Towards Better Generalization on Test Data with Distributional Uncertainty

Pengfei Tian*, Hao Sha*, Jiayun Wu, Yue He, and Peng Cui

Abstract

Due to the shift in distribution from the training to the test dataset, machine learning models are prone to experience a degradation in performance when evaluated on the test distribution. Distributional Robust Optimization (DRO) addresses distribution shifts by characterizing the uncertainty in data distributions, thereby enhancing a model's generalization capability by improving its performance under worst-case scenarios. However, in practical scenarios, it is often feasible to obtain a small portion of unlabeled test data. Ignoring this information can lead to an overly pessimistic approach to the traditional DRO method. In this paper, we investigate the problem of enhancing a model's ability to handle the distributional uncertainty surrounding the test data. We propose a novel method Against Test Data Uncertainty (ATDU) which quantifies uncertainty using the training distribution dataset and adapts it to the test distribution guided by the unlabeled target distribution data. ATDU employs bootstrap resampling to estimate model risk on the posterior distribution of the target distribution and optimizes the α quantile loss for robust prediction. Our theoretical analysis elucidates the properties of the sampling method and provides a comprehensive examination of ATDU. A range of real-world data experiments have demonstrated the effectiveness of our method.

1 Introduction

Despite the widespread success of machine learning across various areas, model performance decreases under distribution shifts between the training and test distributions [Taori et al., 2020, Wiles et al., 2021, Koh et al., 2021]. The distribution shift consists of two components: covariate shift which affects the marginal distribution of features $P(X)$, and concept shift which affects the conditional distribution of outcomes $P(Y|X)$. Since the outcome of test data is not available during the training phase, the *uncertainty* of the test distribution, due to potential covariate shifts and concept shifts, makes it nontrivial to achieve out-of-distribution generalization.

Distributionally robust optimization (DRO) [Rahimian and Mehrotra, 2019, Bertsimas et al., 2019] models the uncertainty of the test distribution by an uncertainty set centered around the training distribution, and optimizes for the worst case within this set, as illustrated in Figure 1. DRO provides robustness guarantees when the target distribution falls within the uncertainty set. However, DRO is known to suffer from over-pessimism [Liu et al., 2022, 2023b], when the uncertainty set is excessively large to contain the test distribution (the blue circle in Figure 1). In such a scenario, the worst case in the uncertainty set may reflect an unrealistic distribution, leading to impractically low performance. Conversely, a small uncertainty set (the red circle in Figure 1) is over-optimistic and lacks a robustness

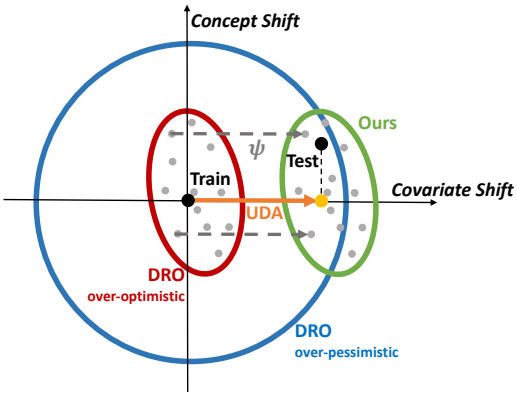

Figure 1: UDA method adapts training distribution toward test distribution assuming no concept shift. DRO method optimizes the worst case of the perturbed distributions around the training distribution. Our method perturbs the source distribution and adapts it to the target distribution according to covariate shift direction.

guarantee, as the true test distribution falls outside the optimization region. The paradox arises from the unregularized uncertainty set, which lacks prior information about the test distribution.

To address the over-pessimism of DRO, we propose incorporating unlabelled test data to better inform the uncertainty set. This approach is more general than a related trend of methods known as unsupervised domain adaptation (UDA) [Ganin and Lempitsky, 2015, Oza et al., 2023], which fully trust the unlabelled test data and performs transfer learning through techniques like sample re-weighting [Chen et al., 2018, Liu et al., 2023c] or generating pseudo labels via self-supervised learning [Liang et al., 2019, Li et al., 2023], among others. Current UDA approaches predominantly assume that the unlabeled test data is unbiased and that only covariate shift occurs (the orange arrow in Figure 1) [Sugiyama et al., 2007, Qiu et al., 2023], while neglecting the distributional uncertainty in the test data, particularly uncertainties caused by concept shifts.

Therefore, we propose leveraging the unlabelled test data to design a novel uncertainty set that mitigates the over-pessimism of traditional DRO methods, accounting for distributional uncertainty induced by both covariate and concept shifts. As shown in Figure 1, we transform the uncertainty set centered around the training distribution $\mathcal{P}$ to an uncertainty set around the test distribution $\Psi\mathcal{P}$ (the green circle in Figure 1). This transformation Ψ (the grey arrow in Figure 1) is guided by the direction of covariate shift inferred from the unlabelled test data, such that for any $P(x, y) \in \mathcal{P}$, we have $\Psi P(x, y) \propto P(x, y) \cdot P_{\text{test}}(x)/P_{\text{train}}(x)$ [1]. With this adapted uncertainty set $\Psi\mathcal{P}$, which is obtained by mapping each distribution in the training-distribution-centered set $\mathcal{P}$ and a predictive function class $\mathcal{F}$, we present the following minimax objective:

$$\min_{f\in\mathcal{F}} \max_{P\in\Psi\mathcal{P}} \mathbb{E}_P\left[(y - f(x))^2\right]. \tag{1}$$

The transformed uncertainty set covers the test distribution with a relatively small size,

[1] For each $P_e(x, y) \in \mathcal{P}$, the transformation Ψ satisfies: $\Psi P_e(y \mid x) = P_e(y|x)$ (no concept shift), $\Psi P_e(x)/P_e(x) \propto P_{\text{test}}(x)/P_{\text{train}}(x)$ (along the covariate shift direction).

leading to a class of more practical distributions that resolves the over-pessimism issue of DRO. Compared to UDA, our learning objective is more robust, as it incorporates an adversary to address distributional uncertainty around the test data. A formal comparison between our learning objective and those of DRO and UDA is presented in Table 1.

We devise a tractable solution for Equation (1). First, we implement the uncertainty set around the train distribution by bootstrap [Efron and Tibshirani, 1994, Chernozhukov et al., 2023]. Bootstrap is effective in simulating the sampling uncertainty of the original dataset by resampling. Bootstrap can simulate not only the region but also the likelihood distribution of the region, that the distribution closer to the training distribution has a large likelihood to be generated. Second, we transform the uncertainty set by a non-parametric maximum likelihood estimation to sample reweight and that's where we incorporate test data and transfer the likelihood toward the test distribution. Last, we do not solve for a hard maximization but a soft maximization implemented by quantile loss.

Our contribution is summarized as follows:

1. We provide a minimax framework to formulate and address distributional uncertainty using unlabelled test data, which overcomes the over-pessimism associated with traditional distributional robustness methods. The objective is optimized by bootstrap which is easier and more efficient than current DRO optimization techniques such as alternate training.

2. We theoretically analyze the properties of our sampling method and provide a thorough analysis of our method, highlighting its robustness and efficacy in handling uncertainty in distribution shift scenarios. We compare our method has a tighter guarantee than the UDA method and DRO method, and we compute the convergence rate when the model is correct.

3. We conduct extensive experiments on a benchmark of real-world distribution shifts [Liu et al., 2024], focusing on US census tabular data with prevalent concept shift, as well as on standard vision benchmarks for domain adaptation. Our evaluation includes 6 datasets in total with thousands of domain pairs. Our method consistently achieves the best performance in the vast majority of these settings.

2 Related Work

2.1 Distributional Robust Optimization

Distributional robust optimization (DRO) aims to optimize the worst-case performance over a set of distributions to ensure out-of-distribution (OOD) generalization performance [Rahimian and Mehrotra, 2019, Wu et al., 2024]. Traditional DRO methods describe the uncertainty of the training set without considering the test distribution. Using test covariate information, Frogner et al. [2019] optimized the worst-case performance within the intersection of the uncertainty regions of both training and test distributions. Compared to their method, our method has a smoother uncertainty region boundary and equipped the likelihood distribution on region, which leads to a more accurate distribution estimation and a tighter loss guarantee.

Bayesian distributionally robust optimization assumes a likelihood parametric model to compute the posterior distribution and samples from it [Wu et al., 2018, Shapiro et al., 2023]. Our method can be understood as a nonparametric version of Bayesian distribution

Table 1: Comparison of Learning Objectives.

Method	Optimization objectives
UDA	$\min_{f\in\mathcal{F}} \mathbb{E}_{\text{train}}\left[(y-f(x))^2 \frac{P_{\text{test}}(x)}{P_{\text{train}}(x)}\right]$
DRO	$\min_{f\in\mathcal{F}} \max_{P\in\mathcal{P}} \mathbb{E}_{\text{train}}\left[(y-f(x))^2 \frac{P(x,y)}{P_{\text{train}}(x,y)}\right]$
Ours	$\min_{f\in\mathcal{F}} \max_{P_e\in\mathcal{P}} \mathbb{E}_{\text{train}}\left[(y-f(x))^2 \frac{P_e(x,y)}{P_{\text{train}}(x,y)} \frac{P_{\text{test}}(x)}{P_{\text{train}}(x)}\right]$

optimization and extended the application scopes with the test data. In contrast to Bayesian distributionally robust optimization, our method does not rely on a parametric model. We operate directly on the dataset to address domain adaptation challenges.

2.2 Domain Adaptation

To facilitate model generalization to target domains with data distribution shifts, various domain adaptation techniques have been proposed. Instance-based methods [Jiang and Zhai, 2007, Warke et al., 2024] reweight or select training samples to align with the test distribution. Feature-based approaches [Sun and Saenko, 2016a, Du et al., 2023] focus on aligning or transferring feature representations between domains. Model-based techniques [Ganin et al., 2016a, Motiian et al., 2017] aim to create models robust to domain shifts or specifically adapted for the target domain.

To tackle the challenge of unlabeled test domain data, unsupervised domain adaptation techniques have been developed. These methods Chen et al. [2018], Liu et al. [2023c] typically transfer the distribution from source to target domains through sample reweighting. Self-supervised learning Liang et al. [2019], Li et al. [2023] generates pseudo-labels for unlabeled data to augment the training set. Other strategies Taigman et al. [2017], Tzeng et al. [2017] involve using generative models to synthesize target domain samples and employing adversarial training to align features between source and target domains. Current approaches always assume the unlabeled test domain data is unbiased. However, sample selection bias easily happens in realistic scenarios. This paper first investigates the uncertainty of test domain distribution in the unsupervised domain adaptation problem.

2.3 Learning Objectives Comparison

We compare the learning objectives and rewrite them as the expectation on training distribution in Table 1. DRO does not introduce test distribution, while UDA assumes $P_{\text{train}}(y|x) = P_{\text{test}}(y|x)$. Our objectives degenerates to DRO forms when $P_{\text{train}}(x) = P_{\text{test}}(x)$ and to UDA forms when $\mathcal{P} = \{P_{\text{train}}(x,y)\}$.

