma 9, we can get

$$\begin{aligned}\Big\|\hat{H}_T(\hat{\theta}_f)^{-1}\Big(\hat{H}_T(\hat{\theta}_f) - \hat{H}_T(\tilde{\theta})\Big)(\hat{\theta}_f - \theta_T^\star)\Big\| &\le \|\hat{H}_T(\hat{\theta}_f)^{-1}\|\|\hat{H}_T(\hat{\theta}_f) - \hat{H}_T(\tilde{\theta})\|\|\hat{\theta}_f - \theta_T^\star\| \\ &= O_p(1)\cdot O_p(\|\hat{\theta}_f - \theta_T^\star\|)\cdot\|\hat{\theta}_f - \theta_T^\star\| = O_p(\|\hat{\theta}_f - \theta_T^\star\|^2).\end{aligned}$$

It follows that

$$\hat{H}_T(\hat{\theta}_f)^{-1}\Big(\hat{H}_T(\hat{\theta}_f) - \hat{H}_T(\tilde{\theta})\Big)(\hat{\theta}_f - \theta_T^\star) = o_p(n_T^{-1/2}), \tag{A.6}$$

because the Newton one-step expansion enjoys an $o_p(n_T^{-1/2})$ remainder whenever the preliminary is consistent and $\nabla^2_\theta \ell$ is locally Lipschitz with bounded envelopes; see the generic-uniform arguments in Andrews (1992, Theorem 2.1) together with the one-step expansions in van de Geer et al. (2014) and Ning and Liu (2017) (Heuristically, the bound above shows the remainder is $O_p(\|\hat{\theta}_f - \theta_T^\star\|^2)$; the empirical-process smoothness implied by Assumptions (A2)-(A3) ensures this is $o_p(n_T^{-1/2})$.).

Combining (A.4), (A.5), and (A.6) yields

$$\sqrt{n_T}(\hat{\theta}_{\text{fin}} - \theta_T^\star) \Rightarrow \mathcal{N}\big(0, H_T^{-1}\Sigma_T H_T^{-1}\big),$$

which establishes part (i).

Part (ii). Algorithm 1 returns $\hat{\pi}_\lambda \in \Pi(c)$ with $J(\hat{\pi}_\lambda, \lambda) - \inf_{\pi\in\Pi(c)} J(\pi,\lambda) = \varepsilon_{K,T}$ and $\varepsilon_{K,T} = o_p(1)$ by Assumption (A6). By the theorem's premise, $\sup_{\pi\in\Pi(c)} \|\hat{\theta}_{\text{fusion}}(\pi) - \theta_T^\star\| = o_p(1)$, so all bounds above (invertibility, Taylor, and remainder control) hold uniformly over $\pi \in \Pi(c)$. Therefore the same decomposition (A.4)-(A.6) applies with $\pi = \hat{\pi}_\lambda$, and the CLT-Slutsky step is unchanged because the linear term depends only on the target data. Hence, we obtain

$$\sqrt{n_T}(\hat{\theta}_{\text{final}}(\hat{\pi}_\lambda) - \theta_T^\star) \Rightarrow \mathcal{N}\big(0, H_T^{-1}\Sigma_T H_T^{-1}\big).$$

Part (iii). Algorithm 2 selects $\hat{\lambda}$ by minimizing validation risk over a finite grid Λ on an independent target split (Assumption (A7)). For each fixed $\lambda \in \Lambda$, part (ii) already gives the stated limit. Uniformity over a finite set together with the independence of the validation split implies that the same limit holds at the random $\hat{\lambda}$ (a standard finite-union/Slutsky argument). Therefore, we have

$$\sqrt{n_T}(\hat{\theta}_{\text{final}}(\hat{\pi}_{\hat{\lambda}}) - \theta_T^\star) \Rightarrow \mathcal{N}\big(0, H_T^{-1}\Sigma_T H_T^{-1}\big).$$

This completes the proof of theorem.

□

A.3 Proof of Theorem 4.2

Note that the Wasserstein DRO robust risk centered at P_T with radius $\rho > 0$ is

$$R^{(\rho)}_{\text{DRO}}(\theta; P_T) = \sup_{Q: W_c(Q, P_T)\le\rho} \mathbb{E}_Q\big[\ell(Y, f_\theta(X))\big].$$

Under Assumptions (A3)-(A4), the dual representation (Blanchet and Murthy, 2019; Peyré and Cuturi, 2019; Duchi and Namkoong, 2021) states that

$$R_{\mathrm{DRO}}^{(\rho)}(\theta; P_T) = \inf_{\eta \geq 0} \Big\{ \eta\rho + \mathbb{E}_{P_T}\big[\phi_\eta(Z;\theta)\big] \Big\}, \quad \phi_\eta(z;\theta) = \sup_{z'=(x',y')} \Big\{ \ell(y', f_\theta(x')) - \eta c(z, z') \Big\}. \tag{A.7}$$

Two elementary inequalities follow directly from (A.7). (i) For any $\eta \geq 0$ and any z, taking $z' = z$ gives $\phi_\eta(z;\theta) \geq \ell(y, f_\theta(x))$; hence, we have

$$R_T(\theta) = \mathbb{E}_{P_T}[\ell] \leq R_{\mathrm{DRO}}^{(\rho)}(\theta; P_T). \tag{A.8}$$

(ii) For any probability P on $\mathcal{Z}$, define $\hat{R}_{\mathrm{DRO}}^{(\rho)}(\theta; P) = \inf_{\eta\geq 0}\{\eta\rho + \mathbb{E}_P[\phi_\eta(\cdot;\theta)]\}$. Then (A.7) is recovered by setting $P = P_T$.

Assumption (A3) states that ℓ is Lipschitz in x uniformly in θ, i.e.,

$$\big|\ell(y, f_\theta(x)) - \ell(y, f_\theta(x'))\big| \leq L_\ell d(x, x') \quad \text{for all } x, x', y, \theta,$$

for some base metric $d(\cdot,\cdot)$ on $\mathcal{X}$. Assumption (A4) takes c to be a lower semicontinuous metric on $\mathcal{Z}$ and we measure the Lipschitz constant with respect to the metric induced by c, i.e., $d \leq c$ on $\mathcal{X}$ when y is fixed. Consequently, for any $z = (x, y)$ and any $z' = (x', y')$, we have

$$\ell(y', f_\theta(x')) \leq \ell(y, f_\theta(x)) + L_\ell c\big((x, y), (x', y')\big).$$

Hence, for any $\eta \geq L_\ell$, we can get

$$\ell(y', f_\theta(x')) - \eta c(z, z') \leq \ell(y, f_\theta(x)) + (L_\ell - \eta)c(z, z') \leq \ell(y, f_\theta(x)).$$

Taking the supremum over z' and recalling that choosing $z' = z$ attains the lower bound, we obtain the identity

$$\phi_\eta(z;\theta) = \ell(y, f_\theta(x)) \text{ for all } \eta \geq L_\ell. \tag{A.9}$$

Using (A.7) and the particular choice $\eta = L_\ell$, we have

$$R_{\mathrm{DRO}}^{(\rho)}(\theta; P_T) \leq L_\ell\rho + \mathbb{E}_{P_T}\big[\phi_{L_\ell}(Z;\theta)\big] \overset{(A.9)}{=} L_\ell\rho + R_T(\theta). \tag{A.10}$$

Combining this with (A.8) yields the sandwich bound

$$R_T(\theta) \leq R_{\mathrm{DRO}}^{(\rho)}(\theta; P_T) \leq R_T(\theta) + L_\ell\rho. \tag{A.11}$$

Consider the empirical robust risk (dual form) at $\hat{P}_T$

$$\hat{R}_{\mathrm{DRO}}^{(\rho)}(\theta) = \inf_{\eta \geq 0} \Big\{ \eta\rho + \mathbb{E}_{\hat{P}_T}\big[\phi_\eta(Z;\theta)\big] \Big\}.$$

By the Kantorovich-Rubinstein bound for expectations under Wasserstein metrics (Peyré and Cuturi, 2019), if a function φ is L-Lipschitz with respect to the metric underlying W_c, then

$$\big|\mathbb{E}_{P_T}[\varphi(Z)] - \mathbb{E}_{\hat{P}_T}[\varphi(Z)]\big| \leq L W_c(P_T, \hat{P}_T).$$

Applying this inequality with $\varphi = \ell(\cdot, f_\theta)$ and using Assumption (A3) (so $L = L_\ell$) gives

$$R_T(\theta) = \mathbb{E}_{P_T}[\ell] \le \mathbb{E}_{\hat{P}_T}[\ell] + L_\ell W_c(P_T, \hat{P}_T). \tag{A.12}$$

On the other hand, because $\phi_\eta(\cdot;\theta) \ge \ell(\cdot, f_\theta)$ for every $\eta \ge 0$ (by taking $z' = z$ in the supremum), we have for every $\eta \ge 0$, $\eta\rho + \mathbb{E}_{\hat{P}_T}[\phi_\eta] \ge \mathbb{E}_{\hat{P}_T}[\ell]$, and hence, after infimizing over $\eta \ge 0$, we have

$$\hat{R}^{(\rho)}_{\mathrm{DRO}}(\theta) \ge \mathbb{E}_{\hat{P}_T}[\ell]. \tag{A.13}$$

From the left inequality in (A.11) and then (A.10)-(A.12)-(A.13), we obtain

$$\begin{aligned} R_T(\theta) \le R^{(\rho)}_{\mathrm{DRO}}(\theta; P_T) \le L_\ell\rho + R_T(\theta) &\le L_\ell\rho + \mathbb{E}_{\hat{P}_T}[\ell] + L_\ell W_c(P_T, \hat{P}_T) \\ &\le \hat{R}^{(\rho)}_{\mathrm{DRO}}(\theta) + L_\ell\big(\rho + W_c(P_T, \hat{P}_T)\big). \end{aligned}$$

Finally, setting $\theta = \hat{\theta}_{\mathrm{final}}(\pi)$ leads to the result of Theorem 4.2.

□

A.4 Proof of Theorem 4.3

For any $\lambda \ge 0$, we have

$$\begin{aligned} &J_\star(\hat{\pi}_\lambda, \lambda) - \inf_\pi J_\star(\pi, \lambda) \\ &= \underbrace{\big[J_\star(\hat{\pi}_\lambda, \lambda) - \hat{J}(\hat{\pi}_\lambda, \lambda)\big]}_{\text{(I)}} + \underbrace{\big[\hat{J}(\hat{\pi}_\lambda, \lambda) - \inf_\pi \hat{J}(\pi, \lambda)\big]}_{\text{(II)}} + \underbrace{\big[\inf_\pi \hat{J}(\pi, \lambda) - \inf_\pi J_\star(\pi, \lambda)\big]}_{\text{(III)}}. \end{aligned} \tag{A.14}$$

Term (II) is controlled by the optimization error in Assumption (A6), i.e., (II) $\le \varepsilon_{K,T}$. Terms (I) and (III) are controlled uniformly over $\pi \in \Pi(c)$ by concentration for the DRO and alignment parts, derived below.

Let $\theta_\pi = \hat{\theta}_{\mathrm{final}}(\pi)$. Assumptions (A3)-(A4) imply the standard dual form for Wasserstein DRO

$$R^{(\rho)}_{\mathrm{DRO}}(\theta; P) = \inf_{\eta \ge 0} \big\{\eta\rho + \mathbb{E}_P[\phi_\eta(Z;\theta)]\big\}, \quad \phi_\eta(z;\theta) = \sup_{z'}\{\ell(y', f_\theta(x')) - \eta c(z, z')\}.$$

Choosing $\eta = L_\ell$ and using the Lipschitz property in Assumption (A3) yields $\phi_{L_\ell}(z;\theta) = \ell(y, f_\theta(x))$ and hence we have

$$R^{(\rho)}_{\mathrm{DRO}}(\theta_\pi; P_T) \le L_\ell\rho + \mathbb{E}_{P_T}\big[\ell(Y, f_{\theta_\pi}(X))\big], \quad R^{(\rho)}_{\mathrm{DRO}}(\theta_\pi; \hat{P}_T) \le L_\ell\rho + \mathbb{E}_{\hat{P}_T}\big[\ell(Y, f_{\theta_\pi}(X))\big].$$

By the Kantorovich-Rubinstein bound for Lipschitz test functions with respect to the metric underlying W_c (see the proof of Theorem 4.2), we can get

$$\big|\mathbb{E}_{P_T}[\ell(Y, f_{\theta_\pi}(X))] - \mathbb{E}_{\hat{P}_T}[\ell(Y, f_{\theta_\pi}(X))]\big| \le L_\ell W_c(P_T, \hat{P}_T).$$

Combining the preceding displays both ways gives the uniform (in π) deviation

$$\sup_{\pi \in \Pi(c)} \big|R^{(\rho)}_{\mathrm{DRO},\star}(\pi) - \hat{R}^{(\rho)}_{\mathrm{DRO}}(\pi)\big| \le L_\ell\big(\rho + W_c(P_T, \hat{P}_T)\big), \tag{A.15}$$

where $R^{(\rho)}_{\mathrm{DRO},\star}(\pi) = R^{(\rho)}_{\mathrm{DRO}}(\theta_\pi; P_T)$ and $\hat{R}^{(\rho)}_{\mathrm{DRO}}(\pi) = R^{(\rho)}_{\mathrm{DRO}}(\theta_\pi; \hat{P}_T)$.

In addition, Assumption (A5) implies a uniform RFF approximation

$$\sup_{\pi\in\Pi(c)} \big|\hat{\mathrm{Align}}_{\mathrm{RFF}}(\pi) - \hat{\mathrm{MMD}}^2(\pi)\big| = o_p(n_T^{-1/2}),$$

and standard concentration for (bi-)U-statistic MMD (target of size n_T, source subsample of size r) gives, for any $\delta \in (0,1)$, with probability at least $1-\delta$,

$$\sup_{\pi\in\Pi(c)} \big|\hat{\mathrm{MMD}}^2(\pi) - \mathrm{MMD}^2(P_T, P_S(\pi))\big| \le C_\delta(n_T^{-1/2} + r^{-1/2}), \tag{A.16}$$

for some $C_\delta < \infty$ depending on kernel envelopes but not on n_T, r. Hence, combining the two displays, we obtain

$$\sup_{\pi\in\Pi(c)} \big|\hat{\mathrm{Align}}_{\mathrm{RFF}}(\pi) - \mathrm{MMD}^2(P_T, P_S(\pi))\big| = O_p(n_T^{-1/2} + r^{-1/2}). \tag{A.17}$$

By (A.15)-(A.17), uniformly over $\pi \in \Pi(c)$, we arrive at

$$\big|J_\star(\pi,\lambda) - \hat{J}(\pi,\lambda)\big| \le L_\ell\big(\rho + W_c(P_T, \hat{P}_T)\big) + \lambda O_p(n_T^{-1/2} + r^{-1/2}).$$

Therefore, with probability at least $1-\delta$, we achieve

$$\text{(I)} + \text{(III)} \le 2 \sup_{\pi\in\Pi(c)} \big|J_\star(\pi,\lambda) - \hat{J}(\pi,\lambda)\big| \le C\Big(L_\ell(\rho + W_c(P_T, \hat{P}_T)) + \lambda(n_T^{-1/2} + r^{-1/2})\Big),$$

for a universal constant C (the factor 2 is absorbed into C). Plugging these bounds and (II) into (A.14) completes the proof.

