Does Distributionally Robust Supervised Learning Give Robust Classifiers?

Weihua Hu [1 2] **Gang Niu** [2] **Issei Sato** [1 2] **Masashi Sugiyama** [2 1]

Abstract

Distributionally Robust Supervised Learning (DRSL) is necessary for building reliable machine learning systems. When machine learning is deployed in the real world, its performance can be significantly degraded because test data may follow a different distribution from training data. DRSL with f-divergences explicitly considers the worst-case distribution shift by minimizing the adversarially reweighted training loss. In this paper, we analyze this DRSL, focusing on the classification scenario. Since the DRSL is explicitly formulated for a distribution shift scenario, we naturally expect it to give a robust classifier that can aggressively handle *shifted distributions*. However, surprisingly, we prove that the DRSL just ends up giving a classifier that exactly fits *the given training distribution*, which is too pessimistic. This pessimism comes from two sources: the particular losses used in classification and the fact that the variety of distributions to which the DRSL tries to be robust is too wide. Motivated by our analysis, we propose simple DRSL that overcomes this pessimism and empirically demonstrate its effectiveness.

1. Introduction

Supervised learning has been successful in many application fields. The vast majority of supervised learning research falls into the Empirical Risk Minimization (ERM) framework (Vapnik, 1998) that assumes a test distribution to be the same as a training distribution. However, such an assumption can be easily contradicted in real-world applications due to sample selection bias or non-stationarity of the environment (Quionero-Candela et al., 2009). Once the distribution shift occurs, the performance of the traditional machine learning techniques can be significantly degraded. This makes the traditional techniques unreliable for practitioners to use in the real world.

Distributionally Robust Supervised Learning (DRSL) is a promising paradigm to tackle this problem by obtaining prediction functions explicitly robust to distribution shift. More specifically, DRSL considers a minimax game between a learner and an adversary: the adversary first shifts the test distribution from the training distribution within a pre-specified uncertainty set so as to maximize the expected loss on the test distribution. The learner then minimizes the adversarial expected loss.

DRSL with f-divergences (Bagnell, 2005; Ben-Tal et al., 2013; Duchi et al., 2016; Namkoong & Duchi, 2016; 2017) is particularly well-studied and lets the uncertainty set for test distributions be an f-divergence ball from a training distribution (see Section 2 for the detail). This DRSL has been mainly studied under the assumption that *the same continuous loss is used for training and testing*. This is not the case in the *classification* scenario, in which we care about the *0-1 loss* (i.e., the mis-classification rate) at test time, while at training time, we use a *surrogate loss* for optimization tractability.

In this paper, we revisit DRSL with f-divergences, providing novel insight for the *classification* scenario. In particular, we prove rather surprising results (Theorems 1–3), showing that when the DRSL is applied to classification, the obtained classifier ends up being optimal for the *training distribution*. This is too pessimistic for DRSL given that DRSL is explicitly formulated for a distribution shift scenario and is naturally expected to give a classifier different from the one that exactly fits the given training distribution. Such pessimism comes from two sources: the particular losses used in classification and the over-flexibility of the uncertainty set used by DRSL with f-divergences.

Motivated by our analysis, we propose simple DRSL that overcomes the pessimism of the previous DRSL by incorporating structural assumptions on distribution shift (Section 4). We establish convergence properties of our proposed DRSL (Theorem 4) and derive efficient optimization algorithms (Section 5). Finally, we demonstrate the effectiveness of our DRSL through experiments (Section 6). All the appendices of this paper are provided in the supplementary material.

Related work: Besides DRSL with f-divergences, different DRSL considers different classes of uncertainty sets for test distributions. DRSL by Globerson & Roweis

[1] University of Tokyo, Japan [2] RIKEN, Tokyo, Japan. Correspondence to: Weihua Hu <weihua916@gmail.com>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018. Copyright 2018 by the author(s).

(2006) considered the uncertainty of features deletion, while DRSL by Liu & Ziebart (2014) considered the uncertainty of unknown properties of the conditional label distribution. DRSL by Esfahani & Kuhn (2015), Blanchet et al. (2016) and Sinha et al. (2017) lets the uncertainty set of test distributions be a Wasserstein ball from the training distribution. DRSL with the Wasserstein distance can make classifiers robust to adversarial examples (Sinha et al., 2017), while DRSL with f-divergences can make classifiers robust against adversarial reweighting of data points as shown in Section 2. Recently, in the context of fair machine learning, Hashimoto et al. (2018) applied DRSL with f-divergences in an attempt to achieve fairness without demographic information.

2. Review of ERM and DRSL

In this section, we first review the ordinary ERM framework. Then, we explain a general formulation of DRSL and review DRSL with f-divergences.

Suppose training samples, $\{(x_1, y_1), \ldots, (x_N, y_N)\} \equiv \mathcal{D}$, are drawn i.i.d. from an unknown training distribution over $\mathcal{X} \times \mathcal{Y}$ with density $p(x, y)$, where $\mathcal{X} \subset \mathbb{R}^d$ and $\mathcal{Y}$ is an output domain. Let g_θ be a prediction function with parameter θ, mapping $x \in \mathcal{X}$ into a real scaler or vector, and let $\ell(\widehat{y}, y)$ be a loss between y and real-valued prediction $\widehat{y}$.