3 Method

Settings Traditional prediction task is to predict outcome or label Y with feature X. We assume $(X^{\mathrm{T}}, Y)^{\mathrm{T}} \sim \mathbb{D}_\eta$, where $\mathbb{D}_\eta$ is a distribution parameterized by η. Specifically, we label training distribution and test distribution with parameters η_S and η_T and simply represent the distribution as $\mathbb{D}_S$ and $\mathbb{D}_T$, respectively. In many scenarios, training distribution $\mathbb{D}_S$ differs from test distribution $\mathbb{D}_T$ [Zhou and Xie, 2015, Saito et al., 2018]. Prediction model

construct by training data $\mathcal{D}_S = \{(X_i, Y_i), i = 1, \ldots, I\}$ and the unlabelled test data $\mathcal{U}_T = \{\tilde{X}_j, j = 1, \ldots, J\}$. We train a model f_θ based on them aimed to predict label of the test data, where θ is the model parameter. To evaluate the prediction, we use $l(Y_j, f_\theta(\tilde{X}_j))$, such as L_2 loss $(Y_j - f_\theta(\tilde{X}_j))^2$ to evaluate the prediction model f_θ on test data $\tilde{X}_j$. We define $L(\theta, \eta) = \mathbb{E}_{(x,y)\sim\mathbb{D}_\eta}[l(Y, f_\theta(X))]$ as the loss of model f_θ on distribution $\mathbb{D}_\eta$. Therefore, the test distribution loss is $L(\theta, \eta_T)$.

Empirical risk minimization obtains θ by optimizing $L(\theta, \eta_S)$ and does not utilize any information of $\mathcal{U}_T$. In contrast, many unlabeled domain adaption method estimate η_T using $\mathcal{U}_T$ and optimize $L(\theta, \eta_T)$. However, it relies on no concept shift which is suitable for image data but usually violated in tabular data.

We generate B datasets $\mathcal{D}^{(b)} = \{(X_{Z_b(s)}, Y_{Z_b(s)})\}_{s=1}^S, b = 1, \ldots, B$. Each dataset is of size S and sampled from $\mathcal{D}_S$ with replacement that $\Pr(Z_b(s) = i) = 1/I$ for $i = 1, \ldots, I$. As pointed out by the bootstrap literature, these B datasets describes the uncertainty of $\mathcal{D}_S$. Considering the shift between $\mathcal{D}^{(b)}$ and $\mathcal{D}_T$, if we have weights $\{w^{(b)}_{Z_b(s)}\}_{s=1}^S$ adapting to test domain, we use weighted loss as following

$$L_b = \sum_{s=1}^{S} w^{(b)}_{Z_b(s)} l(f_\theta(X_{Z_b(s)}), Y_{Z_b(s)}).$$

Then $\{L_b\}_{b=1}^B$ estimate the uncertainty of the loss on the test distribution. We can optimize the α-quantile loss in $\{L_b\}$ in each step until convergence. Algorithm 1 provides the pseudo algorithm.

Here we wish the weights can describe uncertainty of the distribution or estimate the likelihood of the η_T. Traditional weighting methods such as importance weighting are easy to be large since the inverse probability function on the denominator. Traditional weighting method leads to extreme weights and is affected by the density approximation error. Bayesian is also alternative to describe distribution uncertainty but rely on parametric models. Here we introduce empirical likelihood [Owen, 2001, Chen et al., 2008b, Liu, 2023] to provide non-extreme weights with likelihood meaning nonparametrically as follows.

$$\begin{aligned} &\max \sum_{s=1}^{S} \log w^{(b)}_{Z_b(s)} \\ &\text{subject to} \quad \sum_{s=1}^{S} w^{(b)}_{Z_b(s)} = 1, \quad \sum_{s=1}^{S} w^{(b)}_{Z_b(s)} X_{Z_b(s)} = \frac{1}{J}\sum_{j=1}^{J} \tilde{X}_j, \quad w^{(b)}_{Z_b(s)} > 0. \end{aligned} \tag{2}$$

Procedure (2) is a extension for classic maximum likelihood estimation.

4 Theory

Algorithm 1

Input: the labelled training dataset $\mathcal{D}_S = \{(X_i, Y_i)\}_{i=1}^I$, the unlabelled test dataset $\mathcal{U}_T = \{\tilde{X}_j\}_{j=1}^J$, sampling parameter B, quantile parameter α and distribution parameters and model parameters θ.
for $b = 1$ to B **do**
  Sample dataset $\mathcal{D}^{(b)} = \{(X_{Z_b(s)}, Y_{Z_b(s)})\}_{s=1}^S$ from the labelled training dataset $\mathcal{D}_S$ with replacement that $\Pr(Z_b(s) = i) = 1/I$.
  Obtain $\{w_s^{(b)}\}_{s=1}^S$ from Equation (2).
  repeat
    for $b = 1$ to B **do**
      Compute loss $L_b = \sum_{s=1}^S w_s^{(b)} l(Y_{Z_b(s)}, f_\theta(X_{Z_b(s)}))$.
    end for
    Rank $\{L_b\}$ and choose $L_{\lceil \alpha B \rceil}$ to conduct gradient decent and update parameters θ.
  until θ converge
end for
return: the prediction model $f_\theta(\cdot)$ and label $\{f_\theta(\tilde{X}_j)\}_{j=1}^J$.

4.1 Empirical likelihood weighting for distribution shift

Define $\mu = I^{-1}\sum_{i=1}^I X_i$ and $\tilde{\mu} = J^{-1}\sum_{j=1}^{\tilde{J}} X_j$. We solve Equation (2) by Lagrange method, and obtain

$$w_i^{(b)} = \frac{1}{S(\lambda_b^{\mathrm{T}}(X_i - \tilde{\mu}) + 1)}, \quad \text{for } i = Z_b(1), \ldots, Z_b(S),$$

where $\lambda_b = \arg\max_\lambda S^{-1}\sum_{s=1}^S \log\{1 + \lambda^{\mathrm{T}}(X_{Z_b(s)} - \tilde{\mu})\}$ is the Lagrange parameter. Given $\mathcal{D}$ and $\mathcal{U}$, the randomness solely comes from $\{Z_b(s)\}_{s=1}^S$. If we do not introduce the randomness of Z_b, we can obtain a totally optimality-based weighting method for domain adpatation as follows:

$$\begin{aligned} &\max \sum_{i=1}^I \log w_i^* \\ &\text{subject to} \quad \sum_{i=1}^I w_i^* = 1, \quad \sum_{i=1}^I w_i^* X_i = \frac{1}{J}\sum_{j=1}^J \tilde{X}_j, \quad w_i^* > 0, \end{aligned} \tag{3}$$

with $L_* = \sum_{i=1}^I w_i^* l(Y_i, f_\theta(X_i))$ and similarly obtain λ_*. We can obtain Proposition 4.1.

Proposition 4.1. Denote $V = I^{-1}\sum_{i=1}^I X_i X_i^{\mathrm{T}} - (I^{-1}\sum_{i=1}^I X_i)(I^{-1}\sum_{i=1}^I X_i)^{\mathrm{T}}$ and suppose V positive definitive and its minimum eigenvalue is $\nu_1(V)$ and $||X_i - \mu|| \le R_x$, then for the solution of Equation (3), we have

$$\frac{\nu_1(V)}{I(\nu_1(V) + ||\mu - \tilde{\mu}||(R_x + ||\mu - \tilde{\mu}||))} \le w_i^* \le 1,$$

and for the solution of Equation (2) and for $i = Z_b(1), \ldots, Z_b(S)$,

$$\frac{\nu_1(V)}{S(\nu_1(V) + ||\mu - \tilde{\mu}||(R_x + ||\mu - \tilde{\mu}||))} \le w_i^{(b)} \le 1, \quad a.s.$$

Proposition 4.1 guarantees that the weights are not extreme. Compared to density ratio method, the optimization based method is more robust and avoid the high probability problems. We obtain Proposition 4.1 by bounding λ_b and λ_* at the constant order. Traditional empirical literature assumes $\mu = \tilde{\mu}$ and obtain $\lambda_b = O_p(1/S^{1/2})$.

We can decompose L_b into three parts as follows:

$$L_b = \underbrace{L_*}_{\text{Term I}} + \underbrace{\frac{1}{S}\sum_{s=1}^{S}\left\{Iw^*_{Z_b(s)} l_{Z_b(s)} - L_*\right\}}_{\text{Term II: resample error}} + \underbrace{\frac{1}{S}\sum_{s=1}^{S}\left\{l_{Z_b(s)}[Sw^{(b)}_{Z_b(s)} - Iw^*_{Z_b(s)}]\right\}}_{\text{Term III: weight error}}. \tag{4}$$

Theorem 4.2. Under Condition of Proposition 4.1 and assume $|l_i| \le R_l$. for $0 < \alpha \le 1/2$ and when $B \to \infty$, we have that

$$|L_{\lceil (1-\alpha)B \rceil} - L_*| \le \frac{q_{1-\alpha/4}}{\sqrt{S}} \sqrt{I^{-1}\sum_{i=1}^{I}(Iw^* l_i - L_*)^2} + IR_l(R_x + |\mu - \tilde{\mu}|)\sqrt{\frac{2q_{1-\alpha/8}\log S}{\nu_1(V)\sqrt{S}}}(\nu_1(V) + |\mu - \tilde{\mu}|(R_x + |\mu - \tilde{\mu}|)).$$

$$\sqrt{S}\cdot \text{Term II} = \frac{1}{\sqrt{S}}\sum_{s=1}^{S}\left\{Iw^*_{Z_b(s)} l_{Z_b(s)} - L_*\right\} \xrightarrow{d} \mathcal{N}(0, I^{-1}\sum_{i=1}^{I}(Iw^*_i l_i - L_*)^2),$$

and with at least probability $1-\alpha$

$$|\text{Term III}| \le IR_l(R_x + |\mu - \tilde{\mu}|)\sqrt{\frac{2q_{1-\alpha/4}\log S}{\nu_1(V)\sqrt{S}}}(\nu_1 + |\mu - \tilde{\mu}|(R_x + |\mu - \tilde{\mu}|)) \lesssim \frac{\log(S)}{S^{1/4}}.$$

Theorem 4.2 tells us L_b is centered at L_* and with two errors. Term II converges to zero at $O(1/S^{1/2})$ and follows normal distribution while converges to zero at $O(\log(S)/S^{1/4})$. Smaller α decrease the bound in Theorem 4.2.

4.2 Analysis on Adaptation Behavior of Our Methods

Define the α-quantile loss for $\eta' \sim \mathbb{P}_n$: $\text{VaR}^{\alpha}_{\mathbb{P}_n}[L(\theta,\eta')] = \inf\{t : P(L(\theta,\eta') \le t) \ge \alpha, \eta' \sim \mathbb{P}_n\}$. In contrast, define the distribution in which the model has good performance

$$\mathbb{P}^{\alpha}_n(L,\theta) = \{\eta : L(\theta,\eta) \le \text{VaR}^{\alpha}_{\mathbb{P}_n}[L(\theta,\eta')]\}.$$

When we use our algorithms to train the model, the model performance on test distribution is bounded by our optimized objection and total variance distance between our uncertain set and test distribution. This can be concluded as follows.