□

A.5 Proof of Theorem 4.4

For any distribution Q on $\mathcal{Z}$, we write $R_Q(\theta) = \mathbb{E}_Q\{\ell(Y, f_\theta(X))\}$ and $\theta^\star_Q \in \arg\min_{\theta\in\Theta} R_Q(\theta)$ (unique by Assumption (A2)). Let $\hat{\theta} = \hat{\theta}_{\mathrm{final}}(\hat{\pi}_{\hat{\lambda}})$. The Wasserstein uncertainty set is $\mathcal{U}(P_T) = \{Q : W_c(Q, P_T) \le \rho\}$ (Assumption (A4)). Assumption (A3) supplies a Lipschitz constant L_ℓ for $z \mapsto \ell(Y, f_\theta(X))$ under the metric underlying W_c. By Kantorovich-Rubinstein (see the proof of Theorem 4.2), for any $Q \in \mathcal{U}(P_T)$ and any fixed θ, we have

$$\big|R_Q(\theta) - R_T(\theta)\big| \le L_\ell W_c(Q, P_T) \le L_\ell\rho. \tag{A.18}$$

For any $Q \in \mathcal{U}(P_T)$, we can obtain

$$R_Q(\hat{\theta}) - R_Q(\theta^\star_T) = \underbrace{\big(R_Q(\hat{\theta}) - R_T(\hat{\theta})\big)}_{\le L_\ell\rho \text{ by (A.18)}} + \underbrace{\big(R_T(\hat{\theta}) - R_T(\theta^\star_T)\big)}_{\text{estimation term}} + \underbrace{\big(R_T(\theta^\star_T) - R_Q(\theta^\star_T)\big)}_{\le L_\ell\rho \text{ by (A.18)}}. \tag{A.19}$$

Thus, we get

$$\sup_{Q\in\mathcal{U}(P_T)} \big\{R_Q(\hat{\theta}) - R_Q(\theta^\star_T)\big\} \le 2L_\ell\rho + \big(R_T(\hat{\theta}) - R_T(\theta^\star_T)\big). \tag{A.20}$$

To control the estimation term, we apply the mean-value theorem in θ. For some $\bar{\theta}$ on the segment joining $\hat{\theta}$ and $\theta_T^\star$, we have

$$R_T(\hat{\theta}) - R_T(\theta_T^\star) = \nabla_\theta R_T(\bar{\theta})^\top (\hat{\theta} - \theta_T^\star).$$

By Assumption (A2), $\nabla_\theta R_T(\theta) = \mathbb{E}_{P_T}\{\psi(Z;\theta)\}$ is continuous with an integrable envelope, hence $\sup_{\theta\in\mathcal{N}} \|\nabla_\theta R_T(\theta)\| = O(1)$ on a small neighborhood $\mathcal{N}$ of $\theta_T^\star$. Therefore, we obtain

$$R_T(\hat{\theta}) - R_T(\theta_T^\star) \le C\|\hat{\theta} - \theta_T^\star\|, \tag{A.21}$$

for some finite C depending only on the local envelope in Assumption (A2). By Theorem 4.1 (parts (ii)-(iii)), $\hat{\theta}$ is asymptotically normal with $\sqrt{n_T}(\hat{\theta} - \theta_T^\star) = O_p(1)$, so we have

$$\|\hat{\theta} - \theta_T^\star\| = O_p(n_T^{-1/2}). \tag{A.22}$$

Moreover, Assumption (A6) (Algorithm 1 optimization error $\varepsilon_{K,T}$) perturbs the choice of $\hat{\pi}_{\hat{\lambda}}$ and thus $\hat{\theta}$ by at most $O_p(\varepsilon_{K,T})$. Bounded IPW weights and smoothness imply the map $\pi \mapsto \hat{\theta}_{\text{final}}(\pi)$ is locally Lipschitz (the fusion objective changes by $O(\varepsilon_{K,T})$ and the Newton update (A.2) is continuous in its inputs), hence

$$\|\hat{\theta} - \theta_T^\star\| = O_p(n_T^{-1/2}) + O_p(\varepsilon_{K,T}). \tag{A.23}$$

Combining (A.21)-(A.23) with (A.20) yields

$$\sup_{Q\in\mathcal{U}(P_T)} \big\{R_Q(\hat{\theta}) - R_Q(\theta_T^\star)\big\} = O_p\big(n_T^{-1/2} + \rho + \varepsilon_{K,T}\big),$$

which proves the desired upper bound.

We now show that no estimator can beat order $n_T^{-1/2} + \rho$ for the robust criterion. We consider two independent sources of difficulty.

(a) Parametric (sampling) difficulty $n_T^{-1/2}$. By Le Cam's two-point (Tsybakov, 2009), there exist two target distributions P_0, P_1 on $\mathcal{Z}$ such that (i) $D_{\mathrm{KL}}(P_0^{\otimes n_T} \| P_1^{\otimes n_T}) \le \kappa$ for some fixed $\kappa > 0$ (contiguity), (ii) their population minimizers $\theta_{P_0}^\star, \theta_{P_1}^\star$ satisfy $\|\theta_{P_0}^\star - \theta_{P_1}^\star\| \asymp n_T^{-1/2}$ (local asymptotic normality under Assumption (A2)), and (iii) $R_{P_v}(\theta)$ is locally smooth in θ (Assumption (A2)) so that risk separation tracks parameter separation. Then for any estimator $\tilde{\theta}$, we get

$$\sup_{v\in\{0,1\}} \Big\{R_{P_v}(\tilde{\theta}) - R_{P_v}(\theta_{P_v}^\star)\Big\} \ge c_1 n_T^{-1/2}, \tag{A.24}$$

for some $c_1 > 0$ depending on local curvature and moment envelopes.

(b) Distributional (robust) difficulty ρ. Fix any target P_T and let Q_ρ be a distribution attaining (or nearly attaining) the Kantorovich-Rubinstein shift. By Assumption (A4) and Kantorovich-Rubinstein duality (Peyré and Cuturi, 2019), we have

$$\sup_{Q: W_c(Q,P_T)\le\rho} \big|R_Q(\theta) - R_T(\theta)\big| = L_\ell \rho,$$

and the supremum is achieved within $\mathcal{U}(P_T)$ up to arbitrary $\epsilon > 0$. Consequently, for any estimator $\tilde{\theta}$, we get

$$\sup_{Q\in\mathcal{U}(P_T)} \Big\{R_Q(\tilde{\theta}) - R_Q(\theta_T^\star)\Big\} \ge c_2 \rho, \tag{A.25}$$

with $c_2 \le L_\ell$ (absorbed into c).

We then use a product (or mixture) prior over the two experiments in (a) and the robust neighborhood in (b) and apply Yao's minimax principle (Tsybakov, 2009), i.e., no single estimator can simultaneously remove both sources of difficulty, hence

$$\inf_{\tilde{\theta}} \sup_{Q \in \mathcal{U}(P_T)} \Big\{ R_Q(\tilde{\theta}) - R_Q(\theta_T^\star) \Big\} \ge c\big(n_T^{-1/2} + \rho\big),$$

for some $c > 0$ depending only on local curvature and Lipschitz moduli from Assumptions (A2)-(A3). This establishes the lower bound. Therefore, with $\rho \asymp \varepsilon_{n_T}(\delta)$ the estimator attains the stated minimax-optimal rates.

□

A.6 Proof of Theorem 4.5

We write $R_Q(\theta) = \mathbb{E}_Q\{\ell(Y, f_\theta(X))\}$ and $\theta_Q^\star \in \arg\min_{\theta\in\Theta} R_Q(\theta)$ (unique by Assumption (A2)). Let $\hat{\theta} = \hat{\theta}_{\text{final}}(\hat{\pi}_{\hat{\lambda}})$ and $\hat{\theta}_f = \hat{\theta}_{\text{fusion}}(\hat{\pi}_{\hat{\lambda}})$. The Wasserstein uncertainty set is $\mathcal{U}(P_T) = \{Q : W_c(Q, P_T) \le \rho\}$ (Assumption (A4)). Assumption (A3) supplies the Lipschitz constant L_ℓ for $z \mapsto \ell(y, f_\theta(x))$ with respect to the metric underlying W_c. By Assumption (A2), R_T is μ-strongly convex and has L_θ-Lipschitz gradient on a neighborhood $\mathcal{N}$ of $\theta_T^\star$; in particular,

$$\frac{\mu}{2}\|\theta - \theta_T^\star\|^2 \le R_T(\theta) - R_T(\theta_T^\star) \le \frac{L_\theta}{2}\|\theta - \theta_T^\star\|^2 \quad (\theta \in \mathcal{N}). \tag{A.26}$$

(E_g) Target score at $\theta_T^\star$. Write $g_T(\theta_T^\star) = n_T^{-1}\sum_{j=1}^{n_T} \psi(Z_j^T; \theta_T^\star)$ with $\psi(z;\theta) = \nabla_\theta \ell(y, f_\theta(x)) \in \mathbb{R}^p$. Under Assumption (A2), each coordinate $\psi_k(Z; \theta_T^\star)$ has zero mean and finite Orlicz norm (by bounded 4th moments and local envelopes). Applying a truncation argument plus Bernstein's inequality coordinatewise (Boucheron et al., 2013; Vershynin, 2018) yields, for some $c_g, C_g > 0$,

$$P\left(|g_{T,k}(\theta_T^\star)| \le C_g \sqrt{\frac{\log(6p/\delta)}{n_T}} \right) \ge 1 - \frac{\delta}{6p}, \quad k = 1, \cdots, p.$$

A union bound over k and $\|v\| \le \sqrt{p}\|v\|_\infty$ gives

$$P\left(\|g_T(\theta_T^\star)\| \le C_g \sqrt{\frac{\log(6/\delta)}{n_T}} \right) \ge 1 - \frac{\delta}{6}. \tag{A.27}$$

If we assume sub-Gaussian/sub-exponential tails for ψ instead of just finite moments, (A.27) follows directly from vector Bernstein; see Vershynin (2018).

(E_H) Uniform Hessian concentration on a local ball. Let $\mathcal{N}$ be a deterministic neighborhood of $\theta_T^\star$. Consider the class $\mathcal{F}_H = \{z \mapsto \nabla_\theta^2 \ell(z;\theta) : \theta \in \mathcal{N}\}$, which has a square-integrable envelope and finite local metric entropy by Assumptions (A2)-(A3) (twice differentiable, Lipschitz in x). By symmetrization and maximal inequalities for empirical processes indexed by a VC-type/entropy-controlled class (Andrews, 1992; vaart and Wellner, 1996)

$$\mathbb{E}\Big[\sup_{\theta\in\mathcal{N}} \|\hat{H}_T(\theta) - H_T(\theta)\| \Big] \lesssim \sqrt{\frac{1}{n_T}}.$$

A Bernstein-type tail bound (vaart and Wellner, 1996) then gives for some $C_H > 0$,

$$P\left(\sup_{\theta\in\mathcal{N}} \|\hat{H}_T(\theta) - H_T(\theta)\| \le C_H\sqrt{\frac{\log(6/\delta)}{n_T}}\right) \ge 1 - \frac{\delta}{6}. \tag{A.28}$$

Since $H_T \succeq \mu I$ on $\mathcal{N}$ by Assumption (A2), for n_T large the bound in (A.28) implies $\lambda_{\min}(\hat{H}_T(\theta)) \ge \frac{1}{2}\mu$ and hence $\|\hat{H}_T(\theta)^{-1}\| \le 2/\mu$ uniformly on $\mathcal{N}$.

(E_W) Empirical Wasserstein deviation. Let W_c be the Wasserstein metric induced by c in Assumption (A4). Under moment conditions, the empirical measure satisfies (nonasymptotically)

$$P\left(W_c(P_T, \hat{P}_T) > t\right) \le A_1 \exp\{-A_2 n_T t^\alpha\}$$

for some $\alpha > 0$ depending on the effective dimension and tail index (Fournier and Guillin, 2013; Weed and Bach, 2019). Choosing $t = C_W\sqrt{\log(6/\delta)/n_T}$ yields, for some $C_W > 0$,

$$P\left(W_c(P_T, \hat{P}_T) \le C_W\sqrt{\frac{\log(6/\delta)}{n_T}}\right) \ge 1 - \frac{\delta}{6}. \tag{A.29}$$

(E_{MMD}) Alignment approximation and concentration. With a bounded characteristic kernel $\mathcal{K}$ (Assumption (A5)), the unbiased MMD^2 estimator based on n_T target points and r source subsample points is a (bi-)U-statistic. Hoeffding/McDiarmid inequalities for bounded U-statistics imply (Gretton et al., 2012)

$$\sup_{\pi\in\Pi(c)} \left|\hat{\mathrm{MMD}}^2(\pi) - \mathrm{MMD}^2(P_T, P_S(\pi))\right| \le C'_A\left(\sqrt{\frac{\log(6/\delta)}{n_T}} + \frac{1}{\sqrt{r}}\right)$$

with probability at least $1-\delta/6$. For Random Fourier Features (RFF), if D features are used, uniform kernel approximation error is $O_P(D^{-1/2})$ (Rahimi and Recht, 2007; Sutherland and Schneider, 2015). Under Assumption (A5) we take D growing at most polylogarithmically so that $D^{-1/2}$ is dominated by the sampling term, yielding for some $C_A > 0$,

$$P\left(\sup_{\pi\in\Pi(c)} \left|\hat{\mathrm{Align}}_{\mathrm{RFF}}(\pi) - \mathrm{MMD}^2(P_T, P_S(\pi))\right| \le C_A\left(\sqrt{\frac{\log(6/\delta)}{n_T}} + \frac{1}{\sqrt{r}}\right)\right) \ge 1 - \frac{\delta}{6}. \tag{A.30}$$

(E_Λ) Validation over a finite grid. Algorithm 2 selects $\hat{\lambda}$ from a finite grid Λ using an independent validation split (Assumption (A7)). Applying (A.27)-(A.30) for each fixed $\lambda \in \Lambda$ and taking a union bound over $|\Lambda|$ values yields the same bounds with $\log(6/\delta)$ replaced by $\log(6|\Lambda|/\delta)$; we absorb $\log|\Lambda|$ into constants since $|\Lambda|$ is fixed. Thus, with probability at least $1 - \delta/6$, (A.27)-(A.30) hold simultaneously for all $\lambda \in \Lambda$.