ERM: The objective of the risk minimization (RM) is

$$\min_\theta \underbrace{\mathbb{E}_{p(x,y)}[\ell(g_\theta(x), y)]}_{\equiv \mathcal{R}(\theta)}, \tag{1}$$

where $\mathcal{R}(\theta)$ is called the risk. In ERM, we approximate the expectation in Eq. (1) by training data $\mathcal{D}$:

$$\underbrace{\min_\theta \frac{1}{N} \sum_{i=1}^N \ell(g_\theta(x_i), y_i),}_{\equiv \widehat{\mathcal{R}}(\theta)} \tag{2}$$

where $\widehat{\mathcal{R}}(\theta)$ is called the empirical risk. To prevent overfitting, we can add regularization term $\Omega(\theta)$ to Eq. (2) and minimize $\widehat{\mathcal{R}}(\theta) + \lambda\Omega(\theta)$, where $\lambda \geq 0$ is a trade-off hyperparameter.

General formulation of DRSL: ERM implicitly assumes the test distribution to be the same as the training distribution, which does not hold in most real-world applications. DRSL is explicitly formulated for a distribution shift scenario, where test density $q(x, y)$ is different from training density $p(x, y)$. Let $\mathcal{Q}_p$ be an uncertainty set for test distributions. In DRSL, the learning objective is

$$\min_\theta \sup_{q \in \mathcal{Q}_p} \mathbb{E}_{q(x,y)}[\ell(g_\theta(x), y)]. \tag{3}$$

We see that Eq. (3) minimizes the risk w.r.t. the *worst-case test distribution* within the uncertainty set $\mathcal{Q}_p$.

DRSL with f-divergences: Let $q \ll p$ denote that q is absolutely continuous w.r.t. p, i.e., $p(x, y) = 0$ implies $q(x, y) = 0$. Bagnell (2005) and Ben-Tal et al. (2013) considered the particular uncertainty set

$$\mathcal{Q}_p = \{q \ll p \mid \mathrm{D}_f[q \| p] \leq \delta\}, \tag{4}$$

where $\mathrm{D}_f[\cdot\|\cdot]$ is an f-divergence defined as $\mathrm{D}_f[q\|p] \equiv \mathbb{E}_p[f(q/p)]$, and $f(\cdot)$ is convex with $f(1) = 0$. The f-divergence (Ciszar, 1967) measures a discrepancy between probability distributions. When $f(x) = x \log x$, we have the well-known Kullback-Leibler divergence as an instance of it. Hyper-parameter $\delta > 0$ in Eq. (4) controls the degree of the distribution shift. Define $r(x, y) \equiv q(x, y)/p(x, y)$. Through some calculations, the objective of DRSL with f-divergences can be rewritten as

$$\underbrace{\min_\theta \sup_{r \in \mathcal{U}_f} \mathbb{E}_{p(x,y)}[r(x, y)\ell(g_\theta(x), y)],}_{\equiv \mathcal{R}_{\mathrm{adv}}(\theta)} \tag{5}$$

$$\begin{aligned}\mathcal{U}_f \equiv \{r(x, y) \mid \mathbb{E}_{p(x,y)}\left[f\left(r(x, y)\right)\right] \leq \delta,\\ \mathbb{E}_{p(x,y)}[r(x, y)] = 1,\\ r(x, y) \geq 0, \ \forall(x, y) \in \mathcal{X} \times \mathcal{Y}\}.\end{aligned} \tag{6}$$

We call $\mathcal{R}_{\mathrm{adv}}(\theta)$ the *adversarial risk* and call the minimization problem of Eq. (5) the *adversarial risk minimization* (ARM). In ARM, the density ratio, $r(x, y)$, can be considered as the weight put by the adversary on the loss of data (x, y). Then, Eq. (5) can be regarded as a minimax game between the learner (corresponding to $\min_\theta$) and the adversary (corresponding to $\sup_{r \in \mathcal{U}_f}$): the adversary first reweights the losses using $r(\cdot, \cdot)$ so as to maximize the expected loss; the learner then minimizes the *reweighted* expected loss, i.e., adversarial risk $\mathcal{R}_{\mathrm{adv}}(\theta)$.

For notational convenience, we denote $\ell(g_\theta(x_i), y_i)$ by $\ell_i(\theta)$. Also, let $\boldsymbol{r} \equiv (r_1, \ldots, r_N)$ be a vector of density ratios evaluated at training data points, i.e., $r_i \equiv r(x_i, y_i)$ for $1 \leq i \leq N$. Equations (5) and (6) can be empirically approximated as[1]

$$\underbrace{\min_\theta \sup_{\boldsymbol{r} \in \widehat{\mathcal{U}}_f} \frac{1}{N} \sum_{i=1}^N r_i \ell_i(\theta)}_{\equiv \widehat{\mathcal{R}}_{\mathrm{adv}}(\theta)}, \tag{7}$$

$$\widehat{\mathcal{U}}_f = \left\{ \boldsymbol{r} \;\middle|\; \frac{1}{N} \sum_{i=1}^N f(r_i) \leq \delta, \ \frac{1}{N} \sum_{i=1}^N r_i = 1, \ \boldsymbol{r} \geq 0 \right\}, \tag{8}$$