Theorem 4.3. Assume $\mathcal{Y} \subset [0,1]$, $l(\cdot,\cdot)$ is L_1 loss and there exist label functions such that $Y = U_\eta(X)$ in domain $\mathcal{X}_\eta$ and $\text{TV}(\cdot,\cdot)$ is the total variance between two distributions, then we have that

(i)(Unsupervised domain adaptation method)

$$L(\theta,\eta_T) \le L(\theta,\hat{\eta}_T) + \text{TV}(\mathcal{X}_{\hat{\eta}_T}, \mathcal{X}_T)$$

$$+ \min\{\mathbb{E}_{\mathcal{X}_{\hat{\eta}_T}}[|U_{\hat{\eta}_T} - U_T|], \mathbb{E}_{\mathcal{X}_T}[|U_{\hat{\eta}_T} - U_T|]\},$$

(ii)(Our method)

$$L(\theta, \eta_T) \le \mathrm{VaR}^{\alpha}_{\mathbb{P}_n}[L(\theta, \eta)] + \min_{\eta \in \mathbb{P}^{\alpha}_n(L,\theta)} \Big[\mathrm{TV}(\mathcal{X}_\eta, \mathcal{X}_T) + \min\Big\{ \mathbb{E}_{\mathcal{X}_\eta}[|U_\eta - U_T|], \mathbb{E}_{\mathcal{X}_T}[|U_\eta - U_T|] \Big\} \Big].$$

Remark 4.1. The traditional domain adaptation methods actually estimate the targeted distribution parameter η_T based on the unlabelled target domain information. Denote the estimated parameter as $\hat{\eta}_T$. The first inequality in Theorem 4.3 shows the target loss can be bounded to the optimized loss $L(\theta, \hat{\eta}_T)$, the covariate shift term $\mathrm{TV}(\mathcal{D}_{\hat{\eta}_T}, \mathcal{D}_T)$ and the concept shift term $\min\{\mathbb{E}_{\mathcal{D}_{\hat{\eta}_T}}[|l_{\hat{\eta}_T} - l_T|], \mathbb{E}_{\mathcal{D}_T}[|l_{\hat{\eta}_T} - l_T|]\}$. Traditional domain adaptation methods make the covariate shift term $\mathrm{TV}(\mathcal{D}_{\hat{\eta}_T}, \mathcal{D}_T)$ small. However, the concept shift term is unpredictable.

The second inequality of Theorem 4.3 shows the guarantee of our method paralleled to the first. The minimum can help reduce the label shift. And the last part of Theorem 4.3 compares these two methods. We know our method is at least not worse than the traditional domain adaptation methods. Note that $1/2$-quantile is the median; we always choose $\alpha \ge 1/2$ conservatively to obtain a conservative uncertain ball. We estimate the distribution of η_T as $\mathbb{P}_n$, and we can obtain $L(\theta, \hat{\eta}_T) = \hat{\mathbb{E}} L(\theta, \eta_T)$, therefore, when $\alpha \ge 1/2$, then $L(\theta, \hat{\eta}_T) \le \mathrm{VaR}^{\alpha}_{\mathbb{P}_n}[L(\theta, \eta)]$ asymptotically hold. And we follow the classical domain adaptation approach to assume other conditions.

4.3 Bootstrap and Uncertainty Estimation

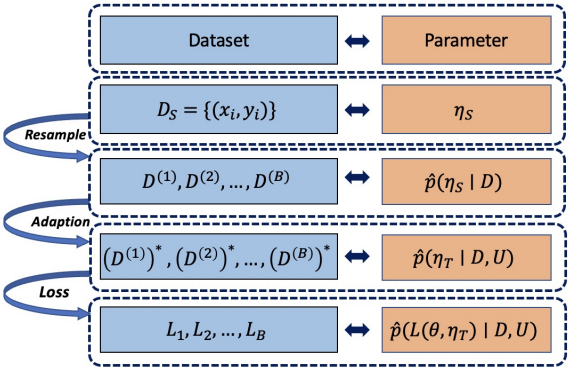

Figure 2: Our method operates the datasets and corresponds to parameter-level operations to quantify the test distribution uncertainty.

As introduced in Section 1, we utilize Bootstrap to construct an uncertainty region from the source domain. Apart from efficiency and simplicity, we study the theoretical connection between bootstrap and uncertainty estimation in this section. To give the intuition, we first analyze the scenario when test data is unavailable. Given a dataset $\mathcal{D}_S = \{(x_i, y_i)\}_{i=1}^n$ sampled from the source distribution, we bootstrap B datasets containing s samples

respectively. Specifically, for each $b = 1, \ldots, B$, we sample $(x_1^{(b)}, y_1^{(b)}), \ldots, (x_s^{(b)}, Y_s^{(b)})$ with replacement from $\mathcal{D}_S$ and construct a sampled dataset $\mathcal{D}^{(b)} = \{(x_1^{(b)}, y_1^{(b)}), \ldots, (x_s^{(b)}, y_s^{(b)})\}$, $b = 1, \ldots, B$. Traditionally, it is preferred to sample datasets of the same size, i.e. $s = n$. We can compute the average loss for each sampled dataset $\mathcal{D}^{(b)}$ as $L_b = s^{-1} d(y_i^{(b)}, f(x_i^{(b)}))$. For each iteration, the α-quantile loss $L_{(\lceil \alpha B \rceil)}$ is computed by ranking $\{L_b\}_{b=1}^B$. In the following proposition, we show bootstrap datasets' α-quantile loss is equivalent as a sample-level variance penalization.

Proposition 4.4. For any $b = 1, \ldots, B$, when $s \to \infty$ and $n^{-1} \sum_{j=1}^n [d(y_j, f(x_j)) - n^{-1} \sum_{i=1}^I d(y_i, f(x_i))]^2 < \infty$, then

$$\begin{aligned} &\sqrt{s}\left(L_b - \frac{1}{n}\sum_{i=1}^{I} d\Big(y_i, f(x_i)\Big)\right) \\ \xrightarrow{d} &\mathcal{N}\left(0, \frac{1}{n}\sum_{j=1}^{n}\Big[d\Big(y_j, f(x_j)\Big) - \frac{1}{n}\sum_{i=1}^{I} d\Big(y_i, f(x_i)\Big)\Big]^2\right). \end{aligned}$$

Furthermore, when $B \to \infty$, for $\alpha \in (0, 1)$,

$$\begin{aligned} L_{(\lceil \alpha B \rceil)} &= \left(\frac{1}{n}\sum_{i=1}^{I} d(y_i, f(x_i))\right) \\ &+ \frac{q_\alpha}{\sqrt{s}}\left(\frac{1}{n}\sum_{i=1}^{I}[d(y_i, f(x_i)) - \frac{1}{n}\sum_{i=1}^{I} d(y_i, f(x_i))]^2\right) + o_{\mathbb{P}}(\frac{1}{\sqrt{s}}), \end{aligned}$$

where q_α is the α quantile of the standard normal distribution.

Remark 4.2.

1. The first argument characterizes the uncertainty among different resampled datasets. With a sufficient number of samples in each dataset, the average loss of each sampled dataset tends to the normal distribution. The mean of bootstrapped loss function L_b is the original source data loss, and the variance of bootstrapped loss function L_b is the s^{-1} times the sample variance of the original dataset. The variance of resampled datasets' losses scale with the sample-level variance of the original dataset, which implies strong association between the uncertainty quantified by bootstrap datasets and the uncertainty of individual samples. The intuition is formally justified by the second argument.

2. As shown in the second argument of the proposition, the α-quantile $L_{(\lceil \alpha B \rceil)}$ of the bootstrap datasets' losses is asymptotically equivalent to the empirical loss regularized by sample-level variances. This exactly recovers the objective of classic variance penalization methods. Since the variance of the model reflects uncertainty of prediction, $L_{(\lceil \alpha B \rceil)}$ encourages the model to deliver consistently confident predictions across samples. Therefore, the quantile α controls the strength of uncertainty regularization, with larger α implying less tolerance towards uncertain predictions. Intuitively, the quantile α characterizes the size of the uncertainty region.

Proposition 4.4 inspires the design of robust learning methods under data uncertainty. Although the underlying distribution parameter $p(\eta_S)$ is unknown, we can utilize the bootstrap datasets' distribution as a surrogate for the distribution of $p(\eta_S)$. As a result, we can characterize data uncertainty in the original distribution by operating on the bootstrap

datasets without explicit access to $p(\eta_S)$. Therefore, we do not need assume the parameter model to update the distribution.

Furthermore, we proceed to a more complex setting where data are collected from multiple distributions. As a result, the training set contains several source domains. In this out-of-distribution generalization scenario, we can adopt a similar idea of bootstrap to obtain a robust learning method. We prove the resulting approach is asymptotically equivalent to VRex, a variance penalization method for domain generalization, and we verify these results in the Supplementary.

In the following, we take the perspective of Distributionally Robust Optimization and show a similar philosophy with our method. With fixed θ and $\eta \sim \mathbb{P}_n$, we have a distribution of $L(\theta, \eta)$. Then we denote the α-quantile of the distribution by $\mathrm{VaR}^{\alpha}_{\mathbb{P}_n} L(\theta, \eta)$. The optimization for $\mathrm{VaR}^{\alpha}_{\mathbb{P}_n} L(\theta, \eta)$ can also be understood as a combination of average losses and the variance penalty. From a Bayesian perspective, Figure 2 shows the relationship between our algorithm and the parameter uncertainty estimation, and we borrow the following theorem.

5 Experiments

In this section, we conduct experiments on diverse datasets to compare the proposed method with baseline methods and evaluate its performance improvement under distribution shifts.

5.1 Setup

5.1.1 Datasets

We conduct experiments on four real-world tabular datasets and two image datasets, performing binary classification tasks across all of them. The tabular datasets include ACS Income, ACS Mobility, and ACS Public Coverage from the nationwide ACS PUMS data [Ding et al., 2021], as well as the Taxi datasets [2]. For image data, we use PACS [Li et al., 2017] and ColoredMNIST [Arjovsky et al., 2020]. These datasets naturally contain distinct internal domains with inherent gaps. Following the criterion of DomainBed [Gulrajani and Lopez-Paz, 2020], we split each dataset's training domain into training and validation sets at a 4:1 ratio and evaluate performance on the test domain. Detailed descriptions of the datasets and tasks are provided in the Appendix.

5.1.2 Baselines

To evaluate the performance of our algorithm , we selected the following three types of baselines:

- Empirical Risk Minimization (ERM) [Vapnik et al., 1998].
- Unsupervised Domain Adaptation (UDA) algorithms that utilize test data information, including DANN [Ganin et al., 2016b], CORAL [Sun and Saenko, 2016b], Pseudo-Labeling [Lee et al., 2013], Noisy Student [Xie et al., 2020], and C-Mixup+SpAR [Eyre et al., 2023]. These algorithms excel in covariate shift cases.