Define $\mathcal{E}_\delta = (E_g) \cap (E_H) \cap (E_W) \cap (E_{\mathrm{MMD}}) \cap (E_\Lambda)$. By the displays above and a final union bound, $P(\mathcal{E}_\delta) \ge 1 - \delta$. By the mean-value expansion (S1) there exists $\tilde{\theta}$ on the segment $[\hat{\theta}_f, \theta_T^\star]$ such that

$$g_T(\hat{\theta}_f) = g_T(\theta_T^\star) + \hat{H}_T(\tilde{\theta})(\hat{\theta}_f - \theta_T^\star).$$

Insert this into the Newton map ((A.2) in S1) to get

$$\begin{aligned}\hat{\theta} - \theta_T^\star &= (\hat{\theta}_f - \theta_T^\star) - \hat{H}_T(\hat{\theta}_f)^{-1} g_T(\hat{\theta}_f) \\ &= -\hat{H}_T(\hat{\theta}_f)^{-1} g_T(\theta_T^\star) + \hat{H}_T(\hat{\theta}_f)^{-1}\big(\hat{H}_T(\hat{\theta}_f) - \hat{H}_T(\tilde{\theta})\big)(\hat{\theta}_f - \theta_T^\star). \end{aligned} \tag{A.31}$$

On $\mathcal{E}_\delta$, $\|\hat{H}_T(\hat{\theta}_f)^{-1}\| \le 2/\mu$ by (E_H). By a mean-value bound and (E_H), there is $C'_H > 0$ such that $\|\hat{H}_T(\hat{\theta}_f) - \hat{H}_T(\tilde{\theta})\| \le C'_H \|\hat{\theta}_f - \theta_T^\star\|$. Hence, on $\mathcal{E}_\delta$, we have

$$\|\hat{\theta} - \theta_T^\star\| \le \frac{2}{\mu}\|g_T(\theta_T^\star)\| + \frac{2C'_H}{\mu}\|\hat{\theta}_f - \theta_T^\star\|^2. \tag{A.32}$$

Because R_T is μ-strongly convex on $\mathcal{N}$ and the fusion objective inherits bounded weights (Assumption (A1)), a standard M-estimation argument (Andrews, 1992) yields, on $\mathcal{E}_\delta$,

$$\|\hat{\theta}_f - \theta_T^\star\| \le \frac{2}{\mu}\Big(\underbrace{\|g_T(\theta_T^\star)\|}_{\text{target sampling}} + \underbrace{C_\rho \rho}_{\text{DRO radius}} + \underbrace{C_A\left(\sqrt{\frac{\log(6/\delta)}{n_T}} + \frac{1}{\sqrt{r}}\right)}_{\text{alignment}} + \underbrace{C_\pi \varepsilon_{K,T}}_{\text{optimization}} \Big), \tag{A.33}$$

where (i) the ρ term comes from replacing P_T by $\hat{P}_T$ in the robustified criterion (Theorem 4.2 together with (E_W)), (ii) the alignment term comes from (A.30), and (iii) $\varepsilon_{K,T}$ propagates through the Lipschitz dependence of the fusion objective and Newton map on π (Assumption (A6)). Any dependence on $\hat{\lambda}$ is harmless because Λ is finite and independent (Assumption (A7)).

Combine (A.32), (A.27), and (A.33). The quadratic term $\|\hat{\theta}_f - \theta_T^\star\|^2$ is the square of the right-hand side of (A.33); for n_T large it is dominated by the leading linear terms and can be absorbed into constants. Thus, on $\mathcal{E}_\delta$,

$$\|\hat{\theta} - \theta_T^\star\| \le C_1 \left(\sqrt{\frac{\log(1/\delta)}{n_T}} + \rho + \frac{1}{\sqrt{r}} + \varepsilon_{K,T}\right), \tag{A.34}$$

for a constant $C_1 > 0$ depending only on μ, the local envelopes in Assumptions (A2)-(A3), and the constants in (A.27)-(A.30). By the smoothness side of (A.26) and $\nabla_\theta R_T(\theta_T^\star) = 0$, we have

$$R_T(\hat{\theta}) - R_T(\theta_T^\star) \le \frac{L_\theta}{2}\|\hat{\theta} - \theta_T^\star\|^2.$$

Insert (A.34) to obtain, on $\mathcal{E}_\delta$,

$$R_T(\hat{\theta}) - R_T(\theta_T^\star) \le C_2 \left(\frac{\log(1/\delta)}{n_T} + \rho^2 + \frac{1}{r} + \varepsilon_{K,T}^2\right),$$

for a constant $C_2 > 0$ depending only on L_θ, μ, and the Lipschitz/moment envelopes of Assumptions (A2)-(A3). Since $P(\mathcal{E}_\delta) \ge 1 - \delta$, the asserted high-probability bounds follow, and thus $\|\hat{\theta} - \theta_T^\star\| = O_p(n_T^{-1/2} + \rho + r^{-1/2} + \varepsilon_{K,T})$ and $R_T(\hat{\theta}) - R_T(\theta_T^\star) = O_p(n_T^{-1} + \rho^2 + r^{-1} + \varepsilon_{K,T}^2)$. □

Appendix B. Justification of Remark 1

We provide a formal justification for the claim in Remark 1 that the weighting parameter α used in the fusion estimator has no first-order effect on the refined estimator $\hat{\theta}_{\text{final}}(\pi;\alpha)$. Let the population target risk be $R_T(\theta)$ with $\theta_T^\star = \arg\min_{\theta\in\Theta} R_T(\theta)$, $\nabla R_T(\theta_T^\star) = 0$, and $H_T = \nabla^2 R_T(\theta_T^\star)$, where H_T is positive definite. Let the empirical target risk be $\hat{R}_T(\theta)$, with score and Hessian $g_T(\theta) = \nabla\hat{R}_T(\theta)$ and $\hat{H}_T(\theta) = \nabla^2\hat{R}_T(\theta)$. For a fixed $\alpha \in [0,1]$, define the fusion estimator

$$\hat{\theta}_{\text{fusion}}(\pi;\alpha) = \arg\min_{\theta\in\Theta}\Big\{\alpha\tilde{R}_S(\theta;\pi) + (1-\alpha)\hat{R}_T(\theta)\Big\},$$

and the refined estimator via a one-step Newton update toward the target optimum,

$$\hat{\theta}_{\text{final}}(\pi;\alpha) = \hat{\theta}_{\text{fusion}}(\pi;\alpha) - \hat{H}_T\left(\hat{\theta}_{\text{fusion}}(\pi;\alpha)\right)^{-1} g_T\left(\hat{\theta}_{\text{fusion}}(\pi;\alpha)\right). \tag{B.1}$$

We impose the following standard conditions.

(A1) Local strong convexity. H_T is positive definite with $\lambda_{\min}(H_T) \geq c_0 > 0$.

(A2) Smoothness. $\nabla^2 R_T(\theta)$ is Lipschitz in a neighborhood $\mathcal{N}$ of $\theta_T^\star$.

(A3) Empirical process control. Uniformly for $\theta \in \mathcal{N}$, $\|g_T(\theta) - \nabla R_T(\theta)\| = O_p(n_T^{-1/2})$ and $\|\hat{H}_T(\theta) - \nabla^2 R_T(\theta)\| = o_p(1)$.

(A4) Quality of the starting point. For each fixed $\alpha \in [0,1]$, $\hat{\theta}_{\text{fusion}}(\pi;\alpha) - \theta_T^\star = O_p(n_T^{-1/2})$ and $\hat{\theta}_{\text{fusion}}(\pi;\alpha) \in \mathcal{N}$ w.p. $\to 1$.

Define $\hat{\theta}_0(\alpha) = \hat{\theta}_{\text{fusion}}(\pi;\alpha)$ and $\Delta_0(\alpha) = \hat{\theta}_0(\alpha) - \theta_T^\star$. By a Taylor expansion of $g_T(\cdot)$ around $\theta_T^\star$, there exists $\tilde{\theta}(\alpha)$ on the line segment between $\theta_T^\star$ and $\hat{\theta}_0(\alpha)$ such that

$$g_T(\hat{\theta}_0(\alpha)) = g_T(\theta_T^\star) + \hat{H}_T(\tilde{\theta}(\alpha))\Delta_0(\alpha). \tag{B.2}$$

Substituting (B.2) into (B.1) yields

$$\begin{aligned}\hat{\theta}_{\text{final}}(\pi;\alpha) - \theta_T^\star &= \Delta_0(\alpha) - \hat{H}_T(\hat{\theta}_0(\alpha))^{-1}\Big[g_T(\theta_T^\star) + \hat{H}_T(\tilde{\theta}(\alpha))\Delta_0(\alpha)\Big] \\ &= -\hat{H}_T(\hat{\theta}_0(\alpha))^{-1} g_T(\theta_T^\star) + \Big[I - \hat{H}_T(\hat{\theta}_0(\alpha))^{-1}\hat{H}_T(\tilde{\theta}(\alpha))\Big]\Delta_0(\alpha). \end{aligned}\tag{B.3}$$

By (A3) and $\nabla R_T(\theta_T^\star) = 0$, $g_T(\theta_T^\star) = g_T(\theta_T^\star) - \nabla R_T(\theta_T^\star) = O_p(n_T^{-1/2})$. Moreover, by (A1)–(A3), $\hat{H}_T(\hat{\theta}_0(\alpha))^{-1} = H_T^{-1} + o_p(1)$, so the first term in (B.3) satisfies

$$-\hat{H}_T(\hat{\theta}_0(\alpha))^{-1} g_T(\theta_T^\star) = -H_T^{-1} g_T(\theta_T^\star) + o_p(n_T^{-1/2}). \tag{B.4}$$

For the second term, Lipschitz continuity in (A2) implies $\|\hat{H}_T(\hat{\theta}_0(\alpha)) - \hat{H}_T(\tilde{\theta}(\alpha))\| \leq C\|\Delta_0(\alpha)\| + o_p(1)$, and hence

$$\Big\|\Big[I - \hat{H}_T(\hat{\theta}_0(\alpha))^{-1}\hat{H}_T(\tilde{\theta}(\alpha))\Big]\Delta_0(\alpha)\Big\| \leq C\|\Delta_0(\alpha)\|^2 + o_p(1)\|\Delta_0(\alpha)\|.$$

By (A4), $\|\Delta_0(\alpha)\| = O_p(n_T^{-1/2})$, so the entire remainder term is $o_p(n_T^{-1/2})$.

Combining above equations, we obtain $\hat{\theta}_{\text{final}}(\pi;\alpha) - \theta_T^\star = -H_T^{-1} g_T(\theta_T^\star) + o_p(n_T^{-1/2})$, uniformly over fixed $\alpha \in [0,1]$. Consequently, for any fixed $\alpha_1, \alpha_2 \in [0,1]$,

$$\hat{\theta}_{\text{final}}(\pi;\alpha_1) - \hat{\theta}_{\text{final}}(\pi;\alpha_2) = o_p(n_T^{-1/2}),$$

which establishes that the choice of α has no first-order effect on the asymptotic distribution of the refined estimator.

□

Appendix C. Practical Tuning Checklist

We summarize a minimal, practitioner-oriented workflow for tuning the proposed method. The steps below are designed to provide reliable default behavior while preserving the theoretical guarantees established in the main text.

Step 0 (Preprocessing). Standardize covariates using target-domain statistics (mean zero and unit variance). When the ambient dimension p is large, apply a dimension-reduction step (e.g., PCA) and define an *effective dimension* d_{eff} as the number of components explaining a fixed proportion (e.g., 95%) of the target variance.

Step 1 (Robustness radius ρ). Calibrate ρ using the target geometry as in Remark 2. Specifically, compute the median nearest-neighbor distance mNN in the target sample and set $\rho = \text{mNN} n_T^{-1/d_{\text{eff}}}$. This choice shrinks with n_T and adapts to the local scale of P_T, yielding a default that is both theoretically justified and stable in practice.

Step 2 (Trade-off parameter λ). Select λ via target-only validation as implemented in Algorithm 2. In practice, a small logarithmic grid $\Lambda = \{0, 10^{-3}, 10^{-2}, 10^{-1}, 1, 10\}$ is typically sufficient. Choose $\hat{\lambda} = \arg\min_{\lambda\in\Lambda} J_{\text{val}}(\lambda)$, where $J_{\text{val}}(\lambda)$ denotes the held-out target validation risk computed after running Algorithm 1 at λ. This procedure directly targets predictive performance on the target domain.

Step 3 (Subsample size r and feature map dimension D). Choose the subsample size r based on computational constraints. Increasing r reduces the subsampling variability term in the error bounds (cf. Section 4) but increases the per-iteration cost through the alignment evaluation. For the alignment term, approximate the kernel using random Fourier features (RFF) with D in the range 10^3-2×10^3 as a default. With RFF, each objective evaluation scales as $O\big((n_T + r)D\big)$ rather than quadratically in the sample size.

Step 4 (PSO hyperparameters and stopping). Default choices $\omega \in [0.6, 0.9]$, $\phi_1 = \phi_2 \in [1,2]$, swarm size $K \in [20,50]$, and iteration budget $T \in [100,200]$ provide a reliable balance between exploration and convergence. Employ a stall-based stopping criterion with tolerance $\tau > 0$ and stall limit S (Algorithm 1) to terminate early once the global-best objective ceases to improve.

Step 5 (Refinement stability). When computing the one-step target refinement, apply ridge regularization to the empirical Hessian, or use a conjugate-gradient solver in high-dimensional settings, if the Hessian is ill-conditioned.

Diagnostics. Report (i) the target validation risk at $\hat{\lambda}$, (ii) the alignment value $\widehat{\text{Align}}(\hat{\pi}_{\hat{\lambda}})$, and (iii) the robust risk $\hat{R}^{(\rho)}_{\text{DRO}}(\hat{\pi}_{\hat{\lambda}})$ to verify that the selected solution lies on a meaningful robustness-alignment trade-off.