[1] The formulation in Eqs. (7) and (8) is similar to Duchi et al. (2016), Namkoong & Duchi (2016) and Namkoong & Duchi (2017) except that they decay δ linearly w.r.t. the number of training data N. Different from us, they assume $\delta = 0$ in Eq. (4) (thus, their objective is the ordinary risk) and try to be robust to apparent distribution fluctuations due to the *finiteness of training samples*. On the other hand, we consider using the same $\delta > 0$ for both Eqs. (4) and (8) and try to be robust to the actual distribution change between training and test stages.

where the inequality constraint for a vector is applied in an element-wise fashion. We call $\widehat{\mathcal{R}}_{\mathrm{adv}}(\theta)$ the *adversarial empirical risk* and call the minimization problem of Eq. (7) the *adversarial empirical risk minimization* (AERM). In AERM, the adversary (corresponding to $\sup_{\boldsymbol{r}\in\widehat{\mathcal{U}}_f}$) reweights data losses through $\boldsymbol{r}$ to maximize the empirical loss in Eq. (7). To prevent overfitting, we can add regularization term $\Omega(\theta)$ to Eq. (7).

3. Analysis of DRSL with f-divergences in classification

At first glance, DRSL with f-divergences (which we call ARM and AERM in this paper) is reasonable to give a distributionally robust classifier in the sense that it explicitly minimizes the loss for the shifted worst-case *test distribution.* However, we show rather surprising results, suggesting that the DRSL, when applied to *classification*, still ends up giving a classifier optimal for a *training* distribution. This is too pessimistic for DRSL because it ends up behaving similarly to ordinary ERM-based supervised classification that does *not* explicitly consider distribution shift. To make a long story short, our results hold because of the particular losses used in classification (especially, the 0-1 loss at test time) and the overly flexible uncertainty sets used by ARM and AERM. We will detail these points after we state our main results.

Classification setting: Let us first briefly review classification settings to set up notations. In binary classification, we have $g_\theta(\cdot) : x \mapsto \widehat{y} \in \mathbb{R}$, $\mathcal{Y} = \{+1, -1\}$ and $\ell(\cdot,\cdot) : \mathbb{R} \times \mathcal{Y} \to \mathbb{R}_{\geq 0}$. In K-class classification for $K \geq 2$, we have $g_\theta(\cdot) : x \mapsto \widehat{y} \in \mathbb{R}^K$, $\mathcal{Y} = \{1, 2, \ldots, K\}$ and $\ell(\cdot,\cdot) : \mathbb{R}^K \times \mathcal{Y} \to \mathbb{R}_{\geq 0}$. The goal of classification is to learn the prediction function that minimizes the mis-classification rate on the test distribution. The mis-classification rate corresponds to the *use of the 0-1 loss*, i.e., $\ell(\widehat{y}, y) \equiv \mathbf{1}\{\mathrm{sign}(\widehat{y}) \neq y\}$ for binary classification, and $\ell(\widehat{y}, y) \equiv \mathbf{1}\{\mathrm{argmax}_k \widehat{y}_k \neq y\}$ for multi-class classification, where $\mathbf{1}\{\cdot\}$ is the indicator function and $\widehat{y}_k$ is the k-th element of $\widehat{y} \in \mathbb{R}^K$. However, since the 0-1 loss is non-convex and non-continuous, learning with it is difficult in practice. Therefore, at training time, we instead use *surrogate losses* that are easy to optimize, such as the logistic loss and the cross-entropy loss.

In the following, we state our main results, analyzing ARM and AERM in the classification scenario by considering the use of the 0-1 loss and a surrogate loss.

The 0-1 loss case: Theorem 1 establishes the non-trivial relationship between the adversarial risk and the ordinary risk when the 0-1 loss is used.

Theorem 1. *Let $\ell(\widehat{y}, y)$ be the 0-1 loss. Then, there is a monotonic relationship between $\mathcal{R}_{\mathrm{adv}}(\theta)$ and $\mathcal{R}(\theta)$ in the sense that for any pair of parameters θ_1 and θ_2, the followings hold.*

If $\mathcal{R}_{\mathrm{adv}}(\theta_1) < 1$, then

$$\mathcal{R}_{\mathrm{adv}}(\theta_1) < \mathcal{R}_{\mathrm{adv}}(\theta_2) \Longleftrightarrow \mathcal{R}(\theta_1) < \mathcal{R}(\theta_2). \tag{9}$$

If $\mathcal{R}_{\mathrm{adv}}(\theta_1) = 1$, then

$$\mathcal{R}(\theta_1) \leq \mathcal{R}(\theta_2) \Longrightarrow \mathcal{R}_{\mathrm{adv}}(\theta_2) = 1. \tag{10}$$

The same monotonic relationship also holds between their empirical approximations: $\widehat{\mathcal{R}}_{\mathrm{adv}}(\theta)$ and $\widehat{\mathcal{R}}(\theta)$.