[2] https://www.kaggle.com/competitions/nyc-taxi-trip-duration/data, https://www.kaggle.com/datasets/mnavas/taxi-routes-for-mexico-city-and-quito

- Distributionary Robust Optimization (DRO) algorithms that does not use test data information, including MMD [Li et al., 2018] and χ^2-dro [Duchi and Namkoong, 2020]. These algorithm are effective in concept shift cases.

5.2 Main Results

	top 1	top 2	top 3	MR
ERM	2	6	**11**	3
χ^2-dro	0	0	0	5
MMD	0	6	9	3
CORAL	2	4	8	3
DANN	0	0	0	6
Pseudo Label	0	0	0	8
C-Mixup+SpAR	0	0	0	7
Noisy Student	0	0	0	9
Ours	**8**	**8**	9	**1**

Table 2: Traversal experiments on Taxi dataset. Taxi dataset contains 4 states. we trained on one state and tested on another, totaling 12 experiments.

	P	A	C	S
ERM	**94.33 ± 1.94**	**71.08 ± 11.64**	68.50 ± 14.74	63.88 ± 16.32
χ^2-dro	74.31 ± 23.71	59.51 ± 24.81	63.28 ± 19.76	62.89 ± 20.51
MMD	66.73 ± 38.64	61.17 ± 32.19	77.82 ± 3.43	76.21 ± 2.22
CORAL	64.75 ± 24.09	58.02 ± 26.72	62.27 ± 23.08	56.16 ± 23.24
DANN	61.14 ± 22.82	48.55 ± 18.64	39.65 ± 20.77	28.22 ± 8.24
Pseudo Label	30.28 ± 6.79	23.80 ± 2.60	16.91 ± 8.11	19.22 ± 7.58
C-Mixup+SpAR	68.08 ± 33.89	59.00 ± 27.76	55.46 ± 28.72	51.44 ± 22.99
Noisy Student	39.70 ± 10.70	12.94 ± 3.95	16.87 ± 0.38	19.47 ± 0.90
Ours	92.67 ± 1.25	63.92 ± 27.47	**79.76 ± 0.92**	**76.24 ± 2.62**

Table 3: Experiments on PACS dataset. PACS consists of four domains: Photos (P), Art painting (A), Cartoon (C), and Sketch (S). We train the model on three of these domains and test it on the remaining one. The table reports the average accuracy and standard deviation for each test domain.

	0.1 → 0.2	0.2 → 0.1	0.2 → 0.9	0.9 → 0.2	0.1 → 0.9	0.9 → 0.1
ERM	79.75 ± 0.24	89.31 ± 0.26	10.07 ± 0.15	20.09 ± 0.47	9.88 ± 0.11	12.42 ± 2.04
χ^2-dro	79.90 ± 0.12	89.36 ± 0.60	10.06 ± 0.16	20.06 ± 0.27	10.07 ± 0.09	12.82 ± 2.38
MMD	80.05 ± 0.03	90.09 ± 0.20	10.15 ± 0.01	19.95 ± 0.11	10.09 ± 0.09	**14.81 ± 1.58**
CORAL	80.18 ± 0.39	89.28 ± 0.38	10.17 ± 0.20	20.16 ± 0.21	9.96 ± 0.17	11.97 ± 2.09
DANN	79.85 ± 0.30	89.21 ± 1.05	10.12 ± 0.09	20.20 ± 0.27	9.84 ± 0.09	10.35 ± 0.54
Pseudo Label	71.36 ± 12.26	**90.17 ± 0.19**	9.99 ± 0.11	20.08 ± 0.17	9.99 ± 0.08	12.39 ± 1.69
C-Mixup+SpAR	80.27 ± 0.14	89.92 ± 0.27	9.89 ± 0.09	19.76 ± 0.03	10.01 ± 0.15	10.52 ± 0.37
Noisy Student	79.85 ± 0.20	89.81 ± 0.08	10.18 ± 0.11	19.99 ± 0.12	10.22 ± 0.41	10.02 ± 0.17
Ours	**80.34 ± 0.12**	90.04 ± 0.07	**10.80 ± 0.43**	**20.32 ± 0.17**	**10.48 ± 0.14**	10.36 ± 0.08

Table 4: Experiments on ColoredMNIST. In the column labeled 'A → B', A and B represent the probability of color flipping in the training and test sets, respectively. A larger difference between A and B indicates a stronger concept shift.

We conducted large-scale traversal experiments on the tabular datasets. For each ACS dataset, we evaluated all possible domain pairs, resulting in a total of 2,550 (51×50) domain pairs. This ensures a comprehensive and fair evaluation of our algorithm's effectiveness in addressing uncertainty in test distributions.

As shown in Table 5, our approach ranks first across all three ACS datasets in both top-1 and top-2 positions. Additionally, the quartiles of our rankings are consistently better than those of the baseline methods. Notably, our method achieves state-of-the-art (SOTA) performance on more than 50% of the domain pairs in both ACS Pubcov and ACS Income,

	ACS Pubcov					ACS Income					ACS Mobility				
	top 1	top 2	top 3	QR	MR	top 1	top 2	top 3	QR	MR	top 1	top 2	top 3	QR	MR
ERM	285	1064	**1803**	2	3	312	951	1604	2	3	282	655	**1040**	**2**	**4**
χ^2-dro	71	119	193	6	7	19	38	48	7	8	276	590	838	3	5
MMD	330	1110	1794	2	3	351	974	1597	2	3	255	633	1012	3	**4**
CORAL	195	459	739	3	5	258	737	1146	2	4	213	585	945	3	**4**
DANN	31	65	104	6	7	35	80	135	6	6	280	602	947	3	5
Pseudo Label	24	65	123	6	7	6	18	41	6	7	263	617	984	3	**4**
C-Mixup+SpAR	287	687	1192	2	4	271	700	1224	2	4	160	404	698		
Ours	**1284**	**1476**	1630	**1**	**1**	**1284**	**1584**	**1836**	**1**	**1**	**1707**	**258**	172		

Table 5: Traversal experiments on three ACS tabular datasets. Each dataset includes data from 51 states of the United States. For each dataset, we conducted 2550 experiments by training on one state and testing on another. We report the number of experiments where each method ranked first, in the top two, and in the top three, as well as the 25% rank (QR) and 50% rank (MR) among the 2550 experiments for each dataset.

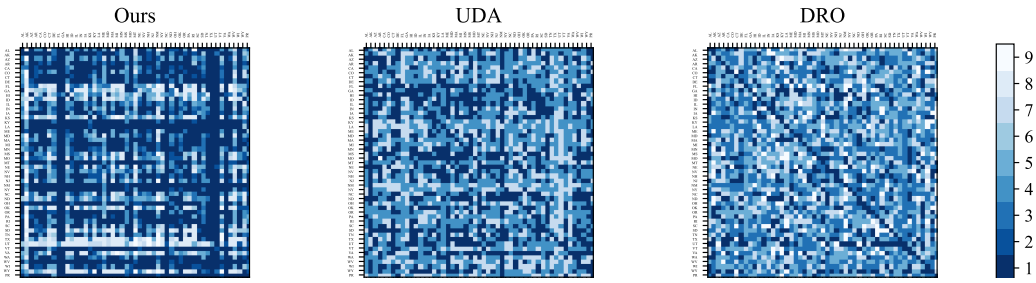

Figure 3: Traversal experiments on ACS Pubcov dataset. The figure presents the rankings of our algorithm compared to baselines across all 51×50 domain pairs. In this visualization, darker colors represent higher rankings. For the two categories of baselines—UDA and DRO—we display the rankings of the top-performing algorithms within each class for each domain pair.

whereas the second-best algorithm achieves SOTA on less than 15% of the pairs. We also outperform the baselines on the smaller tabular dataset, Taxi, as indicated in Table 2.

Figure 3 shows the rankings of our method compared to two categories of baselines—UDA and DRO—across all 2,550 domain pairs in the ACS Pubcov dataset. Our method consistently ranks highly and clearly outperforms both UDA and DRO, demonstrating its ability to adapt to both concept shift and covariate shift. In a few cases, our method slightly underperforms UDA, likely due to an strong covariate shift, during which the DRO method also performs poorly.

5.3 Image Datasets

Experiments on the image datasets are consistent with the main results, confirming the strong performance of our algorithm. As shown in Table 3, our method consistently ranks within the top two across all four test domains of the PACS dataset. On the ColoredMNIST

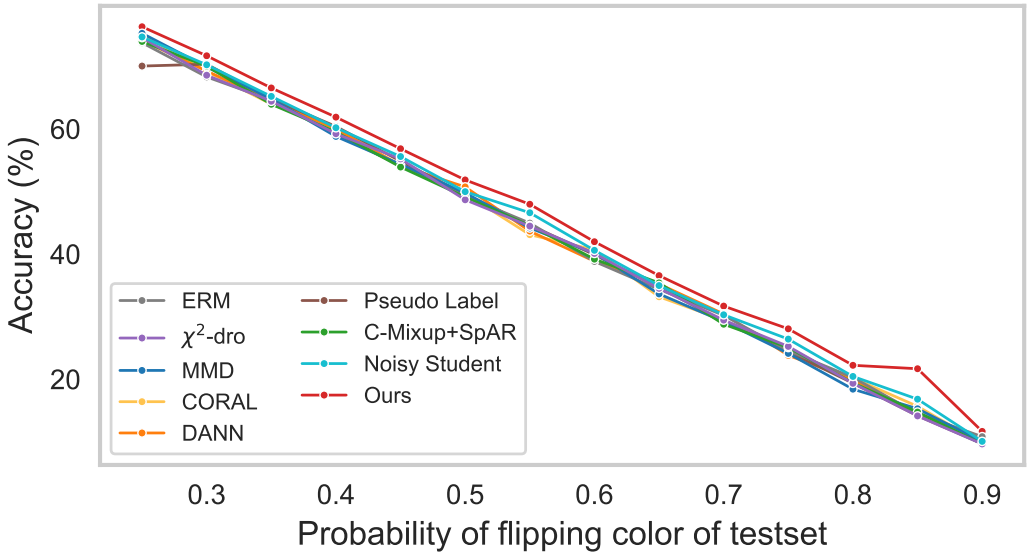

Figure 4: Concept shift of ColoredMNIST. The training set is created by merging datasets with color flipping probabilities of 0.1 and 0.2. The difference in color flipping probabilities between the training and test sets is used to approximate the strength of concept shift. In the figure, a larger horizontal coordinate indicates a stronger concept shift.

dataset, where concept shifts are manually introduced, our method ranks first in most cases, with its advantage becoming more significant under larger concept shifts, as shown in Table 4.

5.4 Concept Shift Analysis

We verified the robustness of our method to concept shift using the ColoredMNIST dataset, as illustrated in Figure 4. As the concept shift increases, the performance rankings of the UDA and DRO methods fluctuate, while our method consistently remains at the SOTA level. This demonstrates that ATDU can effectively handle a certain degree of concept shift and maintains robustness even under more significant concept shifts.

6 Conclusion

Our investigation into the uncertainty of target domain distribution within unsupervised domain adaptation (UDA) problems, coupled with our proposed Against Test Data Uncertainty (ATDU) method, underscores significant advancements in the field. By formulating and addressing this uncertainty from a Bayesian perspective, ATDU offers a robust framework for domain adaptation.