Appendix D. Additional Simulation Results

Table 8: DGP 1 (Covariate shift) with $r = 5000$

(n_S, n_T)	Method	$\Delta = 0.0$		$\Delta = 0.2$		$\Delta = 0.4$	
		MSE/Robust	MMD	MSE/Robust	MMD	MSE/Robust	MMD
(50K,2K)	Uniform	1.0037 (0.0312)/1.1048 (0.0397)	0.0517 (0.0049)	1.1258 (0.0436)/1.2483 (0.0527)	0.0639 (0.0067)	1.2405 (0.0521)/1.3587 (0.0665)	0.0724 (0.0081)
	Target-only	0.9912 (0.0298)/1.0985 (0.0374)	–	1.0921 (0.0412)/1.2186 (0.0505)	–	1.1512 (0.0614)/1.3026 (0.0603)	–
	Leverage	0.9853 (0.0287)/1.0924 (0.0361)	0.0493 (0.0045)	1.0843 (0.0406)/1.2127 (0.0496)	0.0604 (0.0061)	1.1083 (0.0432)/1.2814 (0.0516)	0.0653 (0.0070)
	MMD-only	0.9765 (0.0279)/1.0876 (0.0354)	0.0475 (0.0044)	1.0719 (0.0395)/1.2012 (0.0483)	0.0572 (0.0058)	1.0842 (0.0418)/1.2427 (0.0498)	0.0571 (0.0061)
	IPW-only	0.9794 (0.0281)/1.0891 (0.0359)	0.0488 (0.0043)	1.0784 (0.0399)/1.2065 (0.0489)	0.0591 (0.0059)	1.1026 (0.0485)/1.2211 (0.0574)	0.0632 (0.0064)
	RE-only	0.9752 (0.0276)/1.0859 (0.0351)	0.0479 (0.0042)	1.0642 (0.0387)/1.1948 (0.0475)	0.0578 (0.0056)	1.0648 (0.0409)/1.1965 (0.0489)	0.0593 (0.0052)
	Proposed	**0.9627** (0.0268)/**1.0713** (0.0339)	**0.0451** (0.0040)	**1.0326** (0.0365)/**1.1627** (0.0452)	**0.0536** (0.0052)	**0.9823** (0.0325)/**1.1017** (0.0412)	**0.0512** (0.0041)
(50K,5K)	Uniform	0.9918 (0.0289)/1.0974 (0.0368)	0.0509 (0.0048)	1.1027 (0.0379)/1.2241 (0.0465)	0.0631 (0.0061)	1.2142 (0.0486)/1.3328 (0.0571)	0.0714 (0.0077)
	Target-only	0.9735 (0.0256)/1.0854 (0.0339)	–	1.0548 (0.0306)/1.1764 (0.0394)	–	1.1168 (0.0389)/1.2605 (0.0476)	–
	Leverage	0.9789 (0.0268)/1.0875 (0.0345)	0.0488 (0.0044)	1.0729 (0.0341)/1.1997 (0.0428)	0.0602 (0.0058)	1.1546 (0.0441)/1.2794 (0.0526)	0.0689 (0.0073)
	MMD-only	0.9721 (0.0261)/1.0824 (0.0339)	0.0479 (0.0043)	1.0617 (0.0332)/1.1896 (0.0421)	0.0591 (0.0057)	1.1419 (0.0435)/1.2694 (0.0520)	0.0676 (0.0070)
	IPW-only	0.9816 (0.0270)/1.0893 (0.0347)	0.0492 (0.0044)	1.0763 (0.0345)/1.2031 (0.0431)	0.0606 (0.0058)	1.1592 (0.0446)/1.2839 (0.0531)	0.0693 (0.0073)
	RE-only	0.9687 (0.0257)/1.0798 (0.0336)	0.0482 (0.0043)	1.0568 (0.0330)/1.1847 (0.0418)	0.0588 (0.0056)	1.1286 (0.0426)/1.2618 (0.0513)	0.0669 (0.0069)
	Proposed	**0.9542** (0.0241)/**1.0649** (0.0324)	**0.0443** (0.0039)	**1.0197** (0.0289)/**1.1512** (0.0377)	**0.0528** (0.0051)	**0.9749** (0.0316)/**1.0946** (0.0403)	**0.0507** (0.0040)
(200K,2K)	Uniform	0.9996 (0.0307)/1.1028 (0.0389)	0.0506 (0.0048)	1.1176 (0.0409)/1.2407 (0.0496)	0.0626 (0.0060)	1.2331 (0.0518)/1.3529 (0.0604)	0.0711 (0.0076)
	Target-only	0.9912 (0.0298)/1.0985 (0.0374)	–	1.0921 (0.0412)/1.2186 (0.0505)	–	1.1512 (0.0614)/1.3026 (0.0603)	–
	Leverage	0.9897 (0.0291)/1.0954 (0.0372)	0.0490 (0.0046)	1.0831 (0.0385)/1.2104 (0.0471)	0.0601 (0.0058)	1.1639 (0.0476)/1.2917 (0.0563)	0.0686 (0.0072)
	MMD-only	0.9832 (0.0286)/1.0901 (0.0366)	0.0484 (0.0045)	1.0715 (0.0376)/1.1994 (0.0463)	0.0590 (0.0057)	1.1492 (0.0461)/1.2794 (0.0548)	0.0672 (0.0070)
	IPW-only	0.9948 (0.0295)/1.1007 (0.0376)	0.0496 (0.0047)	1.0876 (0.0388)/1.2147 (0.0474)	0.0607 (0.0059)	1.1707 (0.0479)/1.2971 (0.0565)	0.0690 (0.0073)
	RE-only	0.9786 (0.0282)/1.0881 (0.0364)	0.0487 (0.0045)	1.0682 (0.0371)/1.1958 (0.0460)	0.0586 (0.0057)	1.1396 (0.0456)/1.2708 (0.0544)	0.0666 (0.0069)
	Proposed	**0.9598** (0.0261)/**1.0687** (0.0347)	**0.0449** (0.0040)	**1.0116** (0.0284)/**1.1447** (0.0373)	**0.0524** (0.0050)	**0.9693** (0.0318)/**1.0895** (0.0407)	**0.0505** (0.0041)
(200K,5K)	Uniform	0.9867 (0.0284)/1.0916 (0.0365)	0.0501 (0.0047)	1.1023 (0.0374)/1.2248 (0.0462)	0.0622 (0.0059)	1.2146 (0.0481)/1.3349 (0.0566)	0.0706 (0.0075)
	Target-only	0.9735 (0.0256)/1.0854 (0.0339)	–	1.0548 (0.0306)/1.1764 (0.0394)	–	1.1168 (0.0389)/1.2605 (0.0476)	–
	Leverage	0.9759 (0.0261)/1.0831 (0.0341)	0.0483 (0.0044)	1.0617 (0.0329)/1.1897 (0.0417)	0.0597 (0.0057)	1.1416 (0.0437)/1.2681 (0.0522)	0.0683 (0.0071)
	MMD-only	0.9688 (0.0256)/1.0779 (0.0336)	0.0476 (0.0043)	1.0492 (0.0319)/1.1765 (0.0408)	0.0586 (0.0056)	1.1279 (0.0426)/1.2551 (0.0513)	0.0670 (0.0069)
	IPW-only	0.9806 (0.0265)/1.0864 (0.0344)	0.0487 (0.0045)	1.0651 (0.0332)/1.1931 (0.0421)	0.0601 (0.0058)	1.1463 (0.0440)/1.2724 (0.0525)	0.0688 (0.0072)
	RE-only	0.9634 (0.0251)/1.0751 (0.0333)	0.0479 (0.0043)	1.0446 (0.0316)/1.1732 (0.0406)	0.0583 (0.0056)	1.1214 (0.0418)/1.2512 (0.0506)	0.0665 (0.0069)
	Proposed	**0.9496** (0.0237)/**1.0604** (0.0321)	**0.0439** (0.0039)	**1.0051** (0.0279)/**1.1392** (0.0368)	**0.0520** (0.0049)	**0.9621** (0.0313)/**1.0837** (0.0401)	**0.0501** (0.0041)

Table 9: DGP 1 (Covariate shift) with $r = 10000$

(n_S, n_T)	Method	$\Delta = 0.0$		$\Delta = 0.2$		$\Delta = 0.4$	
		MSE/Robust	MMD	MSE/Robust	MMD	MSE/Robust	MMD
(50K,2K)	Uniform	0.9926 (0.0304)/1.0972 (0.0386)	0.0503 (0.0046)	1.1087 (0.0376)/1.2314 (0.0462)	0.0618 (0.0058)	1.2193 (0.0479)/1.3381 (0.0565)	0.0702 (0.0073)
	Target-only	0.9912 (0.0298)/1.0985 (0.0374)	–	1.0921 (0.0412)/1.2186 (0.0505)	–	1.1512 (0.0614)/1.3026 (0.0603)	–
	Leverage	0.9786 (0.0274)/1.0863 (0.0351)	0.0481 (0.0043)	1.0614 (0.0327)/1.1898 (0.0416)	0.0594 (0.0056)	1.1425 (0.0431)/1.2701 (0.0518)	0.0678 (0.0070)
	MMD-only	0.9714 (0.0268)/1.0815 (0.0346)	0.0472 (0.0042)	1.0498 (0.0319)/1.1779 (0.0408)	0.0584 (0.0055)	1.1291 (0.0420)/1.2583 (0.0509)	0.0668 (0.0069)
	IPW-only	0.9831 (0.0277)/1.0904 (0.0354)	0.0486 (0.0044)	1.0668 (0.0331)/1.1957 (0.0419)	0.0600 (0.0057)	1.1497 (0.0437)/1.2771 (0.0524)	0.0688 (0.0071)
	RE-only	0.9663 (0.0261)/1.0786 (0.0342)	0.0477 (0.0043)	1.0449 (0.0313)/1.1736 (0.0403)	0.0581 (0.0055)	1.1183 (0.0413)/1.2489 (0.0501)	0.0662 (0.0068)
	Proposed	**0.9547** (0.0242)/**1.0638** (0.0325)	**0.0437** (0.0038)	**1.0146** (0.0283)/**1.1469** (0.0371)	**0.0516** (0.0048)	**0.9728** (0.0309)/**1.0927** (0.0397)	**0.0496** (0.0040)
(50K,5K)	Uniform	0.9791 (0.0279)/1.0847 (0.0356)	0.0497 (0.0045)	1.0913 (0.0348)/1.2149 (0.0437)	0.0612 (0.0057)	1.2047 (0.0456)/1.3256 (0.0542)	0.0696 (0.0072)
	Target-only	0.9735 (0.0256)/1.0854 (0.0339)	–	1.0548 (0.0306)/1.1764 (0.0394)	–	1.1168 (0.0389)/1.2605 (0.0476)	–
	Leverage	0.9661 (0.0254)/1.0753 (0.0338)	0.0476 (0.0043)	1.0467 (0.0316)/1.1756 (0.0405)	0.0587 (0.0056)	1.1257 (0.0428)/1.2536 (0.0514)	0.0672 (0.0069)
	MMD-only	0.9597 (0.0249)/1.0706 (0.0333)	0.0468 (0.0042)	1.0349 (0.0307)/1.1638 (0.0397)	0.0580 (0.0055)	1.1128 (0.0418)/1.2423 (0.0506)	0.0661 (0.0068)
	IPW-only	0.9714 (0.0257)/1.0794 (0.0341)	0.0481 (0.0043)	1.0517 (0.0312)/1.1828 (0.0401)	0.0594 (0.0056)	1.1359 (0.0432)/1.2639 (0.0520)	0.0678 (0.0070)
	RE-only	0.9546 (0.0244)/1.0679 (0.0329)	0.0473 (0.0042)	1.0301 (0.0300)/1.1603 (0.0391)	0.0578 (0.0055)	1.1061 (0.0412)/1.2376 (0.0501)	0.0658 (0.0068)
	Proposed	**0.9449** (0.0232)/**1.0563** (0.0318)	**0.0430** (0.0038)	**1.0014** (0.0276)/**1.1357** (0.0365)	**0.0511** (0.0047)	**0.9613** (0.0298)/**1.0831** (0.0387)	**0.0490** (0.0039)
(200K,2K)	Uniform	0.9884 (0.0298)/1.0941 (0.0379)	0.0499 (0.0045)	1.1021 (0.0369)/1.2257 (0.0458)	0.0608 (0.0056)	1.2143 (0.0471)/1.3358 (0.0558)	0.0693 (0.0071)
	Target-only	0.9912 (0.0298)/1.0985 (0.0374)	–	1.0921 (0.0412)/1.2186 (0.0505)	–	1.1512 (0.0614)/1.3026 (0.0603)	–
	Leverage	0.9723 (0.0269)/1.0817 (0.0339)	0.0472 (0.0043)	1.0429 (0.0307)/1.1731 (0.0396)	0.0584 (0.0055)	1.1218 (0.0419)/1.2508 (0.0507)	0.0669 (0.0069)
	MMD-only	0.9659 (0.0263)/1.0769 (0.0334)	0.0465 (0.0042)	1.0317 (0.0299)/1.1617 (0.0389)	0.0577 (0.0054)	1.1084 (0.0411)/1.2398 (0.0500)	0.0658 (0.0068)
	IPW-only	0.9776 (0.0273)/1.0861 (0.0342)	0.0479 (0.0043)	1.0495 (0.0311)/1.1819 (0.0400)	0.0592 (0.0056)	1.1314 (0.0423)/1.2611 (0.0511)	0.0676 (0.0070)
	RE-only	0.9607 (0.0259)/1.0736 (0.0328)	0.0470 (0.0042)	1.0261 (0.0294)/1.1569 (0.0385)	0.0573 (0.0054)	1.1002 (0.0406)/1.2325 (0.0497)	0.0654 (0.0068)
	Proposed	**0.9498** (0.0239)/**1.0617** (0.0319)	**0.0427** (0.0038)	**0.9976** (0.0268)/**1.1328** (0.0357)	**0.0508** (0.0047)	**0.9586** (0.0292)/**1.0812** (0.0381)	**0.0487** (0.0039)
(200K,5K)	Uniform	0.9736 (0.0274)/1.0827 (0.0353)	0.0494 (0.0044)	1.0879 (0.0336)/1.2124 (0.0425)	0.0606 (0.0055)	1.1997 (0.0447)/1.3219 (0.0533)	0.0689 (0.0068)
	Target-only	0.9735 (0.0256)/1.0854 (0.0339)	–	1.0548 (0.0306)/1.1764 (0.0394)	–	1.1168 (0.0389)/1.2605 (0.0476)	–
	Leverage	0.9617 (0.0247)/1.0728 (0.0332)	0.0470 (0.0042)	1.0384 (0.0301)/1.1682 (0.0389)	0.0581 (0.0054)	1.1159 (0.0412)/1.2459 (0.0498)	0.0666 (0.0068)
	MMD-only	0.9554 (0.0242)/1.0678 (0.0327)	0.0463 (0.0041)	1.0267 (0.0290)/1.1564 (0.0379)	0.0573 (0.0054)	1.1027 (0.0407)/1.2341 (0.0496)	0.0652 (0.0067)
	IPW-only	0.9673 (0.0249)/1.0769 (0.0335)	0.0475 (0.0042)	1.0441 (0.0305)/1.1746 (0.0393)	0.0589 (0.0055)	1.1268 (0.0417)/1.2568 (0.0505)	0.0669 (0.0069)
	RE-only	0.9506 (0.0237)/1.0657 (0.0324)	0.0468 (0.0041)	1.0224 (0.0288)/1.1532 (0.0378)	0.0571 (0.0054)	1.0952 (0.0402)/1.2281 (0.0492)	0.0649 (0.0067)
	Proposed	**0.9426** (0.0228)/**1.0541** (0.0315)	**0.0424** (0.0037)	**0.9942** (0.0261)/**1.1294** (0.0350)	**0.0505** (0.0046)	**0.9547** (0.0286)/**1.0786** (0.0376)	**0.0484** (0.0038)