See Appendix A for the proof. Theorem 1 shows a surprising result that *when the 0-1 loss is used*, $\mathcal{R}(\theta)$ and $\mathcal{R}_{\mathrm{adv}}(\theta)$ are essentially equivalent objective functions in the sense that the minimization of one objective function results in the minimization of another objective function. This readily implies that $\mathcal{R}(\theta)$ and $\mathcal{R}_{\mathrm{adv}}(\theta)$ have exactly the same set of global minima in the regime of $\mathcal{R}_{\mathrm{adv}}(\theta) < 1$. An immediate practical implication is that if we select hyper-parameters such as λ for regularization according to *the adversarial risk with the 0-1 loss*, we will end up choosing hyper-parameters that attain the minimum mis-classification rate on the *training distribution*.

The surrogate loss case: We now turn our focus on the training stage of classification, where we use a *surrogate loss* instead of the 0-1 loss. In particular, for binary classification, we consider a class of classification calibrated losses (Bartlett et al., 2006) that are margin-based, i.e., $\ell(\widehat{y}, y)$ is a function of product $y\widehat{y}$. For multi-class classification, we consider a class of classification calibrated losses (Tewari & Bartlett, 2007) that are invariant to class permutation, i.e., for any class permutation $\pi : \mathcal{Y} \to \mathcal{Y}$, $\ell(\widehat{y}^\pi, \pi(y)) = \ell(\widehat{y}, y)$ holds, where $\widehat{y}^\pi_k = \widehat{y}_{\pi(k)}$ for $1 \leq k \leq K$. Although we only consider the sub-class of general classification-calibrated losses (Bartlett et al., 2006; Tewari & Bartlett, 2007), we note that ours still includes some of the most widely used losses: the logistic, hinge, and exponential losses for binary classification and the softmax cross entropy loss for multi-class classification.

We first review Proposition 1 by Bartlett et al. (2006) and Tewari & Bartlett (2007) that justifies the use of classification-calibrated losses in ERM for classification. We then show a surprising fact in Theorem 2 that the similar property also holds for ARM using the sub-class of classification-calibrated losses.

Proposition 1 (Bartlett et al. (2006); Tewari & Bartlett (2007))**.** *Let $\ell(\widehat{y}, y)$ be a classification calibrated loss, and assume that the hypothesis class is equal to all measurable functions. Then, the risk minimization (RM) gives the Bayes optimal classifier*[2].

Theorem 2. *Let $f(\cdot)$ be differentiable, the hypothesis class be all measurable functions, and $\ell(\widehat{y}, y)$ be a classification-calibrated loss that is margin-based or invariant to class*

[2] The classifier that minimizes the mis-classification rate for the training density $p(x, y)$ (the 0-1 loss is considered), i.e., the classifier whose prediction on x is equal to $\arg\max_{y\in\mathcal{Y}} p(y|x)$.

permutation. Let $g^{(\mathrm{adv})}$ be any solution of ARM[3] *under the above setting, and define*

$$r^* \equiv \operatorname*{arg\,max}_{r \in \mathcal{U}_f} \mathbb{E}_{p(x,y)}[r(x,y)\ell(g^{(\mathrm{adv})}(x), y)]. \tag{11}$$

Then, the prediction of $g^{(\mathrm{adv})}$ coincides with that of the Bayes optimal classifier almost surely over $q^*(x) \equiv \sum_{y \in \mathcal{Y}} r^*(x,y)p(x,y)$. Furthermore, among the solutions of ARM, there exists $g^{(\mathrm{adv})}$ whose prediction coincides with that of the Bayes optimal classifier almost surely over $p(x)$.

See Appendix B for the proof. Theorem 2 indicates that ARM, similarly to RM, ends up giving the optimal decision boundary for the *training* distribution, if the hypothesis class is all measurable functions and we have access to true density $p(x,y)$. Even though the assumptions made are strong, Theorem 2 together with Proposition 1 highlight the non-trivial fact that when a certain surrogate loss is used, AERM and ERM demonstrate the similar *asymptotic* behavior in classification.

We proceed to consider a more practical scenario, where we only have a finite amount of training data and the hypothesis class is limited. In the rest of the section, we focus on a differentiable loss and a real-valued scalar output, i.e., $\widehat{y} \in \mathbb{R}$, which includes the scenario of binary classification.

We first define the notion of a *steeper* loss, which will play a central role in our result.

Definition 1 (Steeper loss)**.** Loss function $\ell_{\mathrm{steep}}(\widehat{y}, y)$ is said to be steeper than loss function $\ell(\widehat{y}, y)$, if there exists a non-constant, non-decreasing and non-negative function $h : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ such that

$$\frac{\partial \ell_{\mathrm{steep}}(\widehat{y}, y)}{\partial \widehat{y}} = h(\ell(\widehat{y}, y)) \frac{\partial \ell(\widehat{y}, y)}{\partial \widehat{y}}. \tag{12}$$

For example, following Definition 1, we can show that the exponential loss is *steeper* than the logistic loss. Intuitively, outlier-sensitive losses are *steeper* than more outlier-robust losses. Lemma 1 shows an important property of a steeper loss in a classification scenario.