Our method stands out for its ability to leverage source domain data to quantify uncertainty and guide adaptation to the target domain, even in scenarios where only

unlabeled data from the target domain is available. Through bootstrap resampling and optimization of the α quantile loss, ATDU demonstrates superior performance in managing domain shifts and enhancing prediction accuracy in the target domain. Theoretical analyses highlight the robustness and efficacy of ATDU, confirming its ability to handle uncertainty inherent in domain adaptation scenarios. Furthermore, extensive experimental evaluations across diverse datasets validate the superiority of ATDU, showcasing its adaptability and reliability.

In conclusion, our method significantly advances the field of domain adaptation, offering a powerful solution for addressing label shifts and sampling bias in out-of-distribution (OOD) problems. ATDU represents a promising avenue for future research and practical applications in machine learning and beyond.

References

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization, 2020.

Dimitris Bertsimas, Melvyn Sim, and Meilin Zhang. Adaptive distributionally robust optimization. *Management Science*, 65(2):604–618, 2019.

Jiahua Chen, Asokan Mulayath Variyath, and Bovas Abraham. Adjusted empirical likelihood and its properties. *Journal of Computational and Graphical Statistics*, 17:426 – 443, 2008a. URL https://api.semanticscholar.org/CorpusID:39301928.

Jiahua Chen, Asokan Mulayath Variyath, and Bovas Abraham. Adjusted Empirical Likelihood and its Properties. *Journal of Computational and Graphical Statistics*, 17 (2):426–443, June 2008b. ISSN 1061-8600, 1537-2715. doi: 10.1198/106186008X321068. URL http://www.tandfonline.com/doi/abs/10.1198/106186008X321068.

Qingchao Chen, Yang Liu, Zhaowen Wang, Ian Wassell, and Kevin Chetty. Re-weighted adversarial adaptation network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7976–7985, 2018.

Victor Chernozhukov, Denis Chetverikov, Kengo Kato, and Yuta Koike. High-dimensional data bootstrap. *Annual Review of Statistics and Its Application*, 10(1):427–449, 2023.

Frances Ding, Moritz Hardt, John Miller, and Ludwig Schmidt. Retiring adult: New datasets for fair machine learning. *Advances in neural information processing systems*, 34:6478–6490, 2021.

Yongjie Du, Deyun Zhou, Yu Xie, Yu Lei, and Jiao Shi. Prototype-guided feature learning for unsupervised domain adaptation. *Pattern Recognition*, 135:109154, 2023.

John Duchi and Hongseok Namkoong. Learning models with uniform performance via distributionally robust optimization, 2020. URL https://arxiv.org/abs/1810.08750.

Bradley Efron and Robert J Tibshirani. *An introduction to the bootstrap*. Chapman and Hall/CRC, 1994.

Benjamin Eyre, Elliot Creager, David Madras, Vardan Papyan, and Richard Zemel. Out of the ordinary: Spectrally adapting regression for covariate shift, 2023. URL https://arxiv.org/abs/2312.17463.

Charlie Frogner, Sebastian Claici, Edward Chien, and Justin Solomon. Incorporating Unlabeled Data into Distributionally Robust Learning, December 2019. URL http://arxiv.org/abs/1912.07729. arXiv:1912.07729 [cs, stat].

Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *International conference on machine learning*, pages 1180–1189. PMLR, 2015.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor S. Lempitsky. Domain-adversarial training of neural networks. *J. Mach. Learn. Res.*, 17:59:1–59:35, 2016a. URL http://jmlr.org/papers/v17/15-239.html.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks, 2016b. URL https://arxiv.org/abs/1505.07818.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization, 2020.

Jing Jiang and ChengXiang Zhai. Instance weighting for domain adaptation in NLP. In John Carroll, Antal van den Bosch, and Annie Zaenen, editors, *ACL 2007, Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, June 23-30, 2007, Prague, Czech Republic*. The Association for Computational Linguistics, 2007. URL https://aclanthology.org/P07-1034/.

Pang Wei Koh, Shiori Sagawa, Henrik Marklund, Sang Michael Xie, Marvin Zhang, Akshay Balsubramani, Weihua Hu, Michihiro Yasunaga, Richard Lanas Phillips, Irena Gao, et al. Wilds: A benchmark of in-the-wild distribution shifts. In *International conference on machine learning*, pages 5637–5664. PMLR, 2021.

Dong-Hyun Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896. Atlanta, 2013.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Deeper, broader and artier domain generalization, 2017. URL https://arxiv.org/abs/1710.03077.

Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C Kot. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5400–5409, 2018.

Yundong Li, Longxia Guo, and Yizheng Ge. Pseudo labels for unsupervised domain adaptation: A review. *Electronics*, 12(15):3325, 2023.

Jian Liang, Ran He, Zhenan Sun, and Tieniu Tan. Exploring uncertainty in pseudo-label guided unsupervised domain adaptation. *Pattern Recognition*, 96:106996, 2019.

Jiashuo Liu, Jiayun Wu, Bo Li, and Peng Cui. Distributionally robust optimization with data geometry. *Advances in neural information processing systems*, 35:33689–33701, 2022.

Jiashuo Liu, Tianyu Wang, Peng Cui, and Hongseok Namkoong. On the need for a language describing distribution shifts: Illustrations on tabular datasets, 2023a.

Jiashuo Liu, Jiayun Wu, Tianyu Wang, Hao Zou, Bo Li, and Peng Cui. Geometry-calibrated dro: Combating over-pessimism with free energy implications. *arXiv preprint arXiv:2311.05054*, 2023b.

Jiashuo Liu, Tianyu Wang, Peng Cui, and Hongseok Namkoong. On the need for a language describing distribution shifts: Illustrations on tabular datasets. *Advances in Neural Information Processing Systems*, 36, 2024.

Shikun Liu, Tianchun Li, Yongbin Feng, Nhan Tran, Han Zhao, Qiang Qiu, and Pan Li. Structural re-weighting improves graph domain adaptation. In *International Conference on Machine Learning*, pages 21778–21793. PMLR, 2023c.

Yukun Liu. Biased-sample empirical likelihood weighting for missing data problems: an alternative to inverse probability weighting. *Statistical Methodology*, 85(1), 2023.

Saeid Motiian, Marco Piccirilli, Donald A. Adjeroh, and Gianfranco Doretto. Unified deep supervised domain adaptation and generalization. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 5716–5726. IEEE Computer Society, 2017. doi: 10.1109/ICCV.2017.609. URL https://doi.org/10.1109/ICCV.2017.609.

Art B Owen. *Empirical likelihood*. Chapman and Hall/CRC, 2001.

Poojan Oza, Vishwanath A Sindagi, Vibashan Vishnukumar Sharmini, and Vishal M Patel. Unsupervised domain adaptation of object detectors: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

Hongxiang Qiu, Edgar Dobriban, and Eric Tchetgen Tchetgen. Prediction sets adaptive to unknown covariate shift. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 85(5):1680–1705, 2023.

Hamed Rahimian and Sanjay Mehrotra. Distributionally robust optimization: A review. *arXiv preprint arXiv:1908.05659*, 2019.

Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3723–3732, 2018.

Alexander Shapiro, Enlu Zhou, and Yifan Lin. Bayesian Distributionally Robust Optimization, February 2023. URL http://arxiv.org/abs/2112.08625. arXiv:2112.08625 [math].

Masashi Sugiyama, Matthias Krauledat, and Klaus-Robert Müller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8 (5), 2007.

Baochen Sun and Kate Saenko. Deep CORAL: correlation alignment for deep domain adaptation. In Gang Hua and Hervé Jégou, editors, *Computer Vision - ECCV 2016 Workshops - Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III*, volume 9915 of *Lecture Notes in Computer Science*, pages 443–450, 2016a. doi: 10.1007/978-3-319-49409-8_35. URL https://doi.org/10.1007/978-3-319-49409-8_35.

Baochen Sun and Kate Saenko. Deep coral: Correlation alignment for deep domain adaptation. In *Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III 14*, pages 443–450. Springer, 2016b.

Yaniv Taigman, Adam Polyak, and Lior Wolf. Unsupervised cross-domain image generation. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017. URL https://openreview.net/forum?id=Sk2Im59ex.

Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, and Ludwig Schmidt. Measuring robustness to natural distribution shifts in image classification. *Advances in Neural Information Processing Systems*, 33:18583–18599, 2020.

Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017, Honolulu, HI, USA, July 21-26, 2017*, pages 2962–2971. IEEE Computer Society, 2017. doi: 10.1109/CVPR.2017.316. URL https://doi.org/10.1109/CVPR.2017.316.

Vladimir Naumovich Vapnik, Vlamimir Vapnik, et al. Statistical learning theory. 1998.

Vivek Warke, Satish Kumar, Arunkumar Bongale, and Ketan Kotecha. Robust tool wear prediction using multi-sensor fusion and time-domain features for the milling process using instance-based domain adaptation. *Knowledge-Based Systems*, 288:111454, 2024.

Olivia Wiles, Sven Gowal, Florian Stimberg, Sylvestre Alvise-Rebuffi, Ira Ktena, Krishnamurthy Dvijotham, and Taylan Cemgil. A fine-grained analysis on distribution shift. *arXiv preprint arXiv:2110.11328*, 2021.

Di Wu, Helin Zhu, and Enlu Zhou. A Bayesian Risk Approach to Data-driven Stochastic Optimization: Formulations and Asymptotics. *SIAM Journal on Optimization*, 28(2): 1588–1612, January 2018. ISSN 1052-6234, 1095-7189. doi: 10.1137/16M1101933. URL https://epubs.siam.org/doi/10.1137/16M1101933.

Junkang Wu, Jiawei Chen, Jiancan Wu, Wentao Shi, Xiang Wang, and Xiangnan He. Understanding contrastive learning via distributionally robust optimization. *Advances in Neural Information Processing Systems*, 36, 2024.

Qizhe Xie, Minh-Thang Luong, Eduard Hovy, and Quoc V. Le. Self-training with noisy student improves imagenet classification, 2020. URL https://arxiv.org/abs/1911.04252.

Han Zhao, Remi Tachet des Combes, Kun Zhang, and Geoffrey J. Gordon. On learning invariant representations for domain adaptation. In Kamalika Chaudhuri and Ruslan Salakhutdinov, editors, *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 7523–7532. PMLR, 2019. URL http://proceedings.mlr.press/v97/zhao19a.html.

Enlu Zhou and Wei Xie. Simulation optimization when facing input uncertainty. In *2015 Winter Simulation Conference (WSC)*, pages 3714–3724, Huntington Beach, CA, USA, December 2015. IEEE. ISBN 978-1-4673-9743-8. doi: 10.1109/WSC.2015.7408529. URL http://ieeexplore.ieee.org/document/7408529/.

Supplementary Material

Section A studies the bootstrap method with multi-source data. Section B provides the proof of the main results. Section C introduces the datasets and tasks used in our experiments. Section D describes the algorithms used to obtain weights $\{p_i^{(b)}\}_{i=1}^s$. Section E analyzes the increased complexity of ATDU compared to ERM.