Table 10: DGP 2 (Label shift) with $r = 5000$

(n_S, n_T)	Method	Mild	Moderate	Severe
		Error/Robust	Error/Robust	Error/Robust
(50K,2K)	Uniform	18.7261 (0.8972)/20.4926 (0.9793)	22.9836 (1.0457)/25.5164 (1.1895)	26.7327 (1.2291)/29.3518 (1.3548)
	Target-only	18.1935 (0.8597)/19.9362 (0.9379)	21.4631 (0.9971)/23.9548 (1.1263)	24.9128 (1.1518)/27.5413 (1.2796)
	IWERM	17.2619 (0.8123)/19.5861 (0.9317)	20.7987 (0.9385)/23.4658 (1.0824)	24.3985 (1.1076)/27.4316 (1.2425)
	MMD-only	16.7426 (0.7748)/19.2783 (0.8894)	20.2263 (0.9061)/23.1597 (1.0372)	23.6135 (1.0697)/26.8926 (1.2041)
	IPW-only	17.5742 (0.8316)/19.8231 (0.9462)	21.1167 (0.9584)/23.6712 (1.1005)	24.6134 (1.1229)/27.4395 (1.2602)
	RE-only	17.0385 (0.8017)/19.4536 (0.9165)	20.5849 (0.9274)/23.3637 (1.0716)	24.0754 (1.0924)/27.1562 (1.2284)
	Proposed	**15.8437** (0.7463)/**18.4069** (0.8326)	**19.2836** (0.8819)/**21.9794** (0.9734)	**22.2794** (1.0061)/**25.4472** (1.1338)
(50K,5K)	Uniform	17.9682 (0.8419)/19.6726 (0.9228)	22.1246 (1.0192)/24.6719 (1.1608)	25.8641 (1.1962)/28.4762 (1.3219)
	Target-only	17.4829 (0.8056)/19.2284 (0.8824)	20.0286 (0.9107)/22.4873 (1.0463)	23.2259 (1.0239)/25.8921 (1.1512)
	IWERM	16.1234 (0.6931)/18.5487 (0.8206)	19.3179 (0.8234)/22.0456 (0.9772)	22.2264 (0.9328)/25.3471 (1.0681)
	MMD-only	15.5473 (0.6162)/18.2196 (0.6937)	16.9472 (0.6974)/21.7214 (0.5036)	18.8463 (0.7962)/22.9961 (0.6119)
	IPW-only	16.6237 (0.7249)/18.9594 (0.7803)	18.9726 (0.8735)/21.3827 (0.9586)	21.6513 (0.9528)/24.4927 (1.0386)
	RE-only	16.1785 (0.7038)/18.7561 (0.7684)	18.5893 (0.8426)/21.1952 (0.9326)	21.3186 (0.9227)/24.2359 (1.0281)
	Proposed	**14.5362** (0.5891)/**17.2083** (0.6728)	**15.9345** (0.6382)/**19.6084** (0.7816)	**17.8453** (0.5241)/**22.2479** (0.8857)
(200K,2K)	Uniform	18.9826 (0.9227)/20.7414 (1.0069)	23.2947 (1.0859)/25.8316 (1.2326)	27.1348 (1.2642)/29.7526 (1.3928)
	Target-only	18.1935 (0.8597)/19.9362 (0.9379)	21.4631 (0.9971)/23.9548 (1.1263)	24.9128 (1.1518)/27.5413 (1.2796)
	IWERM	17.1426 (0.8069)/19.4728 (0.9286)	20.6653 (0.9275)/23.3207 (1.0706)	24.2517 (1.1013)/27.2769 (1.2371)
	MMD-only	16.6281 (0.7701)/19.1742 (0.8813)	20.0746 (0.8874)/23.0138 (1.0169)	23.4127 (1.0605)/26.7029 (1.1957)
	IPW-only	17.4462 (0.8238)/19.6994 (0.9441)	20.9938 (0.9485)/23.5551 (1.0897)	24.4824 (1.1234)/27.3041 (1.2586)
	RE-only	16.9187 (0.7959)/19.3401 (0.9082)	20.4675 (0.9179)/23.2416 (1.0634)	23.9528 (1.0843)/27.0236 (1.2197)
	Proposed	**15.7894** (0.7394)/**18.3567** (0.8262)	**19.1518** (0.8742)/**21.8496** (0.9661)	**22.1487** (0.9994)/**25.3174** (1.1276)
(200K,5K)	Uniform	17.8462 (0.8385)/19.5416 (0.9207)	22.0129 (1.0136)/24.5537 (1.1547)	25.7471 (1.1884)/28.3581 (1.3128)
	Target-only	17.4829 (0.8056)/19.2284 (0.8824)	20.0286 (0.9107)/22.4873 (1.0463)	23.2259 (1.0239)/25.8921 (1.1512)
	IWERM	15.9871 (0.6852)/18.4173 (0.8124)	19.1946 (0.8156)/21.9284 (0.9691)	22.0864 (0.9241)/25.2126 (1.0602)
	MMD-only	15.4218 (0.6087)/18.0897 (0.6854)	16.8247 (0.6892)/21.6274 (0.4981)	18.7135 (0.7841)/22.8316 (0.6074)
	IPW-only	16.4796 (0.7141)/18.8207 (0.7696)	18.8462 (0.8628)/21.2604 (0.9486)	21.5286 (0.9421)/24.3748 (1.0279)
	RE-only	16.0483 (0.6935)/18.6254 (0.7601)	18.4729 (0.8326)/21.0796 (0.9234)	21.1884 (0.9127)/24.1128 (1.0172)
	Proposed	**14.3961** (0.5827)/**17.0764** (0.6663)	**15.8127** (0.6325)/**19.4872** (0.7735)	**17.6425** (0.5178)/**22.1206** (0.8731)

Table 11: DGP 2 (Label shift) with $r = 10000$

(n_S, n_T)	Method	Mild	Moderate	Severe
		Error/Robust	Error/Robust	Error/Robust
(50K,2K)	Uniform	18.3129 (0.8724)/20.0217 (0.9541)	22.5614 (1.0395)/25.0976 (1.1826)	26.2947 (1.2158)/28.9142 (1.3417)
	Target-only	17.7936 (0.8358)/19.5213 (0.9137)	20.9846 (0.9907)/23.4617 (1.1229)	24.4137 (1.1451)/27.0316 (1.2734)
	IWERM	16.8947 (0.7896)/19.2271 (0.9142)	20.3468 (0.9314)/23.0061 (1.0762)	23.9271 (1.1034)/26.9584 (1.2391)
	MMD-only	16.3826 (0.7514)/18.9341 (0.8673)	19.7863 (0.8987)/22.7306 (1.0319)	23.1459 (1.0618)/26.4327 (1.1986)
	IPW-only	17.1907 (0.8073)/19.4605 (0.9298)	20.6672 (0.9491)/23.2281 (1.0928)	24.1817 (1.1164)/27.0173 (1.2549)
	RE-only	16.6729 (0.7786)/19.1017 (0.9006)	20.1463 (0.9195)/22.9264 (1.0657)	23.6528 (1.0879)/26.7371 (1.2246)
	Proposed	**15.5073** (0.7372)/**18.0762** (0.8204)	**18.9437** (0.8627)/**21.6314** (0.9554)	**21.9438** (0.9871)/**25.1217** (1.1156)
(50K,5K)	Uniform	17.6248 (0.8145)/19.3237 (0.8941)	21.7863 (1.0048)/24.3368 (1.1467)	25.5128 (1.1803)/28.1246 (1.3061)
	Target-only	17.2481 (0.7816)/18.9864 (0.8583)	19.8128 (0.8964)/22.2729 (1.0315)	23.0017 (1.0118)/25.6714 (1.1402)
	IWERM	15.9183 (0.6691)/18.3564 (0.7985)	19.1186 (0.8112)/21.8561 (0.9651)	21.9974 (0.9182)/25.1317 (1.0558)
	MMD-only	15.3628 (0.5947)/18.0491 (0.6738)	16.6429 (0.6784)/21.4827 (0.4924)	18.5458 (0.7705)/22.7461 (0.6028)
	IPW-only	16.4127 (0.7061)/18.7619 (0.7596)	18.7562 (0.8514)/21.1758 (0.9375)	21.4248 (0.9321)/24.2796 (1.0186)
	RE-only	15.9876 (0.6864)/18.5637 (0.7534)	18.3762 (0.8221)/20.9884 (0.9142)	21.0461 (0.9032)/23.9753 (1.0096)
	Proposed	**14.1836** (0.5748)/**16.8647** (0.6592)	**15.7214** (0.6216)/**19.3016** (0.7637)	**17.5562** (0.5117)/**21.9617** (0.8615)
(200K,2K)	Uniform	18.7231 (0.9081)/20.4816 (0.9919)	22.9826 (1.0791)/25.5129 (1.2248)	26.7061 (1.2567)/29.3218 (1.3847)
	Target-only	18.3129 (0.8724)/20.0217 (0.9541)	20.9846 (0.9907)/23.4617 (1.1229)	24.4137 (1.1451)/27.0316 (1.2734)
	IWERM	16.9987 (0.7931)/19.3384 (0.9167)	20.4493 (0.9221)/23.1027 (1.0669)	24.0361 (1.0962)/27.0628 (1.2321)
	MMD-only	16.4973 (0.7551)/19.0618 (0.8697)	19.9224 (0.8881)/22.8597 (1.0206)	23.2764 (1.0561)/26.5691 (1.1927)
	IPW-only	17.1018 (0.8106)/19.3867 (0.9342)	20.7771 (0.9436)/23.3338 (1.0854)	24.2064 (1.1119)/27.0374 (1.2478)
	RE-only	16.5871 (0.7824)/19.0156 (0.9043)	20.2568 (0.9142)/22.9981 (1.0587)	23.7646 (1.0798)/26.8536 (1.2168)
	Proposed	**15.3684** (0.7305)/**17.9431** (0.8162)	**18.8163** (0.8527)/**21.4961** (0.9451)	**21.8137** (0.9749)/**24.9985** (1.1046)
(200K,5K)	Uniform	17.4819 (0.8013)/19.1827 (0.8816)	21.6485 (0.9884)/24.2048 (1.1295)	25.3749 (1.1638)/27.9846 (1.2881)
	Target-only	17.2481 (0.7816)/18.9864 (0.8583)	19.8128 (0.8964)/22.2729 (1.0315)	23.0017 (1.0118)/25.6714 (1.1402)
	IWERM	15.7816 (0.6615)/18.2269 (0.7931)	18.9862 (0.8047)/21.7237 (0.9592)	21.8554 (0.9106)/24.9853 (1.0486)
	MMD-only	15.2364 (0.5894)/17.9296 (0.6708)	16.4528 (0.6681)/21.3718 (0.4884)	18.3594 (0.7606)/22.6382 (0.5996)
	IPW-only	16.2719 (0.6987)/18.6316 (0.7519)	18.6424 (0.8412)/21.0571 (0.9283)	21.3276 (0.9196)/24.1879 (1.0067)
	RE-only	15.8537 (0.6802)/18.4368 (0.7471)	18.2681 (0.8129)/20.8836 (0.9052)	20.9448 (0.8935)/23.8786 (0.9997)
	Proposed	**13.9741** (0.5673)/**16.6648** (0.6531)	**15.6049** (0.6161)/**19.1164** (0.7539)	**17.4326** (0.5079)/**21.7793** (0.8521)

Table 12: DGP 3 (Concept shift) with $r = 5000$

(n_S, n_T)	Method	$\theta = 0.0$	$\theta = 0.1$	$\theta = 0.2$
		MSE/Robust	MSE/Robust	MSE/Robust
(50K,2K)	Uniform	1.0127 (0.0326)/1.1128 (0.0398)	1.2175 (0.0487)/1.3706 (0.0579)	1.4012 (0.0734)/1.5537 (0.0816)
	Target-only	0.9985 (0.0304)/1.1067 (0.0386)	1.1624 (0.0451)/1.3205 (0.0538)	1.3025 (0.0627)/1.4623 (0.0705)
	Leverage	0.9876 (0.0298)/1.0973 (0.0372)	1.1438 (0.0439)/1.3032 (0.0521)	1.2847 (0.0609)/1.4448 (0.0687)
	MMD-only	1.0051 (0.0315)/1.1096 (0.0389)	1.2114 (0.0481)/1.3657 (0.0568)	1.3916 (0.0719)/1.5453 (0.0803)
	IPW-only	0.9783 (0.0287)/1.0928 (0.0365)	1.1284 (0.0425)/1.2916 (0.0513)	1.2746 (0.0594)/1.4361 (0.0671)
	RE-only	0.9724 (0.0281)/1.0875 (0.0361)	1.1042 (0.0413)/1.2657 (0.0502)	1.2045 (0.0502)/1.3441 (0.0587)
	Proposed	**0.9618** (0.0276)/**1.0734** (0.0354)	**1.0583** (0.0386)/**1.1956** (0.0463)	**1.1186** (0.0431)/**1.2438** (0.0514)
(50K,5K)	Uniform	0.9974 (0.0291)/1.1014 (0.0367)	1.1948 (0.0459)/1.3476 (0.0547)	1.3759 (0.0628)/1.5301 (0.0717)
	Target-only	0.9841 (0.0268)/1.0957 (0.0346)	1.1461 (0.0412)/1.3024 (0.0497)	1.2872 (0.0537)/1.4491 (0.0625)
	Leverage	0.9806 (0.0276)/1.0903 (0.0354)	1.1731 (0.0446)/1.3284 (0.0532)	1.3421 (0.0593)/1.5003 (0.0679)
	MMD-only	0.9927 (0.0287)/1.0998 (0.0364)	1.1892 (0.0454)/1.3432 (0.0541)	1.3634 (0.0619)/1.5196 (0.0706)
	IPW-only	0.9768 (0.0272)/1.0881 (0.0351)	1.1567 (0.0431)/1.3143 (0.0519)	1.3098 (0.0561)/1.4702 (0.0648)
	RE-only	0.9701 (0.0263)/1.0827 (0.0345)	1.1274 (0.0407)/1.2869 (0.0494)	1.2439 (0.0506)/1.4052 (0.0593)
	Proposed	**0.9569** (0.0248)/**1.0706** (0.0333)	**1.0461** (0.0378)/**1.1837** (0.0459)	**1.1079** (0.0426)/**1.2338** (0.0511)
(200K,2K)	Uniform	1.0069 (0.0307)/1.1097 (0.0383)	1.2026 (0.0468)/1.3561 (0.0557)	1.3818 (0.0626)/1.5376 (0.0715)
	Target-only	0.9985 (0.0304)/1.1067 (0.0386)	1.1624 (0.0451)/1.3205 (0.0538)	1.3025 (0.0627)/1.4623 (0.0705)
	Leverage	0.9921 (0.0292)/1.1031 (0.0369)	1.1816 (0.0456)/1.3369 (0.0545)	1.3506 (0.0602)/1.5082 (0.0689)
	MMD-only	1.0017 (0.0304)/1.1076 (0.0381)	1.1968 (0.0465)/1.3509 (0.0554)	1.3754 (0.0618)/1.5316 (0.0706)
	IPW-only	0.9851 (0.0285)/1.0971 (0.0362)	1.1539 (0.0438)/1.3126 (0.0526)	1.3117 (0.0579)/1.4735 (0.0667)
	RE-only	0.9787 (0.0279)/1.0916 (0.0357)	1.1246 (0.0417)/1.2839 (0.0504)	1.2371 (0.0511)/1.3981 (0.0598)
	Proposed	**0.9676** (0.0266)/**1.0814** (0.0344)	**1.0415** (0.0369)/**1.1798** (0.0452)	**1.1029** (0.0418)/**1.2297** (0.0502)
(200K,5K)	Uniform	0.9918 (0.0289)/1.0976 (0.0368)	1.1842 (0.0452)/1.3381 (0.0541)	1.3602 (0.0613)/1.5173 (0.0701)
	Target-only	0.9841 (0.0268)/1.0957 (0.0346)	1.1461 (0.0412)/1.3024 (0.0497)	1.2872 (0.0537)/1.4491 (0.0625)
	Leverage	0.9784 (0.0259)/1.0892 (0.0341)	1.1708 (0.0441)/1.3259 (0.0530)	1.3391 (0.0589)/1.4978 (0.0676)
	MMD-only	0.9886 (0.0276)/1.0958 (0.0354)	1.1817 (0.0447)/1.3362 (0.0535)	1.3543 (0.0608)/1.5119 (0.0696)
	IPW-only	0.9749 (0.0251)/1.0857 (0.0336)	1.1592 (0.0435)/1.3169 (0.0524)	1.3174 (0.0571)/1.4771 (0.0658)
	RE-only	0.9696 (0.0247)/1.0809 (0.0332)	1.1306 (0.0405)/1.2897 (0.0492)	1.2464 (0.0502)/1.4081 (0.0589)
	Proposed	**0.9538** (0.0236)/**1.0681** (0.0320)	**1.0339** (0.0361)/**1.1724** (0.0446)	**1.0956** (0.0412)/**1.2229** (0.0497)