Lemma 1. *Let $\ell(\widehat{y}, y)$ be a margin-based convex classification-calibrated loss. Then, its steeper loss defined in Eq.* (12) *is also convex classification-calibrated if $h(\ell(0, y)) > 0$.*

See Appendix C for the proof.

Now we are ready to state our result in Theorem 3 that considers $\widehat{y} \in \mathbb{R}$. Theorem 3 holds for *any* hypothesis class that is parametrized by θ and sub-differentiable w.r.t. θ, e.g., linear-in-parameter models and deep neural networks.

Theorem 3. *Let θ^* be a stationary point of AERM in Eq.* (7) *using $\ell(\widehat{y}, y)$. Then, there exists a steeper loss function, $\ell_{\mathrm{DRSL}}(\widehat{y}, y)$, such that θ^* is also a stationary point of the following ERM.*

$$\min_{\theta} \frac{1}{N} \sum_{i=1}^{N} \ell_{\mathrm{DRSL}}(g_\theta(x_i), y_i). \tag{13}$$

See Appendix D for the proof.

Remark 1 (Conditions for convexity)**.** Let $\ell(\widehat{y}, y)$ be convex in $\widehat{y}$, $g_\theta(x)$ be a linear-in-parameter model. Then, both AERM in Eq. (7) and ERM in Eq. (13) become convex in θ. This implies that the stationary point θ^* in Theorem 3 turns out to be the *global optimum* for both Eqs. (7) and (13) in this usual setting.

Note that Theorem 3 holds for general real-valued scalar prediction, i.e., $\widehat{y} \in \mathbb{R}$; thus, the result holds for ordinary regression (using the same loss for training and testing) as well as for binary classification. However, as we discuss in the following, the implication of Theorem 3 is drastically different for the two scenarios.

Implication for classification: Theorem 3 together with Lemma 1 indicate that under a mild condition,[4] AERM using a convex classification-calibrated margin-based loss reduces to Eq. (13), which is ERM using a convex classification-calibrated loss. This implies that AERM, similarly to ordinary ERM using a classification-calibrated loss, will try to give a classifier optimal for the *training distribution*.

Why does the use of the steeper surrogate loss fail to give meaningful robust classifiers? This is because we are dealing with classification tasks, where we care about the performance *in terms of the 0-1 loss* at test time. The use of the steeper *surrogate* loss may make a classifier distributionally robust *in terms of the surrogate loss*,[5] but not necessarily so in terms of the 0-1 loss. Moreover, even if we obtain a classifier that minimizes the adversarial risk *in terms of the 0-1 loss*, the obtained classifier ends up being optimal for the training distribution (see Theorem 1). In any case, the use of the steeper loss does not in general give classifiers that are robust to change from a training distribution.

In summary, in the *classification* scenario, the use of the steeper loss does more harm (making a classifier sensitive

[3] There can be multiple solutions that achieve the same minimum adversarial risk.

[4] The condition that $h(\ell(0, y)) > 0$ in Lemma 1. Whether the condition holds or not generally depends on the uncertainty set, the model, the loss function, and training data. Nonetheless, the condition is mild in practice; especially, the condition always holds when the Kullback-Leibler divergence is used. See Appendix C for detailed discussion.

[5] For fixed δ (non-decaying w.r.t. N), whether AERM is consistent with ARM or not is an open problem. Nevertheless, we empirically confirm in Section 6 that AERM achieves lower adversarial risk than other baselines *in terms of the surrogate loss*.

to outliers due to the use of the steeper surrogate loss) than good (making a classifier robust to change from a training distribution).

Implication for ordinary regression: For comparison, let us rethink about the classical regression scenario, in which we use *the same loss*, e.g., the squared loss, during training and testing. In such a case, the use of the steeper loss may indeed make regressors distributionally robust *in terms of the same loss*. Nonetheless, learning can be extremely sensitive to outliers due to the use of the *steeper* loss. Hence, when applying DRSL with f-divergences to real-world regression tasks, we need to pay extra attention to ensure that there are no outliers in datasets.

4. DRSL with Latent Prior Probability Change

In this section, motivated by our theoretical analysis in Section 3, we propose simple yet practical DRSL that overcomes the over pessimism of ARM and AERM in the classification scenario. We then analyze its convergence property and discuss the practical use of our DRSL.

Theoretical motivation: What insight can we get from our theoretical analyses in Section 3? Our key insight from proving the theorems is that the adversary of ARM has too much (non-parametric) freedom to shift the test distribution, and as a result, the learner becomes overly pessimistic. In fact, the proofs of all the theorems rely on the over-flexibility of the uncertainty set $\mathcal{U}_f$ in Eq. (6), i.e., the values of $r(\cdot,\cdot)$ are *not tied together* for different (x, y) within $\mathcal{U}_f$ (see Eqs. (5) and (6)). Consequently, the adversary of ARM simply assigns larger weight $r(x, y)$ to data (x, y) with a larger loss. This fact, combined with the fact that we use the different losses during training and testing in classification (see discussion at the end of Section 3), led to the pessimistic results of Theorems 1–3.