A Multi-source Bootstrap

For Domain $\mathcal{D}_1, \ldots, \mathcal{D}_n$, we can compute the loss of each domain $L_1, \ldots, L_n$. Similarly, we can sample $\mathcal{D}_1^{(b)}, \mathcal{D}_2^{(b)}, \ldots, \mathcal{D}_s^{(b)}$ with replacement and denote $D^{(b)} = \mathcal{D}_1^{(b)} \cup \mathcal{D}_2^{(b)} \cup \ldots \cup \mathcal{D}_s^{(b)}$, For each batch, we can compute $\text{Loss}(D^{(b)})$. For each iteration, rank $\{\text{Loss}(D^{(b)})\}$ and choose $\text{Loss}(D^{(b)})_{(\lceil \alpha B \rceil)}$.

Table 6: Multi-source Bootstrap result on different data of RotatedMIST. The training sets rotate the images by 0 and 75 degrees.

Method	15°	30°	45°	60°	Overall	Mean	Var	Max	Min
ERM	0.937345	0.853004	0.909917	0.976599	0.919215	0.919216	0.002023	0.976599	0.853004
VREx	0.95243	0.879575	0.923716	0.976684	0.9331	0.933101	0.001307	0.976684	0.879575
Ours	0.958601	0.893374	0.918917	0.972827	0.935929	0.93593	0.000994	0.972827	0.893374

Proposition A.1. For any $b = 1, \ldots, B$,

$$\mathbb{E}\text{Loss}(D^{(b)}) = \frac{1}{n}\sum_{i=1}^{I} L_i,$$

$$\text{var}(L_b) = \frac{1}{sn}\sum_{j=1}^{n}[L_j - \frac{1}{n}\sum_{i=1}^{I} L_i]^2$$

Therefore, for large B,

$$\text{Loss}_{(\lceil \alpha B \rceil)} = \left(\frac{1}{n}\sum_{i=1}^{I} L_i\right) + \frac{q_{\alpha/2}}{\sqrt{s}}\left(\sum_{j=1}^{n}[L_j - \frac{1}{n}\sum_{i=1}^{I} L_i]^2\right) + o(\frac{1}{\sqrt{n}}).$$

B Proof of the Result

B.1 Proof of Proposition 4.1 and Theorem 4.2

Define $\tilde{\mu}$ Define $G(\lambda, i) = \log|1 + \lambda^{\mathrm{T}}(X_i - \tilde{\mu})|$ and $g(\lambda, i) = (X_i - \tilde{\mu})/\{1 + \lambda^{\mathrm{T}}(X_i - \tilde{\mu})\}$. Then

$$w_i^{(b)} = S^{-1}[1 - \lambda_b^{\mathrm{T}} g(\lambda_b, i)]$$

Recall $L_b = \sum_{s=1}^{S} w_{Z_b(s)}^{(b)} l(f_\theta(X_{Z_b(s)}, Y_{Z_b(s)}))$, for convenience, write $l_i = l(f_\theta(X_i, Y_i))$, then

$$L_b = \sum_{s=1}^{S} w_{Z_b(s)}^{(b)} l_{Z_b(s)} = \frac{1}{S}\sum_{i=1}^{I} N_{bi} l_i [1 - \lambda_b^{\mathrm{T}} g(\lambda_b, i)].$$

Define

$$\lambda_* = \arg\max \frac{1}{I}\sum_{i=1}^{I} G(\lambda, i).$$

Then

$$L_b = \frac{1}{S}\sum_{i=1}^{I} l_i N_{bi}[1 - \lambda_*^{\mathrm{T}} g(\lambda_*, i)] + \frac{1}{S}\sum_{i=1}^{I} l_i N_{bi}[\lambda_*^T g(\lambda_*, i) - \lambda_b^T g(\lambda_b, i)]. \tag{5}$$

We can prove Define $L_* = I^{-1}\sum_{i=1}^{I} l_i[1 - \lambda_* g(\lambda_*, i)]$ and $\sigma^2 = I^{-1}\sum_{i=1}^{I}(l_i[1 - \lambda_* g(\lambda_*, i)] - L_*)^2$, if $\sigma^2 < \infty$, then

$$\frac{\sqrt{S}(L_b - L_*)}{\sigma} \xrightarrow{d} \mathcal{N}(0, 1).$$

Lemma B.1. For the solution of Equation 2, let $\mu = \sum_{i=1}^{I} X_i/I$ and $V = I^{-1}\sum_{i=1}^{I} X_i X_i^{\mathrm{T}} - (I^{-1}\sum_{i=1}^{I} X_i)(I^{-1}\sum_{i=1}^{I} X_i)^{\mathrm{T}}$ and suppose V positive definitive and its minimum eigenvalue is $\nu_1(V)$, then

$$||\lambda_*|| \le ||\mu - \tilde{\mu}||/\nu_1(V)$$

Furthermore, If $\sum_{i=1}^{I} ||X_i||^2/I < \infty$, we have

$$||\lambda_b|| \le ||\mu - \tilde{\mu}||/\nu_1(V), a.s.$$

If $\mu = \tilde{\mu}$ Traditional empirical likelihood assumes $\mu = \tilde{\mu}$, and they obtain $||\lambda_b|| = O_p(S^{-1/2})$ in traditional empirical likelihood literature.

Proof. Let $\rho = ||\lambda||$ and denote $\xi = \lambda/\rho$. Then

$$\sum_{s=1}^{S} \frac{X_{Z_b(s)} - \tilde{\mu}}{1 + \rho\xi^{\mathrm{T}}(X_i - \tilde{\mu})} = 0.$$

Multiplying ξ^{T} and because $x/(1+x) = x - x^2/(1+x)$,

$$\sum_{s=1}^{S} \xi^{\mathrm{T}}(X_{Z_b(s)} - \tilde{\mu}) = \rho\sum_{s=1}^{S} \frac{\{\xi^{\mathrm{T}}(X_{Z_b(s)} - \tilde{\mu})\}^2}{1 + \rho\xi^{\mathrm{T}}(X_{Z_b(s)} - \tilde{\mu})} = S\rho\sum_{s=1}^{S}\{\xi^{\mathrm{T}}(X_{Z_b(s)} - \tilde{\mu})\}^2 w_{Z_b(s)}^{(b)} \ge 0. \tag{6}$$

Therefore, from Equation (6),

$$\sum_{s=1}^{S} \xi^{\mathrm{T}}(X_{Z_b(s)} - \tilde{\mu}) = \rho\sum_{s=1}^{S} \frac{\{\xi^{\mathrm{T}}(X_{Z_b(s)} - \tilde{\mu})\}^2}{1 + \rho\xi^{\mathrm{T}}(X_{Z_b(s)} - \tilde{\mu})} \ge \rho\frac{\{\sum_{s=1}^{S}\xi^{\mathrm{T}}(X_{Z_b(s)} - \tilde{\mu})\}^2}{S + \rho\sum_{s=1}^{S}\xi^{\mathrm{T}}(X_{Z_b(s)} - \tilde{\mu})}.$$

Therefore,

$$\begin{aligned}\frac{1}{S}\sum_{s=1}^{S} \xi^{\mathrm{T}}(X_{Z_b(s)} - \tilde{\mu}) \ge& \rho([\frac{1}{S}\sum_{s=1}^{S}\{\xi^{\mathrm{T}}(X_{Z_b(s)} - \tilde{\mu})\}^2] - [\frac{1}{S}\sum_{s=1}^{S}\xi^{\mathrm{T}}(X_{Z_b(s)} - \tilde{\mu})]^2) \\ \ge& \rho([\frac{1}{S}\sum_{s=1}^{S}\{\xi^{\mathrm{T}}(X_{Z_b(s)} - \tilde{\mu}) - \frac{1}{S}\sum_{s=1}^{S}\xi^{\mathrm{T}}(X_{Z_b(s)} - \tilde{\mu})\}^2])\end{aligned}$$

$$
\begin{aligned}
&= \rho([\frac{1}{S}\sum_{s=1}^{S}\{\xi^{\mathrm{T}}X_{Z_b(s)} - \frac{1}{S}\sum_{s=1}^{S}\xi^{\mathrm{T}}X_{Z_b(s)}\}^2]) \\
&= \rho([\frac{1}{S}\sum_{s=1}^{S}\{\xi^{\mathrm{T}}X_{Z_b(s)}\}^2] - [\frac{1}{S}\sum_{s=1}^{S}\xi^{\mathrm{T}}X_{Z_b(s)}]^2) \\
&\xrightarrow{a.s.} \rho([\frac{1}{I}\sum_{i=1}^{I}\{\xi^{\mathrm{T}}X_i\}^2] - [\frac{1}{I}\sum_{i=1}^{I}\xi^{\mathrm{T}}X_i]^2) \\
&= \rho\xi^{\mathrm{T}}V\xi \\
&\geq \rho\nu_1(V).
\end{aligned}
$$

On the other hand

$$
\frac{1}{S}\sum_{s=1}^{S}\xi^{\mathrm{T}}(X_{Z_b(s)} - \tilde{\mu}) \xrightarrow{a.s.} \xi^{\mathrm{T}}(\mu - \tilde{\mu}) \leq ||\mu - \tilde{\mu}||.
$$

Therefore, $\rho \leq ||\mu - \tilde{\mu}||/\nu_1(V), \quad a.s.$ □

Now we delve into

$$
\frac{1}{S}\sum_{i=1}^{I} l_i N_{bi}[\lambda_*^T g(\lambda_*, i) - \lambda_b^T g(\lambda_b, i)] = \frac{1}{S}\sum_{s=1}^{S}(\sum_{i=1}^{I} 1(Z_b(s) = i) l_i [\lambda_*^T g(\lambda_*, i) - \lambda_b^T g(\lambda_b, i)]).
$$

If we assume $|X_i - \mu| \leq R_x$, then we have

$$
\begin{aligned}
\lambda_*^T g(\lambda_*, i) - \lambda_b^T g(\lambda_b, i) &= \frac{\lambda_*^{\mathrm{T}}(X_i - \tilde{\mu})}{1 + \lambda_*^{\mathrm{T}}(X_i - \tilde{\mu})} - \frac{\lambda_b^{\mathrm{T}}(X_i - \tilde{\mu})}{1 + \lambda_b^{\mathrm{T}}(X_i - \tilde{\mu})} \\
&= \frac{(\lambda_*^{\mathrm{T}} - \lambda_b^{\mathrm{T}})(X_i - \tilde{\mu})}{(1 + \lambda_*^{\mathrm{T}}(X_i - \tilde{\mu}))(1 + \lambda_b^{\mathrm{T}}(X_i - \tilde{\mu}))} \\
&= SI w_i^{(b)} w_i^* (\lambda_* - \lambda_b)^{\mathrm{T}}(X_i - \tilde{\mu})
\end{aligned}
$$