Table 13: DGP 3 (Concept shift) with $r = 10000$

(n_S, n_T)	Method	$\theta = 0.0$	$\theta = 0.1$	$\theta = 0.2$
		MSE/Robust	MSE/Robust	MSE/Robust
(50K,2K)	Uniform	1.0012 (0.0306)/1.1059 (0.0383)	1.2019 (0.0465)/1.3546 (0.0554)	1.3847 (0.0623)/1.5382 (0.0710)
	Target-only	0.9962 (0.0298)/1.1038 (0.0379)	1.1578 (0.0449)/1.3162 (0.0535)	1.2961 (0.0617)/1.4579 (0.0702)
	Leverage	0.9843 (0.0281)/1.0948 (0.0362)	1.1687 (0.0437)/1.3246 (0.0523)	1.3362 (0.0596)/1.4941 (0.0682)
	MMD-only	0.9951 (0.0294)/1.1031 (0.0376)	1.1942 (0.0461)/1.3481 (0.0549)	1.3728 (0.0615)/1.5286 (0.0703)
	IPW-only	0.9748 (0.0275)/1.0889 (0.0356)	1.1516 (0.0429)/1.3112 (0.0518)	1.3092 (0.0568)/1.4715 (0.0655)
	RE-only	0.9691 (0.0270)/1.0847 (0.0351)	1.1237 (0.0406)/1.2831 (0.0494)	1.2392 (0.0499)/1.4016 (0.0586)
	Proposed	**0.9527** (0.0253)/**1.0708** (0.0336)	**1.0336** (0.0362)/**1.1731** (0.0447)	**1.0951** (0.0411)/**1.2223** (0.0496)
(50K,5K)	Uniform	0.9887 (0.0289)/1.0943 (0.0366)	1.1846 (0.0451)/1.3382 (0.0539)	1.3629 (0.0608)/1.5191 (0.0696)
	Target-only	0.9813 (0.0267)/1.0914 (0.0348)	1.1432 (0.0411)/1.2997 (0.0496)	1.2846 (0.0535)/1.4465 (0.0622)
	Leverage	0.9721 (0.0257)/1.0841 (0.0339)	1.1658 (0.0440)/1.3221 (0.0527)	1.3368 (0.0587)/1.4956 (0.0674)
	MMD-only	0.9826 (0.0269)/1.0902 (0.0346)	1.1797 (0.0447)/1.3341 (0.0534)	1.3564 (0.0602)/1.5136 (0.0689)
	IPW-only	0.9659 (0.0248)/1.0806 (0.0330)	1.1504 (0.0432)/1.3093 (0.0519)	1.3091 (0.0563)/1.4708 (0.0650)
	RE-only	0.9608 (0.0242)/1.0761 (0.0325)	1.1234 (0.0404)/1.2834 (0.0491)	1.2401 (0.0496)/1.4032 (0.0583)
	Proposed	**0.9462** (0.0231)/**1.0642** (0.0316)	**1.0234** (0.0354)/**1.1658** (0.0439)	**1.0856** (0.0403)/**1.2141** (0.0488)
(200K,2K)	Uniform	0.9948 (0.0296)/1.0997 (0.0378)	1.1887 (0.0447)/1.3429 (0.0535)	1.3657 (0.0603)/1.5218 (0.0689)
	Target-only	0.9962 (0.0298)/1.1038 (0.0379)	1.1578 (0.0449)/1.3162 (0.0535)	1.2961 (0.0617)/1.4579 (0.0702)
	Leverage	0.9789 (0.0277)/1.0918 (0.0359)	1.1654 (0.0438)/1.3217 (0.0524)	1.3346 (0.0582)/1.4931 (0.0669)
	MMD-only	0.9881 (0.0286)/1.0961 (0.0365)	1.1829 (0.0443)/1.3374 (0.0532)	1.3594 (0.0599)/1.5176 (0.0685)
	IPW-only	0.9706 (0.0269)/1.0847 (0.0349)	1.1517 (0.0431)/1.3113 (0.0517)	1.3106 (0.0568)/1.4726 (0.0655)
	RE-only	0.9651 (0.0263)/1.0796 (0.0343)	1.1241 (0.0402)/1.2843 (0.0488)	1.2416 (0.0498)/1.4047 (0.0586)
	Proposed	**0.9498** (0.0248)/**1.0671** (0.0329)	**1.0172** (0.0348)/**1.1593** (0.0434)	**1.0796** (0.0395)/**1.2081** (0.0479)
(200K,5K)	Uniform	0.9806 (0.0278)/1.0873 (0.0359)	1.1765 (0.0439)/1.3312 (0.0526)	1.3538 (0.0596)/1.5104 (0.0683)
	Target-only	0.9813 (0.0267)/1.0914 (0.0348)	1.1432 (0.0411)/1.2997 (0.0496)	1.2846 (0.0535)/1.4465 (0.0622)
	Leverage	0.9671 (0.0256)/1.0796 (0.0338)	1.1613 (0.0432)/1.3178 (0.0519)	1.3298 (0.0581)/1.4891 (0.0667)
	MMD-only	0.9765 (0.0264)/1.0851 (0.0345)	1.1714 (0.0441)/1.3261 (0.0529)	1.3489 (0.0593)/1.5063 (0.0681)
	IPW-only	0.9596 (0.0247)/1.0738 (0.0326)	1.1486 (0.0428)/1.3075 (0.0516)	1.3071 (0.0563)/1.4687 (0.0648)
	RE-only	0.9541 (0.0241)/1.0689 (0.0321)	1.1217 (0.0396)/1.2818 (0.0482)	1.2407 (0.0495)/1.4028 (0.0581)
	Proposed	**0.9398** (0.0229)/**1.0576** (0.0313)	**1.0083** (0.0341)/**1.1507** (0.0426)	**1.0712** (0.0387)/**1.1998** (0.0472)

Table 14: DGP 4 (Heavy-tailed errors) with $r = 5000$

(n_S, n_T)	Method	$\nu = 5$	$\nu = 3$	$\nu = 2$
		MSE/Robust	MSE/Robust	MSE/Robust
(50K,2K)	Uniform	1.1827 (0.0548)/1.2936 (0.0624)	1.6235 (0.0912)/1.7628 (0.1027)	2.0526 (0.1243)/2.2875 (0.1396)
	Target-only	1.1045 (0.0486)/1.2184 (0.0573)	1.4116 (0.0785)/1.5527 (0.0892)	1.7824 (0.1108)/2.0217 (0.1262)
	Leverage	1.2067 (0.0569)/1.3218 (0.0651)	1.6904 (0.0963)/1.8357 (0.1087)	2.1218 (0.1296)/2.3654 (0.1463)
	MMD-only	1.1739 (0.0537)/1.2871 (0.0618)	1.5928 (0.0887)/1.7351 (0.1004)	2.0236 (0.1227)/2.2651 (0.1391)
	IPW-only	1.1681 (0.0531)/1.2804 (0.0613)	1.5746 (0.0872)/1.7218 (0.0989)	2.0069 (0.1213)/2.2551 (0.1376)
	RE-only	1.0641 (0.0447)/1.1769 (0.0536)	1.3187 (0.0718)/1.4619 (0.0837)	1.6935 (0.1056)/1.9278 (0.1215)
	Proposed	**1.0218** (0.0402)/**1.1246** (0.0495)	**1.2834** (0.0648)/**1.3527** (0.0735)	**1.5842** (0.0973)/**1.8047** (0.1126)
(50K,5K)	Uniform	1.1684 (0.0532)/1.2816 (0.0614)	1.6062 (0.0897)/1.7473 (0.1012)	2.0341 (0.1214)/2.2708 (0.1376)
	Target-only	1.0921 (0.0468)/1.2068 (0.0559)	1.3932 (0.0769)/1.5348 (0.0881)	1.7626 (0.1086)/2.0019 (0.1241)
	Leverage	1.1928 (0.0551)/1.3083 (0.0634)	1.6767 (0.0941)/1.8229 (0.1064)	2.1037 (0.1279)/2.3498 (0.1443)
	MMD-only	1.1574 (0.0526)/1.2718 (0.0608)	1.5797 (0.0874)/1.7236 (0.0991)	2.0074 (0.1206)/2.2508 (0.1370)
	IPW-only	1.1521 (0.0519)/1.2649 (0.0603)	1.5634 (0.0861)/1.7106 (0.0978)	1.9903 (0.1194)/2.2392 (0.1358)
	RE-only	1.0507 (0.0436)/1.1631 (0.0524)	1.3018 (0.0706)/1.4452 (0.0824)	1.6718 (0.1038)/1.9056 (0.1197)
	Proposed	**1.0116** (0.0394)/**1.1143** (0.0488)	**1.2696** (0.0637)/**1.3391** (0.0724)	**1.5669** (0.0958)/**1.7872** (0.1111)
(200K,2K)	Uniform	1.1967 (0.0559)/1.3098 (0.0642)	1.6397 (0.0911)/1.7814 (0.1026)	2.0675 (0.1246)/2.3038 (0.1407)
	Target-only	1.1045 (0.0486)/1.2184 (0.0573)	1.4116 (0.0785)/1.5527 (0.0892)	1.7824 (0.1108)/2.0217 (0.1262)
	Leverage	1.1818 (0.0547)/1.2961 (0.0629)	1.7079 (0.0973)/1.8539 (0.1096)	2.1362 (0.1303)/2.3819 (0.1468)
	MMD-only	1.1696 (0.0539)/1.2837 (0.0622)	1.6005 (0.0894)/1.7426 (0.1011)	2.0267 (0.1237)/2.2694 (0.1401)
	IPW-only	1.1607 (0.0532)/1.2736 (0.0614)	1.5846 (0.0882)/1.7318 (0.0998)	2.0089 (0.1221)/2.2581 (0.1386)
	RE-only	1.0563 (0.0449)/1.1697 (0.0538)	1.3247 (0.0721)/1.4683 (0.0839)	1.7018 (0.1049)/1.9364 (0.1210)
	Proposed	**1.0257** (0.0411)/**1.1298** (0.0503)	**1.2928** (0.0662)/**1.3624** (0.0748)	**1.5843** (0.0989)/**1.8061** (0.1143)
(200K,5K)	Uniform	1.1779 (0.0542)/1.2891 (0.0624)	1.6178 (0.0892)/1.7596 (0.1007)	2.0426 (0.1223)/2.2806 (0.1386)
	Target-only	1.0921 (0.0468)/1.2068 (0.0559)	1.3932 (0.0769)/1.5348 (0.0881)	1.7626 (0.1086)/2.0019 (0.1241)
	Leverage	1.1651 (0.0526)/1.2794 (0.0608)	1.6868 (0.0952)/1.8336 (0.1076)	2.1159 (0.1287)/2.3621 (0.1451)
	MMD-only	1.1523 (0.0518)/1.2668 (0.0602)	1.5906 (0.0885)/1.7331 (0.1002)	2.0164 (0.1216)/2.2591 (0.1380)
	IPW-only	1.1448 (0.0511)/1.2571 (0.0595)	1.5756 (0.0873)/1.7227 (0.0990)	1.9957 (0.1204)/2.2441 (0.1369)
	RE-only	1.0439 (0.0438)/1.1562 (0.0527)	1.3076 (0.0708)/1.4507 (0.0825)	1.6804 (0.1027)/1.9146 (0.1186)
	Proposed	**1.0154** (0.0398)/**1.1182** (0.0491)	**1.2761** (0.0647)/**1.3457** (0.0732)	**1.5597** (0.0951)/**1.7806** (0.1103)