Our theoretical insight suggests that in order to overcome the pessimism of ARM applied to classification, it is crucial to *structurally constrain* $r(\cdot,\cdot)$ in $\mathcal{U}_f$, or equivalently, to impose *structural assumptions* on the distribution shift. To this end, in this section, we propose DRSL that overcomes the limitation of the DRSL by incorporating structural assumptions on distribution shift.

Practical structural assumptions: In practice, there can be a variety of ways to impose structural assumptions on distribution shift. Here, as one possible way, we adopt the *latent prior probability change assumption* (Storkey & Sugiyama, 2007) because this particular class of assumptions enjoys the following two practical advantages.

1. Within the class, users of our DRSL can easily and intuitively model their assumptions on distribution shift (see the discussion at the end of this section).
2. Efficient learning algorithms can be derived (see Section 5).

Let us introduce a latent variable $z \in \mathcal{Z} \equiv \{1, \ldots, S\}$, which we call a *latent category*, where S is a constant. The latent prior probability change assumes

$$p(x, y|z) = q(x, y|z), \quad q(z) \neq p(z), \tag{14}$$

where p and q are the training and test densities, respectively. The intuition is that we assume a two-level *hierarchical* data-generation process: we first sample latent category z from the prior and then sample actual data (x, y) conditioned on z. We then assume that only the prior distribution over the latent categories changes, leaving the conditional distribution intact.

We assume the structural assumption in Eq. (14) to be provided externally by users of our DRSL based on their knowledge of potential distribution shift, rather than something to be inferred from data. As we will see at the end of this section, specifying Eq. (14) amounts to grouping training data points according to their latent categories, which is quite intuitive to do in practice.

Objective function of our DRSL: With the latent prior probability change of Eq. (14), the uncertainty set for test distributions in our DRSL becomes

$$\begin{aligned}\mathcal{Q}_p = \{q \ll p \mid \mathrm{D}_f[q(x, y, z)||p(x, y, z)] \leq \delta,\\ q(x, y|z) = p(x, y|z)\}.\end{aligned} \tag{15}$$

Then, corresponding to Eq. (3), the objective of our DRSL can be written as

$$\min_{\theta} \underbrace{\sup_{w \in \mathcal{W}_f} \mathbb{E}_{p(x,y,z)}\left[w(z)\ell(g_\theta(x), y)\right]}_{\equiv \mathcal{R}_{\text{s-adv}}(\theta)}, \tag{16}$$

$$\begin{aligned}\mathcal{W}_f \equiv \Bigg\{ w(z) \,\Bigg|\, &\sum_{z \in \mathcal{Z}} p(z) f\left(w(z)\right) \leq \delta, \\ &\sum_{z \in \mathcal{Z}} p(z)w(z) = 1,\ w(z) \geq 0,\ \forall z \in \mathcal{Z} \Bigg\},\end{aligned} \tag{17}$$

where $w(z) \equiv q(z)/p(z) = q(x, y, z)/p(x, y, z)$ because of $q(x, y|z) = p(x, y|z)$. We call $\mathcal{R}_{\text{s-adv}}(\theta)$ the *structural adversarial risk* and call the minimization problem of Eq. (16) the *structural adversarial risk minimization* (structural ARM). Similarly to ARM, structural ARM is a minimax game between the learner and the adversary. Differently from ARM, the adversary of structural ARM (corresponding to $\sup_{w \in \mathcal{W}_f}$) uses $w(\cdot)$ to reweight data; hence, it has much less (only parametric) freedom to shift the test distribution compared to the adversary of ARM that uses non-parametric weight $r(\cdot,\cdot)$ (see Eq. (5)). Because of this limited freedom for the adversary, we can show that Theorems 1–3 do *not* hold for structural ARM, and we can expect to learn meaningful classifiers that are robust to *structurally constrained* distribution shift.

Discussion and proposal of evaluation metric for distributional robustness: Recall from Theorem 1 that when the 0-1 loss is used, the adversarial risk ends up being equivalent to the ordinary risk as an evaluation metric, which is too pessimistic as a metric for the distributional robustness of a classifier. In contrast, we can easily verify that our structural adversarial risk using the 0-1 loss does not suffer from the pessimism. We argue that our structural adversarial risk can be an alternative metric in distributionally robust classification. To better understand its property, inspired by Namkoong & Duchi (2017), we decompose it as[6]

$$\mathcal{R}_{\text{s-adv}}(\theta) = \underbrace{\mathcal{R}(\theta)}_{\text{(a) ordinary risk}} + \sqrt{\delta}\cdot\underbrace{\sqrt{\sum_{z\in\mathcal{Z}} p(z)(\mathcal{R}_z(\theta)-\mathcal{R}(\theta))^2}}_{\text{(b) sensitivity}}, \quad (18)$$

where $\mathcal{R}_z(\theta)(\equiv \mathbb{E}_{p(x,y|z)}[\ell(g_\theta(x),y)])$ is the risk of the classifier on latent category $z\in\mathcal{Z}$. We see that $\mathcal{R}_{\text{s-adv}}(\theta)$ in Eq. (18) contains the *risk variance* term (b). This variance term (b) can be large when the obtained classifier performs extremely poorly on a small number of latent categories. Once a test density concentrates on those poorly-performed latent categories, the test accuracy of the classifier can extremely deteriorate. In this sense, the classifier with large (b) is sensitive to distribution shift. In contrast, the small risk variance (b) indicates that the obtained classifier attains similar accuracy on all the latent categories. In such a case, the test accuracy of the classifier is insensitive to latent category prior change. In this sense, the classifier with small (b) is robust to distribution shift. To sum up, the additional term (b) measures the sensitivity of the classifier to the specified structural distribution shift.