Therefore, if we assume $l_i \leq R_l$, then

$$
\begin{aligned}
&|\frac{1}{S}\sum_{s=1}^{S}(\sum_{i=1}^{I} 1(Z_b(s) = i) l_i [\lambda_*^T g(\lambda_*, i) - \lambda_b^T g(\lambda_b, i)])| \\
=&|\frac{1}{S}\sum_{s=1}^{S}(\lambda_* - \lambda_b)^{\mathrm{T}}(\sum_{i=1}^{I} 1(Z_b(s) = i) l_i SI w_i^{(b)} w_i^* (\lambda_* - \lambda_b)^{\mathrm{T}}(X_i - \tilde{\mu}))| \\
=&|I\sum_{s=1}^{S} l_{Z_b(s)} w_{Z_b(s)}^{(b)} w_{Z_b(s)}^* (\lambda_* - \lambda_b)^{\mathrm{T}}(X_{Z_b(s)} - \tilde{\mu}))| \\
\leq& IR_l(R_x + |\mu - \tilde{\mu}|)|\sum_{s=1}^{S} w_{Z_b(s)}^{(b)}||\lambda_* - \lambda_b| \\
=& IR_l(R_x + |\mu - \tilde{\mu}|)|\lambda_* - \lambda_b|.
\end{aligned}
$$

At last, we deal with $|\lambda_* - \lambda_b|$. Denote $\mathcal{G}_*(\lambda) = I^{-1}\sum_{i=1}^{I} G(\lambda, i)$, First, we prove the strong convexity of $\mathcal{G}_*(\lambda)$ in $[\min\{1/I, 1/S\}, ||\mu - \tilde{\mu}||/\nu_1(V)]$. Denote $\mathcal{G}_b(\lambda) = S^{-1}\sum_{s=1}^{S} G(\lambda, Z_b(s))$. Then we have

$$\frac{\sqrt{S}(\mathcal{G}_b(\lambda) - \mathcal{G}_*(\lambda))}{\sigma_\lambda} \xrightarrow{d} \mathcal{N}(0,1),$$

where $\sigma_\lambda^2 = I^{-1}\sum_{i=1}^{I}[G_*(\lambda, i) - \mathcal{G}_*(\lambda, i)]^2$.

First, we prove the strong convexity of $\mathcal{G}_*$. Denote $R_\lambda = ||\mu - \tilde{\mu}||/\nu_1(V)$, then we have

$$\begin{aligned}
&||(\lambda_1 - \lambda_2)^{\mathrm{T}}\{\partial_\lambda \mathcal{G}_*(\lambda_1) - \partial_\lambda \mathcal{G}_b(\lambda_2)\}|| \\
=&||\frac{1}{I}\sum_{i=1}^{I}(\lambda_1 - \lambda_2)^{\mathrm{T}}\{\frac{X_i - \tilde{\mu}}{1 + \lambda_1^{\mathrm{T}}(X_i - \tilde{\mu})} - \frac{X_i - \tilde{\mu}}{1 + \lambda_2^{\mathrm{T}}(X_i - \tilde{\mu})}\}|| \\
=&||(\lambda_1 - \lambda_2)^{\mathrm{T}}\frac{1}{I}\sum_{i=1}^{I}\{\frac{(X_i - \tilde{\mu})(X_i - \tilde{\mu})^{\mathrm{T}}}{\{1 + \lambda_1^{\mathrm{T}}(X_i - \tilde{\mu})\}\{1 + \lambda_2^{\mathrm{T}}(X_i - \tilde{\mu})\}}\}(\lambda_2 - \lambda_1)|| \\
\geq&(\lambda_1 - \lambda_2)^{\mathrm{T}}\left\{\frac{1}{I}\sum_{i=1}^{I}\frac{(X_i - \tilde{\mu})(X_i - \tilde{\mu})^{\mathrm{T}}}{\{1 + R_\lambda(R_x + |\mu - \tilde{\mu}|)\}^2}\right\}(\lambda_1 - \lambda_2) \\
=&(\lambda_1 - \lambda_2)^{\mathrm{T}}\left\{\frac{1}{I}\sum_{i=1}^{I}\frac{(X_i - \mu)(X_i - \mu)^{\mathrm{T}} + (\mu - \tilde{\mu})(\mu - \tilde{\mu})^{\mathrm{T}}}{\{1 + R_\lambda(R_x + |\mu - \tilde{\mu}|)\}^2}\right\}(\lambda_1 - \lambda_2) \\
\geq&(\lambda_1 - \lambda_2)^{\mathrm{T}}\left\{\frac{1}{I}\sum_{i=1}^{I}\frac{(X_i - \mu)(X_i - \mu)^{\mathrm{T}}}{\{1 + R_\lambda(R_x + |\mu - \tilde{\mu}|)\}^2}\right\}(\lambda_1 - \lambda_2) \\
\geq&\frac{\nu_1(V)}{\{1 + R_\lambda(R_x + |\mu - \tilde{\mu}|)\}^2}||\lambda_1 - \lambda_2||^2.
\end{aligned}$$

Therefore, $\mathcal{G}_*(\cdot)$ is strongly convex over $\{\lambda : 1 + \lambda^{\mathrm{T}}(X_i - \tilde{\mu}) \in (0, 1 + R_\lambda(R_x + ||\mu - \tilde{\mu}||))\}$ with $u = \nu_1(V)/\{1 + R_\lambda(R_x + |\mu - \tilde{\mu}|)\}^2$.

Second, if we have the strong convexity. With at least $1 - \alpha$ probability,

$$\begin{aligned}
\frac{u}{2}||\lambda_* - \lambda_b||^2 \leq&\mathcal{G}_*(\lambda_*) - \mathcal{G}_*(\lambda_*) \\
=&\mathcal{G}_*(\lambda_*) - \mathcal{G}_b(\lambda_*) + \mathcal{G}_b(\lambda_*) - \mathcal{G}_b(\lambda_b) + \mathcal{G}_b(\lambda_b) - \mathcal{G}_*(\lambda_b) \\
\leq&\frac{q_{1-\alpha/4}}{\sqrt{S}}\sigma_{\lambda_*} + 0 + \frac{q_{1-\alpha/4}}{\sqrt{S}}\sigma_{\lambda_*} \\
=&\frac{q_{1-\alpha/4}}{\sqrt{S}}(\sigma_{\lambda_*} + \sigma_{\lambda_b})
\end{aligned}$$

Because

$$\begin{aligned}
\sigma_\lambda^2 =&\frac{1}{I}\sum_{i=1}^{I}[G_*(\lambda, i) - \mathcal{G}_*(\lambda, i)]^2 \\
\leq&\frac{1}{I}\sum_{i=1}^{I}G_*(\lambda, i)^2 \\
=&\frac{1}{I}\sum_{i=1}^{I}\log^2(1 + \lambda^{\mathrm{T}}(X_i - \tilde{\mu})).
\end{aligned}$$

Because $1/S \le 1 + \lambda_b^T(X_i - \mu) \le 1 + R_\lambda(R_x + |\mu - \tilde{\mu}|)$ and $1/I \le 1 + \lambda_*^T(X_i - \mu) \le 1 + R_\lambda(R_x + |\mu - \tilde{\mu}|)$, we have

$$\begin{aligned}\frac{u}{2}||\lambda_* - \lambda_b||^2 \le& \frac{q_{1-\alpha/4}}{\sqrt{S}}(\sigma_{\lambda_*} + \sigma_{\lambda_b}) \\ \le& \frac{q_{1-\alpha/4}}{\sqrt{S}}(\max\{\log S, R_\lambda(R_x + |\mu - \tilde{\mu}|)\} + \max\{\log I, R_\lambda(R_x + |\mu - \tilde{\mu}|)\}) \\ \le& \frac{q_{1-\alpha/4}}{\sqrt{S}}\log S,\end{aligned}$$

for large S. Therefore,

$$||\lambda_* - \lambda_b|| \le \sqrt{\frac{2q_{1-\alpha/4}\log S}{u\sqrt{S}}} = \sqrt{\frac{2q_{1-\alpha/4}\log S}{\nu_1(V)\sqrt{S}}}(\nu_1 + |\mu - \tilde{\mu}|(R_x + |\mu - \tilde{\mu}|)).$$

B.2 Proof of Theorem 4.3

For convenience, denote $\xi = (x, y)$ is the sample point and the point-wise loss over sample point ξ and model θ is $h(\theta, \xi) = d(y, f_\theta(x))$.

Consider a kind of distance

$$d_{\tilde{H},\theta}(D, D') = \sup_{h,h' \in \mathcal{H}} \left|\mathbb{P}_{\xi \in D}(|h(\theta,\xi) - h'(\theta,\xi)| \ge t) - \mathbb{P}_{\xi \in D'}(|h(\theta,\xi) - h'(\theta,\xi)| \ge t)\right|.$$

Actually, total variance distance can bound this kind of distance, which is concluded as the following lemma.

Lemma B.2.

$$d_{\tilde{H},\theta}(D, D') \le \mathrm{TV}(D, D'), \forall \theta \in \Theta.$$

Proof of Lemma B.2. Since $|h - h'| \in \mathcal{H}$, then

$$\begin{aligned}d_{\tilde{H},\theta}(D, D') =& \sup_{h,h' \in \mathcal{H}} \left|\mathbb{P}_{\xi \in D}(|h(\theta,\xi) - h'(\theta,\xi)| \ge t) - \mathbb{P}_{\xi \in D'}(|h(\theta,\xi) - h'(\theta,\xi)| \ge t)\right| \\ \le& \sup_{h \in \mathcal{H}} \left|\mathbb{P}_{\xi \in D}(h(\theta,\xi) \ge t) - \mathbb{P}_{\xi \in D'}(h(\theta,\xi) \ge t)\right| \\ =& \sup_{h \in \mathcal{H}} \left| \int I(h(\theta,\xi) \ge t) p_D(\xi) d\xi - \int I(h(\theta,\xi) \ge t) p_{D'}(\xi) d\xi \right| \\ \le& \sup_{h \in \mathcal{H}} \int \left| p_D(\xi) - p_{D'}(\xi) \right| d\xi = \mathrm{TV}(D, D').\end{aligned}$$

□

Lemma B.3 (From Zhao et al. [2019]Theorem 4.1)**.**

$$L(\theta, \eta_T) \le L(\theta, \eta) + d_{\tilde{H},\theta}(D_\eta, D_T) + \min\{\mathbb{E}_{D_\eta}[|l_\eta - l_T|], \mathbb{E}_{D_T}[|l_\eta - l_T|]\},$$

where $y = l_T(x)$ for target domain D_T and $y = l_\eta(x)$ for domain D_η.