Table 15: DGP 4 (Heavy-tailed errors) with $r = 10000$

(n_S, n_T)	Method	$\nu = 5$	$\nu = 3$	$\nu = 2$
		MSE/Robust	MSE/Robust	MSE/Robust
(50K,2K)	Uniform	1.1567 (0.0529)/1.2684 (0.0608)	1.5936 (0.0874)/1.7358 (0.0989)	2.0184 (0.1197)/2.2568 (0.1359)
	Target-only	1.0981 (0.0479)/1.2114 (0.0568)	1.4047 (0.0781)/1.5453 (0.0891)	1.7736 (0.1097)/2.0119 (0.1251)
	Leverage	1.1829 (0.0551)/1.2978 (0.0634)	1.6648 (0.0946)/1.8103 (0.1069)	2.0947 (0.1276)/2.3419 (0.1441)
	MMD-only	1.1478 (0.0517)/1.2606 (0.0599)	1.5662 (0.0859)/1.7074 (0.0976)	1.9963 (0.1184)/2.2361 (0.1347)
	IPW-only	1.1419 (0.0511)/1.2536 (0.0593)	1.5524 (0.0848)/1.6992 (0.0966)	1.9801 (0.1171)/2.2286 (0.1336)
	RE-only	1.0386 (0.0431)/1.1508 (0.0519)	1.2978 (0.0697)/1.4416 (0.0815)	1.6664 (0.1018)/1.9002 (0.1179)
	Proposed	**1.0071** (0.0392)/**1.1104** (0.0486)	**1.2637** (0.0639)/**1.3331** (0.0726)	**1.5481** (0.0946)/**1.7703** (0.1098)
(50K,5K)	Uniform	1.1412 (0.0516)/1.2531 (0.0597)	1.5776 (0.0861)/1.7206 (0.0978)	2.0017 (0.1181)/2.2398 (0.1343)
	Target-only	1.0867 (0.0464)/1.1996 (0.0553)	1.3901 (0.0768)/1.5314 (0.0879)	1.7581 (0.1083)/1.9964 (0.1237)
	Leverage	1.1631 (0.0534)/1.2776 (0.0616)	1.6541 (0.0932)/1.8007 (0.1056)	2.0836 (0.1267)/2.3311 (0.1432)
	MMD-only	1.1317 (0.0508)/1.2448 (0.0591)	1.5518 (0.0846)/1.6937 (0.0964)	1.9798 (0.1169)/2.2204 (0.1333)
	IPW-only	1.1261 (0.0501)/1.2384 (0.0586)	1.5386 (0.0837)/1.6851 (0.0954)	1.9657 (0.1158)/2.2139 (0.1321)
	RE-only	1.0268 (0.0426)/1.1382 (0.0515)	1.2837 (0.0689)/1.4274 (0.0806)	1.6498 (0.1009)/1.8831 (0.1170)
	Proposed	**0.9971** (0.0386)/**1.1008** (0.0481)	**1.2514** (0.0627)/**1.3211** (0.0714)	**1.5327** (0.0937)/**1.7564** (0.1089)
(200K,2K)	Uniform	1.1824 (0.0546)/1.2948 (0.0628)	1.6231 (0.0906)/1.7642 (0.1023)	2.0508 (0.1239)/2.2884 (0.1403)
	Target-only	1.0981 (0.0479)/1.2114 (0.0568)	1.4047 (0.0781)/1.5453 (0.0891)	1.7736 (0.1097)/2.0119 (0.1251)
	Leverage	1.1737 (0.0538)/1.2886 (0.0621)	1.6828 (0.0951)/1.8291 (0.1075)	2.1104 (0.1278)/2.3572 (0.1442)
	MMD-only	1.1439 (0.0511)/1.2576 (0.0596)	1.5678 (0.0868)/1.7084 (0.0987)	1.9951 (0.1196)/2.2358 (0.1360)
	IPW-only	1.1384 (0.0506)/1.2507 (0.0591)	1.5531 (0.0856)/1.6997 (0.0974)	1.9796 (0.1183)/2.2279 (0.1348)
	RE-only	1.0349 (0.0429)/1.1471 (0.0518)	1.2907 (0.0696)/1.4348 (0.0813)	1.6613 (0.1007)/1.8958 (0.1167)
	Proposed	**1.0058** (0.0389)/**1.1093** (0.0484)	**1.2438** (0.0618)/**1.3142** (0.0705)	**1.5219** (0.0926)/**1.7463** (0.1078)
(200K,5K)	Uniform	1.1648 (0.0528)/1.2761 (0.0609)	1.6043 (0.0884)/1.7461 (0.1001)	2.0306 (0.1211)/2.2694 (0.1373)
	Target-only	1.0867 (0.0464)/1.1996 (0.0553)	1.3901 (0.0768)/1.5314 (0.0879)	1.7581 (0.1083)/1.9964 (0.1237)
	Leverage	1.1574 (0.0521)/1.2716 (0.0604)	1.6746 (0.0941)/1.8218 (0.1064)	2.0987 (0.1263)/2.3452 (0.1428)
	MMD-only	1.1286 (0.0497)/1.2421 (0.0583)	1.5487 (0.0843)/1.6906 (0.0961)	1.9758 (0.1164)/2.2163 (0.1329)
	IPW-only	1.1229 (0.0492)/1.2348 (0.0577)	1.5361 (0.0831)/1.6827 (0.0948)	1.9624 (0.1153)/2.2107 (0.1317)
	RE-only	1.0216 (0.0418)/1.1339 (0.0507)	1.2761 (0.0687)/1.4206 (0.0803)	1.6407 (0.0996)/1.8752 (0.1156)
	Proposed	**0.9967** (0.0381)/**1.1001** (0.0476)	**1.2318** (0.0609)/**1.3026** (0.0696)	**1.5058** (0.0911)/**1.7297** (0.1063)