On the basis of the above discussion, we see that $\mathcal{R}_{\text{s-adv}}(\theta)$ in Eq. (18) simultaneously captures (a) the ordinary risk, i.e., the mis-classification rate *when no distribution shift occurs*, and (b) the *sensitivity* to distribution shift. In this sense, our structural adversarial risk is an intuitive and reasonable metric for distributional robustness of a classifier, and we will employ this metric in our experiments in Section 6.

Empirical approximation: We explain how to empirically approximate the objective functions in Eqs. (16) and (17) using training data $\mathcal{D}' \equiv \{(x_1,y_1,z_1),\ldots,(x_N,y_N,z_N)\}$ drawn independently from $p(x,y,z)$.

Define $\mathcal{G}_s \equiv \{i \mid z_i = s, 1\le i\le N\}$ for $1\le s\le S$, which is a set of data indices belonging to latent category s. In our DRSL, users are responsible for specifying the groupings of training data points, i.e., $\{\mathcal{G}_s\}_{s=1}^{S}$. By specifying these groupings, the users incorporate their structural assumptions on distribution shift into our DRSL. We will discuss how this can be done in practice at the end of this section.

For notational convenience, let $w_s \equiv w(s)$, $1\le s\le S$, and define $\boldsymbol{w}\equiv(w_1,\ldots,w_S)$. Equations (16) and (17) can be empirically approximated as follows using $\mathcal{D}'$:

$$\underbrace{\min_{\theta}\sup_{\boldsymbol{w}\in\widehat{\mathcal{W}}_f}\frac{1}{N}\sum_{s=1}^{S} n_s w_s \widehat{\mathcal{R}}_s(\theta)}_{\equiv\,\widehat{\mathcal{R}}_{\text{s-adv}}(\theta)} \quad (19)$$

$$\widehat{\mathcal{W}}_f = \left\{\boldsymbol{w}\in\mathbb{R}^S \,\middle|\, \frac{1}{N}\sum_{s=1}^{S} n_s f(w_s)\le\delta, \right. \quad \left. \frac{1}{N}\sum_{s=1}^{S} n_s w_s = 1,\ \boldsymbol{w}\ge 0\right\}, \quad (20)$$

where n_s is the cardinality of $\mathcal{G}_s$ and $\widehat{\mathcal{R}}_s(\theta)(\equiv \frac{1}{n_s}\sum_{i\in\mathcal{G}_s}\ell_i(\theta))$ is the average loss of all data points in $\mathcal{G}_s$. We call $\widehat{\mathcal{R}}_{\text{s-adv}}(\theta)$ the *structural adversarial empirical risk* and call the minimization problem of Eq. (19) the *structural adversarial empirical risk minimization* (structural AERM). We can add regularization term $\Omega(\theta)$ to Eq. (19) to prevent overfitting.

Convergence rate and estimation error: We establish the convergence rate of the model parameter and the order of the estimation error for structural AERM in terms of the number of training data points N. Due to the limited space, we only present an informal statement here. The formal statement can be found in Appendix G and its proof can be found in Appendix H.

Theorem 4 (Convergence rate and estimation error, informal statement)**.** *Let θ^* be the solution of structural ARM, and $\widehat{\theta}_N$ be the solution of regularized structural AERM given training data of size N. Assume $g_\theta(x)$ is linear in θ, and regularization hyper-parameter λ decreases at a rate of $\mathcal{O}(N^{-1/2})$. Under mild conditions, as $N\to\infty$, we have $\|\widehat{\theta}_N-\theta^*\|_2 = \mathcal{O}(N^{-1/4})$ and consequently, $|\mathcal{R}_{\text{s-adv}}(\widehat{\theta}_N)-\mathcal{R}_{\text{s-adv}}(\theta^*)| = \mathcal{O}(N^{-1/4})$.*

Notice that the convergence rate of $\widehat{\theta}_N$ to θ^* is not the optimal parametric rate $\mathcal{O}(N^{-1/2})$. This is because the inner maximization of Eq. (19) converges in $\mathcal{O}(N^{-1/4})$ that slows down the entire convergence rate. Theorem 4 applies to any f-divergence where $f(t)$ is nonlinear in t, while knowing which f-divergence is used may improve the result to the optimal parametric rate.

Discussion on groupings: In our structural ARM and AERM, users need to incorporate their structural assumptions by grouping training data points. Here, we discuss how this can be done in practice.