Proof. Recell that

$$\mathbb{P}_n^{\alpha}(L,\theta) = \{\eta : L(\theta,\eta) \le \mathrm{VaR}_{\mathbb{P}_n}^{\alpha}[L(\theta,\eta')]\}$$

In Lemma B.3, then assume that

$$\hat{\eta}_T = \arg\min_{\eta\in\mathbb{P}_n^{\alpha}(L,\theta)} \mathrm{TV}(D_\eta, D_T).$$

Then from Lemma B.2 and B.3,

$$\begin{aligned} L(\theta,\eta_T) \le & L(\theta,\hat{\eta}_T) + \mathrm{TV}(D_{\hat{\eta}_T}, D_T) + \min\{\mathbb{E}_{D_{\hat{\eta}_T}}[|l_{\hat{\eta}_T} - l_T|], \mathbb{E}_{D_T}[|l_{\hat{\eta}_T} - l_T|]\} \\ \le & \mathrm{VaR}_{\mathbb{P}_n}^{\alpha}[L(\theta,\eta)] + \min_{\eta\in\mathbb{P}_n^{\alpha}(L,\theta)} \mathrm{TV}(D_\eta, D_T) + \min\{\mathbb{E}_{D_{\hat{\eta}_T}}[|l_{\hat{\eta}_T} - l_T|], \mathbb{E}_{D_T}[|l_{\hat{\eta}_T} - l_T|]\}. \end{aligned}$$

Then from Lemma B.2 and B.3, (i) holds.
Because $L(\theta,\hat{\eta}_T) \le \mathrm{VaR}_{\mathbb{P}_n}^{\alpha}[L(\theta,\eta)]$, $\hat{\eta}_T \in \mathbb{P}_n^{\alpha}(L,\theta)$. Then

$$\begin{aligned} & \min_{\eta\in\mathbb{P}_n^{\alpha}(L,\theta)} \Big[\mathrm{TV}(D_\eta, D_T) + \min\Big\{\mathbb{E}_{D_\eta}[|l_\eta - l_T|], \mathbb{E}_{D_T}[|l_\eta - l_T|]\Big\}\Big] \\ \le & \mathrm{TV}(\mathcal{D}_{\hat{\eta}_T}, \mathcal{D}_T) + \min\{\mathbb{E}_{\mathcal{D}_{\hat{\eta}_T}}[|l_{\hat{\eta}_T} - l_T|], \mathbb{E}_{\mathcal{D}_T}[|l_{\hat{\eta}_T} - l_T|]\}. \end{aligned}$$

□

B.3 Proof of Proposition 4.4

Proof. Let Z_i denote the random variable iid sampled from the uniform distribution on $\{1,\dots,n\}$, then we can express L_b as the following:

$$L_b = \frac{1}{s}\sum_{k=1}^{n} 1(Z_i = k) d(y_k, f(x_k)),$$

then because $\mathbb{P}(Z_i = k) = 1/n$ for $1 \le i \le s, 1 \le k \le n$, then

$$\mathbb{E}L_b = \frac{1}{n}\sum_{i=1}^{I} d(y_i, f(x_i)).$$

and we can compute the variance

$$\begin{aligned} \mathrm{var}(L_b) = & \mathrm{var}(\frac{1}{s}\sum_{k=1}^{n} 1(Z_i = k) d(y_k, f(x_k))) \\ = & \frac{1}{s}\mathrm{var}(\sum_{k=1}^{n} 1(Z_1 = k) d(y_k, f(x_k))). \end{aligned}$$

Denote $l_k^+ = d(y_k, f(x_k)) - n^{-1}\sum_{k=1}^{n} d(y_k, f(x_k))$ and note that $\sum_{k=1}^{n} l_k^+ = 0$, then

$$\begin{aligned} \mathrm{var}(\sum_{k=1}^{n} 1(Z_1 = k) d(y_k, f(x_k))) = & \sum_{k=1}^{n} \mathrm{var}(1(Z_1 = k))[l_k^+]^2 + \sum_{1\le s<t\le n} \mathrm{cov}(1(Z_1 = s), 1(Z_1 = t)) l_s^+ l_t^+ \\ = & \sum_{k=1}^{n} \{\mathrm{var}(1(Z_1 = 1)) - \mathrm{cov}(1(Z_1 = 1), 1(Z_1 = 2))\}[l_k^+]^2 \end{aligned}$$

$$= \frac{1}{n} \sum_{k=1}^{n} [l_k^+]^2.$$

Because

$$L_b = \frac{1}{s} \left(\sum_{k=1}^{n} 1(Z_i = k) d(y_k, f(x_k)) \right)$$

is the mean of s independent variables, when $\text{var}(\sum_{k=1}^{n} 1(Z_1 = k) d(y_k, f(x_k))) < \infty$, then from the central limit theorem,

$$\sqrt{s} \left(L_b - \frac{1}{n} \sum_{i=1}^{I} d\Big(y_i, f(x_i)\Big) \right) \xrightarrow{d} \mathcal{N} \left(0, \frac{1}{n} \sum_{j=1}^{n} \Big[d\Big(y_j, f(x_j)\Big) - \frac{1}{n} \sum_{i=1}^{I} d\Big(y_i, f(x_i)\Big) \Big]^2 \right).$$

□

C Datasets and Tasks

For tabular datasets, we follow the guidelines of WhyShift [Liu et al., 2023a] for downloading and pre-processing data. Similarly, for image datasets, we follow the guidelines of DomainBed [Gulrajani and Lopez-Paz, 2020].

- **ACS Income.** The task is to predict whether an individual's income is above $50,000. We filter the dataset to only include individuals above the age of 16, usual working hours of at least 1 hour per week in the past year, and an income of at least $100. We use Puerto Rico as training set and California as test set.

- **ACS Public Coverage (Pubcov).** The task is to predict whether an individual has public health insurance. We focus on low-income individuals who are not eligible for Medicare by filtering the dataset to only include individuals under the age of 65 and with an income of less than $30,000. We use New York 2010 as training set and New York 2017 as test set.

- **ACS Mobility.** The task is to predict whether an individual had the same residential address one year ago. We filter the dataset to only include individuals between the ages of 18 and 35, which increases the difficulty of the prediction task. We use California as training set and Oregon as test set.

- **Taxi.** The task is to predict whether the total ride duration time exceeds 30 minutes, based on location and temporal features. We filter the data in 2017 and remove some extremely large or small features (e.g. samples with too long distances which can be easily classified). We use New York as the training set and Bogota as the test set.

- **PACS.** This dataset consists of four domains: Photos (P), Art painting (A), Cartoon (C), and Sketch (S). This dataset contains 9991 examples of dimension (3, 224, 224) and 7 classes.

- **ColoredMNIST.** This dataset contains a disjoint set of digits colored either red or blue with 70; 000 examples of dimension (2, 28, 28) and 2 classes.

D Optimization in the Empirical Likelihood

To obtain weights $\{p_i^{(b)}\}_{i=1}^s$ for the resampled samples, we need to solve the following convex optimization problem:

$$\max \sum_{i=1}^{s} \log p_i^{(b)} \tag{7}$$
$$\text{subject to} \quad \sum_{i=1}^{s} p_i^{(b)} = 1, \quad \sum_{i=1}^{s} p_i^{(b)} \mathcal{T}(x_i^{(b)}) = \frac{1}{m} \sum_{j=1}^{m} \mathcal{T}(\tilde{x}_j).$$

In general, we solve the problem above using algorithm 2 [Chen et al., 2008a]:

Algorithm 2 Details about obtain $\{p_i^{(b)}\}_{i=1}^s$

Considering $G^{(b)}(\lambda) = \sum_{i=1}^{n} \log\{1 + \lambda^{\mathrm{T}}(\mathcal{T}(x_i^{(b)}) - \frac{1}{m}\sum_{j=1}^{m} \mathcal{T}(\tilde{x}_j))\}$, which is a concave function. We can easily obtain $\lambda^{(k)} = \arg\max G^{(b)}(\lambda)$.
for $i = 1$ to n **do**
$p_i^{(b)} = 1/\{n[1 + (\lambda^{(b)})^{\mathrm{T}}(\mathcal{T}(x_i^{(b)}) - \frac{1}{m}\sum_{j=1}^{m} \mathcal{T}(\tilde{x}_j))]\}$.
end for

It is worth noting that the solution to Equation 2 exists when $\frac{1}{m}\sum_{j=1}^{m} \mathcal{T}(\tilde{x}_j)$ is an interior point of the convex hull of $\{\mathcal{T}(x_i^{(b)})\}_{i=1}^s$. However, no solution exists if it is an exterior point [Chen et al., 2008a]. For the exterior point case, we approximate to obtain $\{p_i^{(b)}\}_{i=1}^s$ using algorithm 3 [Chen et al., 2008a]:

Algorithm 3 Approximate solution for $\{p_i^{(b)}\}_{i=1}^s$ in the exterior point case

Let $\mathcal{T}(x_{s+1}^{(b)}) = -\frac{\log s}{2s}\sum_{j=i}^{s} \mathcal{T}(x_i^{(b)})$.
Obtain $\{p_i^{(b)}\}_{i=1}^{s+1}$ for $\{\mathcal{T}(x_i^{(b)})\}_{i=1}^{s+1}$ using algorithm 2 and take the first s terms.

Further, when s is large, solving $\arg\max G^{(b)}(\lambda)$ may be constrained by memory limitations, necessitating the batching of the s samples. However, direct batch solving leads to differences from the original problem, so we introduce an iterative approach combined with random shuffling to better approximate the solution, as shown in algorithm 4.

Algorithm 4 Batch Approximation of $\{p_i^{(b)}\}_{i=1}^s$

Initialize: Set $t = \sqrt{s}$ and $p_i^{(b),(0)} = \frac{1}{s}$ for all $i \in \{1, \ldots, s\}$.
for $q = 0$ to $Q - 1$ **do**
  Randomly shuffle $\{\mathcal{T}(x_i^{(b)})\}_{i=1}^s$, and partition into t subsets: $T_1, \ldots, T_t$, where $T_k = \{\mathcal{T}(x_{i,k}^{(b)})\}_{i=1}^t$ for each $k \in \{1, \ldots, t\}$.
  for $k = 1$ to t **do**
    Solve the following convex optimization problem:
    $\max \sum_{i=1}^t \log p_{i,k}^{(b),(q+1)}$ subject to $\sum_{i=1}^t p_{i,k}^{(b),(q+1)} = \sum_{i=1}^t p_{i,k}^{(b),(q)}$, $\sum_{i=1}^t p_{i,k}^{(b),(q+1)} \mathcal{T}(x_{i,k}^{(b)}) = \frac{1}{tm} \sum_{j=1}^m \mathcal{T}(\tilde{x}_j)$
  end for
end for
Combine results: Set $\{p_i^{(b)}\}_{i=1}^s = \bigcup_{k=1}^t P_k^{(b),(Q)}$, where $P_k^{(b),(Q)} = \{p_{i,k}^{(b),(Q)}\}_{i=1}^t$.

E Complexity Analysis

As shown in Algorithm 1, the ATDU method introduces additional computational complexity compared to the ERM approach. This increase is observed in both the calculation of weights $\{p_i^{(b)}\}_{i=1}^s$ and the training process.

Weight Calculation. According to Algorithms 2 and 3, the ATDU method requires solving B convex optimization problems, each involving s samples. This results in a complexity of $\mathcal{O}(Bs)$. When s is large, Algorithm 4 requires Q iterations of Algorithms 2 and 3, leading to a total complexity of $\mathcal{O}(QBs)$.

Training Process. In comparison to ERM, ATDU involves sampling B times more samples for each gradient descent step, thereby increasing the training complexity by a factor of B.

In our experiments, we typically set $B = 40$, $s = n$, and $Q = 10$.