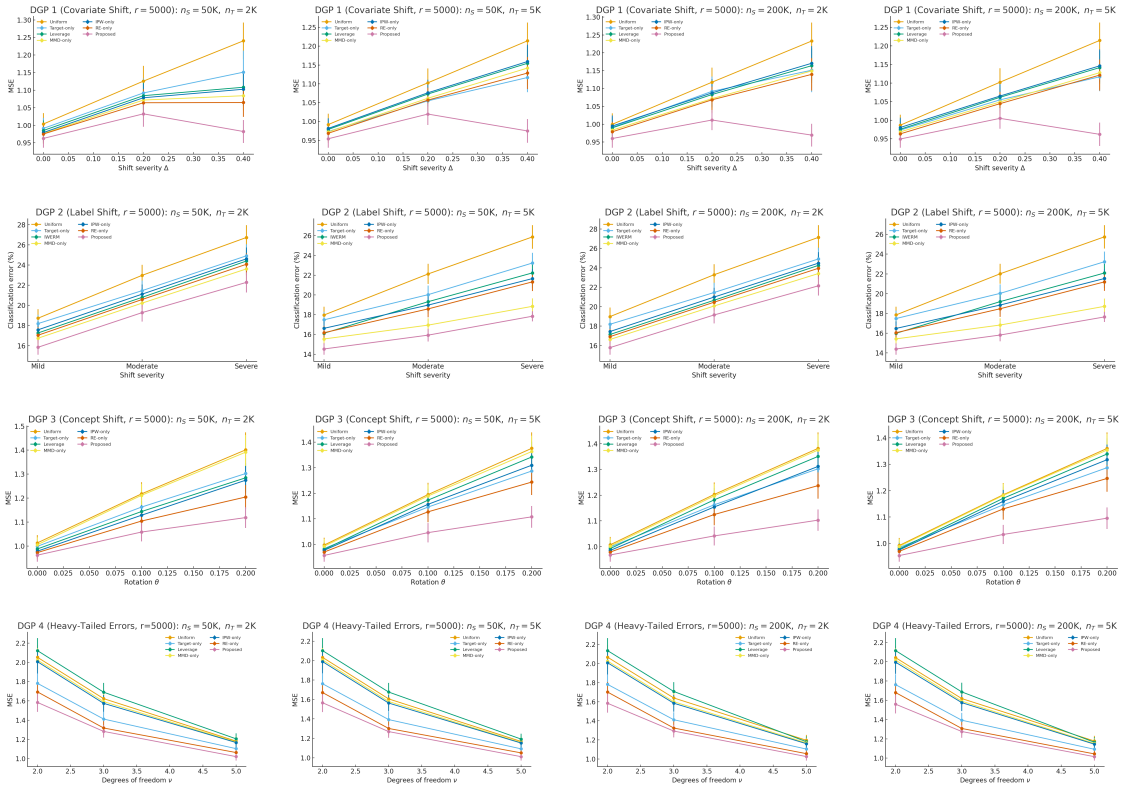

Figure 6: MSE/Error±SE under Different DGP 1—$r = 5000$

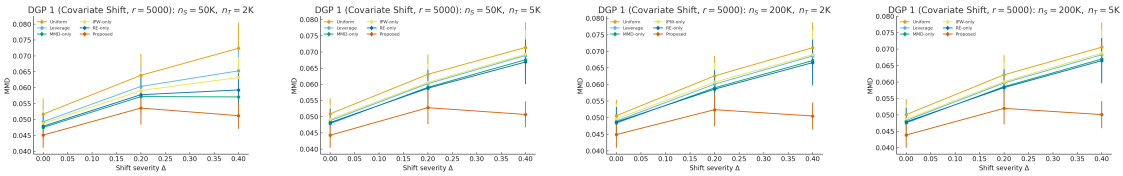

Figure 7: MMD±SE under Different DGP 1—$r = 5000$

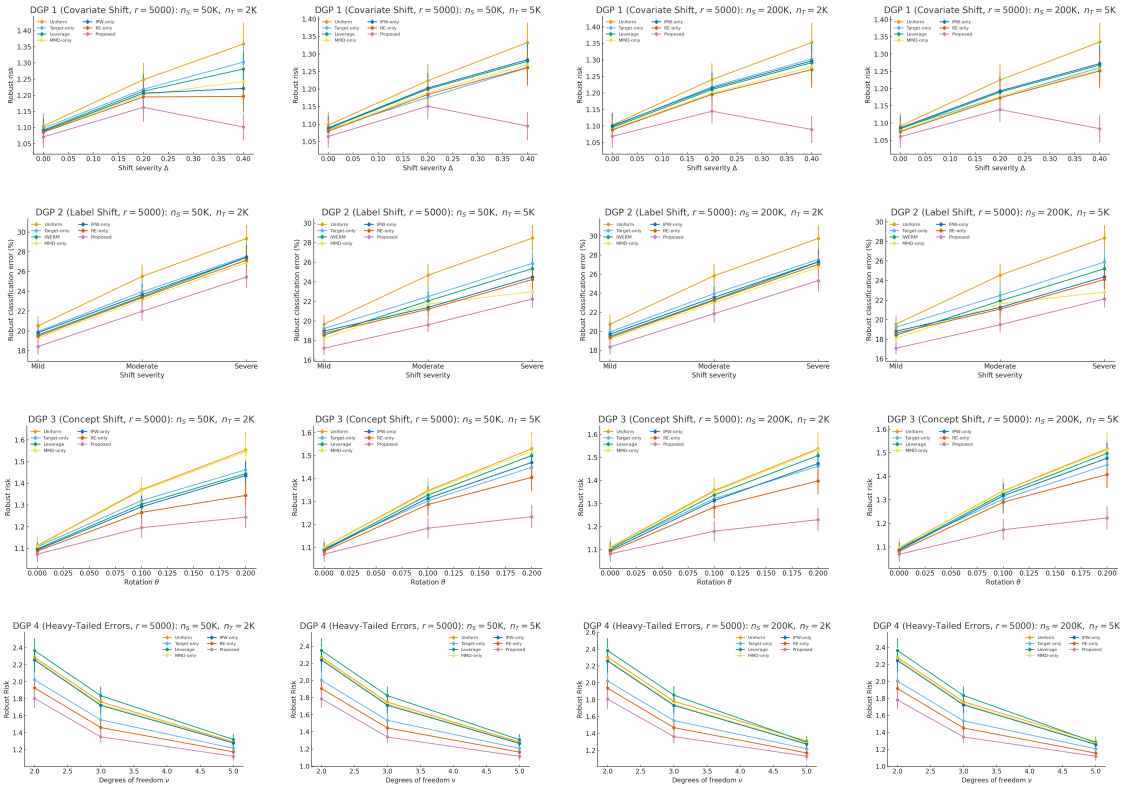

Figure 8: Robust Risk±SE under Different DGP 1—$r = 5000$

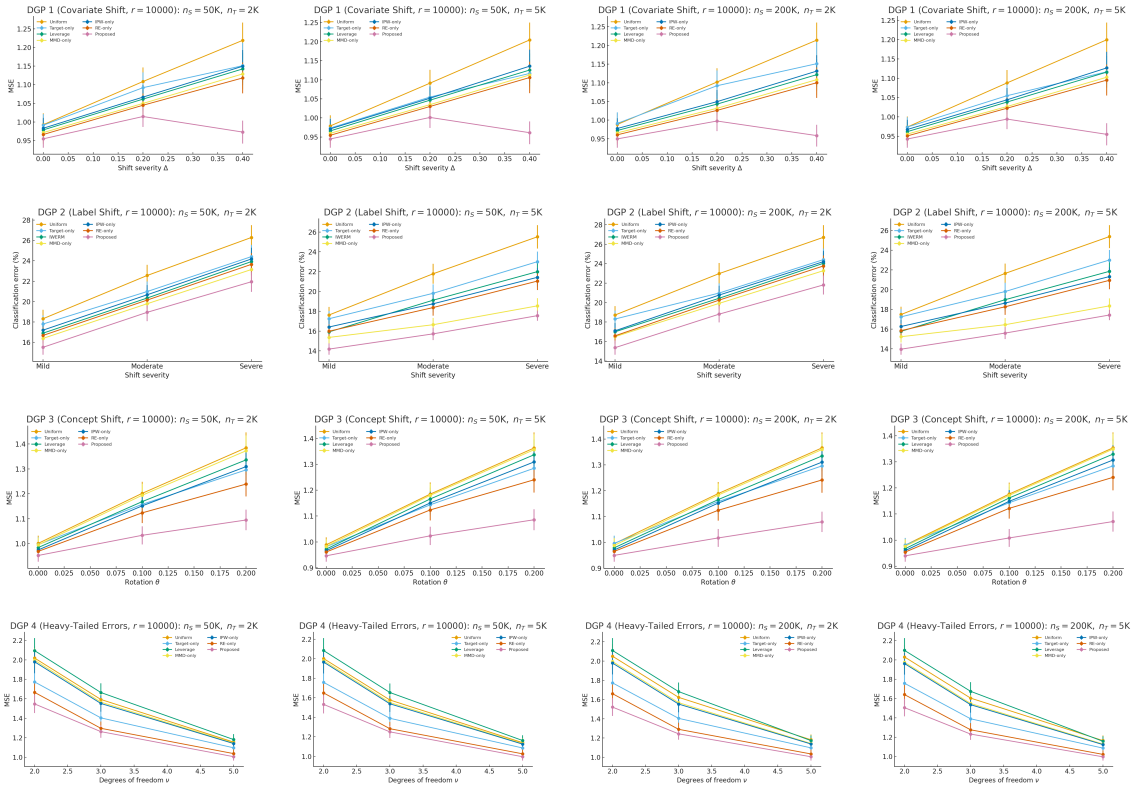

Figure 9: MSE/Error±SE under Different DGP 1—$r = 10000$

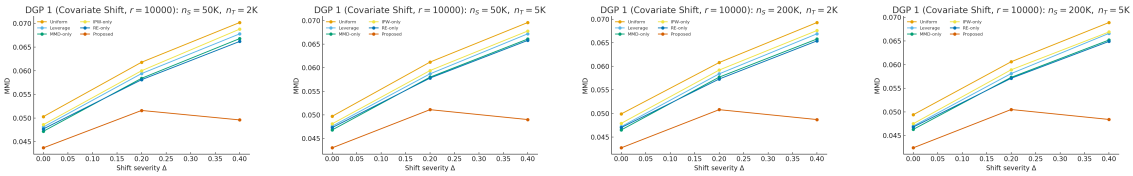

Figure 10: MMD±SE under Different DGP 1—$r = 10000$

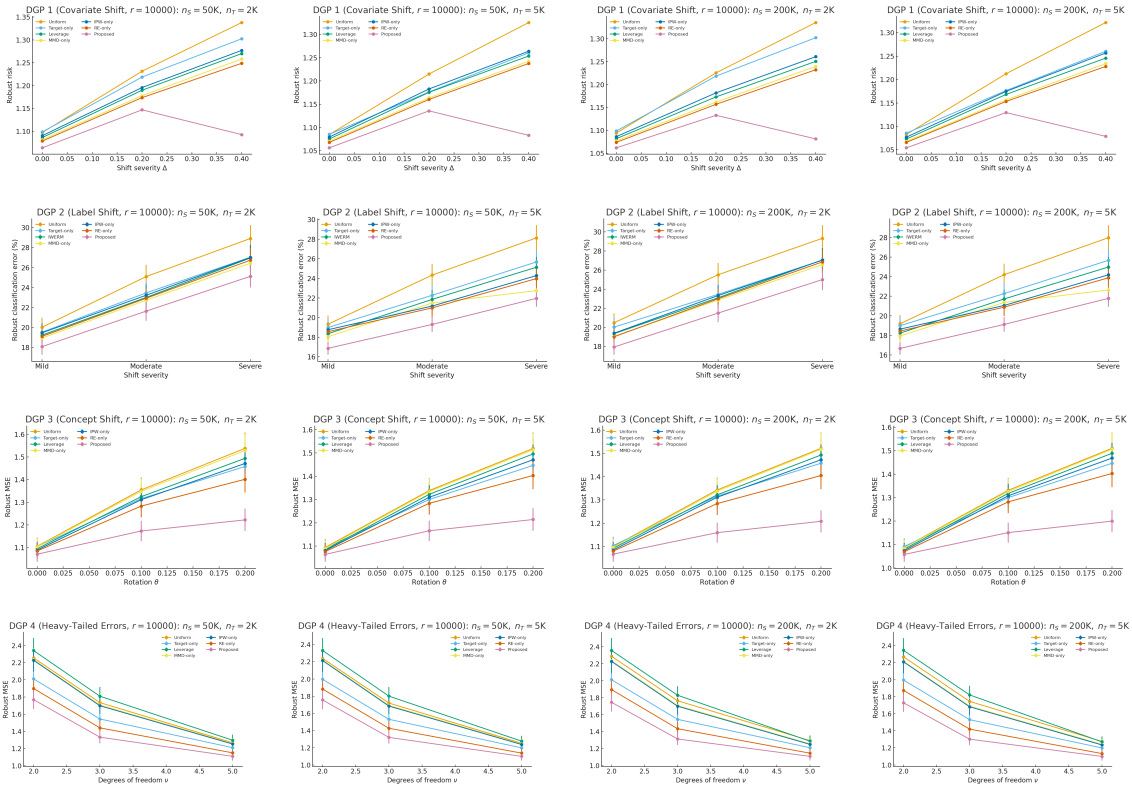

Figure 11: Robust Risk±SE under Different DGP 1—$r = 10000$

References

Ai, M., Yu, J., Zhang, H., and Wang, H. (2021). Optimal Subsampling Algorithms for Big Data Regressions. *Statistica Sinica*, 31, 749-772.

Andrews, D. W. K. (1992). Generic Uniform Convergence. *Econometric Theory*, 8 (2), 241-257.

Arlot, S. and Celisse, A. (2010). A Survey of Cross-Validation Procedures for Model Selection. *Statistics Surveys*, 4, 40-79.

Bachem, O., Lucic, M., and Krause, A. (2017). Practical Coreset Constructions for Machine Learning. *arXiv:1703.06476*.

Bartlett, P. L. and Mendelson, S. (2006). Empirical Minimization. *Probability Theory and Related Fields*, 135, 311-334.

Ben-Tal, A., den Hertog, D., De Waegenaere, A., Melenberg, B., and Rennen, G. (2012). Robust Solutions of Optimization Problems Affected by Uncertain Probabilities. *Management Science*, 59 (2), 341-357.

Blanchet, J. and Murthy, K. (2019). Quantifying Distributional Model Risk via Optimal Transport. *Mathematics of Operations Research*, 44 (2), 565-600.

Blitzer, J., Dredze, M., and Pereira, F. (2007). Biographies, Bollywood, Boom-Boxes and Blenders: Domain Adaptation for Sentiment Classification. *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, 440-447.

Boucheron, S., Lugosi, G., and Massart, P. (2013). *Concentration Inequalities: A Nonasymptotic Theory of Independence*. Online edn, Oxford Academic.

Boutsidis, C., Drineas, P., and Magdon-Ismail, M. (2013). Near-Optimal Column-Based Matrix Reconstruction. *SIAM Journal on Computing*, 43 (2), 687-717.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. (2020). Language Models are Few-Shot Learners. *Proceedings of the 34th International Conference on Neural Information Processing Systems*, 1877-1901.

Chen, R. and Paschalidis, I. Ch. (2018). *Distributionally Robust Learning*. Now Foundations and Trends.

Clarkson, K. L. and Woodruff, D. P. (2017). Low-Rank Approximation and Regression in Input Sparsity Time. *Journal of the ACM*, 63 (6), 1-45.

Coello, C. A. C., Pulido, G. T., and Lechuga, M. S. (2004). Handling Multiple Objectives with Particle Swarm Optimization. *IEEE Transactions on Evolutionary Computation*, 8 (3), 256-279.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2019). BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. *Proceedings of NAACL-HLT*, 4171-4186.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021). An Image is Worth 16×16 Words: Transformers for Image Recognition at Scale. *International Conference on Learning Representations*.

Drineas, P., Mahoney, M. W., and Muthukrishnan, S. (2006). Sampling Algorithms for l_2 Regression and Applications. *Proceedings of the 17th Annual ACM-SIAM Symposium on Discrete Algorithm*, 1127-1136.

Duchi, J. C. and Namkoong, H. (2021). Learning Models with Uniform Performance via Distributionally Robust Optimization. *Annals of Statistics*, 49 (3), 1378-1406.

Esfahani, P. M. and Kuhn, D. (2018). Data-Driven Distributionally Robust Optimization Using the Wasserstein Metric: Performance Guarantees and Tractable Reformulations. *Mathematical Programming*, 171, 115-166.

Fournier, N. and Guillin, A. (2013). On the Rate of Convergence in Wasserstein Distance of the Empirical Measure. *arXiv:1312.2128v1*.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. (2016). Domain-Adversarial Training of Neural Networks. *Journal of Machine Learning Research*, 17, 1-35.

Gao, R. and Kleywegt, A. (2022). Distributionally Robust Stochastic Optimization with Wasserstein Distance. *Mathematics of Operations Research*, 48 (2), 603-655.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. (2012). A Kernel Two-Sample Test. *Journal of Machine Learning Research*, 13, 723-773.

Györfi, L., Kohler, M., Krzyżak, A., and Walk, H. (2002). *A Distribution-Free Theory of Nonparametric Regression*. Springer New York.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep Residual Learning for Image Recognition. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 770-778.

Horvitz, D. G. and Thompson, D. J. (1952). A Generalization of Sampling without Replacement from a Finite Universe. *Journal of the American Statistical Association*, 47 (260), 663-685.

Huang, J., Smola, A. J., Gretton, A., Borgwardt, K. M., and Schölkopf, B. (2006). Correcting Sample Selection Bias by Unlabeled Data. *Proceedings of the 20th International Conference on Neural Information Processing Systems*, 601-608.

Kennedy, J. and Eberhart, R. (1995). Particle Swarm Optimization. *Proceedings of IEEE International Conference on Neural Networks*, 4, 1942-1948.

Li, M., Andersen, D. G., Park, J. W., Smola, A. J., Ahmed, A., Josifovski, V., Long, J., Shekita, E. J., and Su, B.-Y. (2014). Scaling Distributed Machine Learning with the Parameter Server. *Proceedings of the 11th USENIX Symposium on Operating Systems Design and Implementation*, 583-598.

Lukemire, J., Mandal, A., and Wong, W. K. (2018). d-QPSO: A Quantum-Behaved Particle Swarm Technique for Finding D-optimal Designs with Discrete and Continuous Factors and a Binary Response. *Technometrics*, 61 (1), 77-87.

Ma, P., Mahoney, M. W., and Yu, B. (2015). A Statistical Perspective on Algorithmic Leveraging. *Journal of Machine Learning Research*, 16, 861-911.

Maalouf, A., Tukan, M., Braverman, V., and Rus, D. (2023). AutoCoreset: An Automatic Practical Coreset Construction Framework. *Proceedings of the 40th International Conference on Machine Learning*, 23451-23466.

Mansour, Y., Mohri, M., and Rostamizadeh, A. (2008). Domain Adaptation with Multiple Sources. *Proceedings of the 22nd International Conference on Neural Information Processing Systems*, 1041-1048.

Munteanu, A. and Schwiegelshohn, C. (2018). Coresets-Methods and History: A Theoreticians Design Pattern for Approximation and Streaming Algorithms. *KI-Künstliche Intelligenz*, 32, 37-53.

Ning, Y. and Liu, H. (2017). A General Theory of Hypothesis Tests and Confidence Regions for Sparse High-Dimensional Models. *Annals of Statistics*, 45 (1), 158-195.

Pan, S. J. and Yang, Q. (2010). A Survey on Transfer Learning. *IEEE Transactions on Knowledge and Data Engineering*, 22 (10), 1345-1359.

Peyré, G. and Cuturi, M. (2019). *Computational Optimal Transport: With Applications to Data Science*. Now Foundations and Trends.

Poli, R., Kennedy, J., and Blackwell, T. (2007). Particle Swarm Optimization: An Overview. *Swarm Intelligence*, 1, 33-57.

Prettenhofer, P. and Stein, B. (2010). Cross-Language Text Classification Using Structural Correspondence Learning. *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, 1118-1127.

Quiñonero-Candela, J., Sugiyama, M., Schwaighofer, A., and Lawrence, N. D. (2009). *Dataset Shift in Machine Learning*. MIT Press.

Raghu, M., Zhang, C., Kleinberg, J., and Bengio, S. (2019). Transfusion: Understanding Transfer Learning for Medical Imaging. *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, 3347-3357.

Rahimi, A. and Recht, B. (2007). Random Features for Large-Scale Kernel Machines. *Proceedings of the 21st International Conference on Neural Information Processing Systems*, 1177-1184.

Rosenbaum, P. R. and Rubin, D. B. (1983). The Central Role of the Propensity Score in Observational Studies for Causal Effects. *Biometrika*, 70 (1), 41-55.

Shafieezadeh-Abadeh, S., Esfahani, P. M., and Kuhn, D. (2015). Distributionally Robust Logistic Regression. *Proceedings of the 29th International Conference on Neural Information Processing Systems*, 1576-1584.

Shalev-Shwartz, S. and Ben-David, S. (2014). *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press.

Shimodaira, H. (2000). Improving Predictive Inference Under Covariate Shift by Weighting the Log-Likelihood Function. *Journal of Statistical Planning and Inference*, 90 (2), 227-244.

Shin, H.-C., Roth, H. R., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D., and Summers, R. M. (2016). Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning. *IEEE Transactions on Medical Imaging*, 35 (5), 1285-1298.

Stehlik, M., Wong, W. K., Chen, P. Y., and Kisevlak, J. (2024). A NovelDoubleExponential Particle Swarm Optimization (DExPSO) with Guaranteed Convergence and Applications to Find Optimal Exact Designs. *Applied Soft Computing Journal*, 163, 111913.

Sugiyama, M., Krauledat, M., and Müller, K.-R. (2007). Covariate Shift Adaptation by Importance Weighted Cross-Validation. *Journal of Machine Learning Research*, 8, 985-1005.

Sutherland, D. J. and Schneider, J. (2015). On the Error of Random Fourier Features. *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*, 862-871.

Tsiatis, A. A. (2006). *Semiparametric Theory and Missing Data*. Springer New York.

Tsybakov, A. B. (2009). *Introduction to Nonparametric Estimation*. Springer New York.

Tzeng, E., Hoffman, J., Saenko, K., and Darrell, T. (2017). Adversarial Discriminative Domain Adaptation. *IEEE Conference on Computer Vision and Pattern Recognition*, 2962-2971.

vaart, A. W. and Wellner, J. A. (1996). *Weak Convergence and Empirical Processes: With Applications to Statistics*. Springer New York.

van de Geer, S., Bühlmann, P., Ritov, Y., and Dezeure, R. (2014). On Asymptotically Optimal Confidence Regions and Tests for High-Dimensional Models. *Annals of Statistics*, 42 (3), 1166-1202.

Venkateswara, H., Eusebio, J., Chakraborty, S., and Panchanathan, S. (2017). Deep Hashing Network for Unsupervised Domain Adaptation. *IEEE Conference on Computer Vision and Pattern Recognition*, 5385-5394.

Vershynin, R. (2018). *High-Dimensional Probability: An Introduction with Applications in Data Science*. Cambridge University Press.

Wang, H. and Ma, Y. (2021). Optimal Subsampling for Quantile Regression in Big Data. *Biometrika*, 108 (1), 99-112.

Wang, H., Zhu, R., and Ma, P. (2018). Optimal Subsampling for Large Sample Logistic Regression. *Journal of the American Statistical Association*, 113 (522), 829-844.

Weed, J. and Bach, F. (2019). Sharp Asymptotic and Finite-Sample Rates of Convergence of Empirical Measures in Wasserstein Distance. *Bernoulli*, 25 (4A), 2620-2648.

Zhang, K., Schölkopf, B., Muandet, K., and Wang, Z. (2013). Domain Adaptation under Target and Conditional Shift. *Proceedings of the 30th International Conference on International Conference on Machine Learning*, 819-827.

Zhang, S., Yao, L., Sun, A., and Tay, Y. (2019). Deep Learning Based Recommender System: A Survey and New Perspectives. *ACM Computing Surveys*, 52 (1), 1-38.

Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., and He, Q. (2021). A Comprehensive Survey on Transfer Learning. *Proceedings of the IEEE*, 109 (1), 43-76.