Most straightforwardly, a user of our DRSL may assume

[6] This particular decomposition holds when the Pearson (PE) divergence is used and δ is not so large. Refer to Appendix E for the derivation. Analogous decomposition can be also derived when other f-divergences are used.

class prior change (Saerens et al., 2002) or sub-category[7] prior change. To incorporate such assumptions into our DRSL, the user can simply group training data by class labels or a sub-categories, respectively.

Alternatively, a user of our DRSL can group data by available meta-information of data such as time and places in which data are collected. The intuition is that data collected in the same situations (e.g., time and places) are likely to *"share the same destiny"* in the future distribution shift; hence, it is natural to assume that only the *prior* over the situations changes at test time while the conditionals remain the same.

In any case, it is crucial that the users provide structural assumptions on distribution shift so that we can overcome the pessimism of ARM and AERM for classification raised in Section 3.

5. Efficient Learning Algorithms

In this section, we derive efficient gradient-based learning algorithms for our structural AERM in Eq. (19). Thanks to Danskin's theorem (Danskin, 1966), we can obtain the gradient $\nabla_\theta \widehat{\mathcal{R}}_{\text{s-adv}}(\theta)$ as

$$\nabla_\theta \widehat{\mathcal{R}}_{\text{s-adv}}(\theta) = \frac{1}{N}\sum_{s=1}^{S} n_s w_s^* \nabla_\theta \widehat{\mathcal{R}}_s(\theta), \tag{21}$$

where $\boldsymbol{w}^* = (w_1^*, \ldots, w_S^*)$ is the solution of inner maximization of AERM in Eq. (19).

In the following, we show that $\boldsymbol{w}^*$ can be obtained very efficiently for two well-known instances of f-divergences.

Kullback-Leibler (KL) divergence: For the KL divergence, $f(x) = x \log x$, we have

$$w_s^* = \frac{N}{Z(\gamma)} \cdot \exp\left(\frac{\widehat{\mathcal{R}}_s(\theta)}{\gamma}\right),\ 1 \le s \le S, \tag{22}$$

where γ is a scalar such that the first constraint of $\widehat{\mathcal{W}}_f$ in Eq. (20) holds with equality, and $Z(\gamma) \equiv \sum_{s=1}^{S} n_s \exp\left(\widehat{\mathcal{R}}_s(\theta)/\gamma\right)$ is a normalizing constant in order to satisfy the second constraint of $\widehat{\mathcal{W}}_f$. To compute γ, we can simply perform a binary search.

Pearson (PE) divergence: For the PE divergence, $f(x) = (x-1)^2$. For small δ, $\boldsymbol{w} \ge 0$ of $\widehat{\mathcal{W}}_f$ is often satisfied in practice. We drop the inequality for simplicity; then, the solution of the inner maximization of Eq. (19) becomes analytic and efficient to obtain:

$$\boldsymbol{w}^* = \sqrt{\frac{N\delta}{\sum_{s=1}^{S} n_s v_s^2}}\boldsymbol{v} + \mathbf{1}_S, \tag{23}$$

where $\mathbf{1}_S$ is the S-dimensional vector with all the elements equal to 1. $\boldsymbol{v}$ is the S-dimensional vector such that $v_s = \widehat{\mathcal{R}}_s(\theta) - \widehat{\mathcal{R}}(\theta),\ 1 \le s \le S$.

Computational complexity: The time complexity for obtaining $\boldsymbol{w}^*$ is: $\mathcal{O}(mS)$ for the KL divergence and $\mathcal{O}(S)$ for the PE divergence, where m is the number of the binary search iterations to compute γ in Eq. (22). Calculating the adversarial weights therefore adds negligible computational overheads to computing $\nabla \ell_i(\theta)$ and $\ell_i(\theta)$ for $1 \le i \le N$, which for example requires $\mathcal{O}(Nb)$-time for a b-dimensional linear-in-parameter model.

6. Experiments

In this section, we experimentally analyze our DRSL (structural AERM) in classification by comparing it with ordinary ERM and DRSL with f-divergences (AERM). We empirically demonstrate (i) the undesirability of AERM in classification and (ii) the robustness of structural AERM against specified distribution shift.

Datasets: We obtained six classification datasets from the UCI repository (Blake & Merz, 1998), two of which are for multi-class classification. We also obtained MNIST (LeCun et al., 1998) and 20newsgroups (Lang, 1995). Refer to Appendix I for the details of the datasets.

Evaluation metrics: We evaluated the three methods (ordinary ERM, AERM and structural AERM) with three kinds of metrics: the ordinary risk, the adversarial risk, and the structural adversarial risk, where the *0-1 loss is used* for all the metrics.[8] We did not explicitly report the adversarial risk in our experiments because of Theorem 1.

Both the risk and structural adversarial risk are estimated using held-out test data. In particular, the structural adversarial risk can be estimated similarly to Eqs. (19) and (20), i.e., calculating the mis-classification rate on the held-out test data and *structurally and adversarially reweight* them. See discussion of Eq. (18) for why the structural adversarial risk is a meaningful evaluation metric to measure distributional robustness of classifiers.

Experimental protocols: For our DRSL, we consider learning classifiers robust against (a) the class prior change and (b) the sub-category prior change. This corresponds to grouping training data by (a) class labels and (b) sub-category labels, respectively. In the benchmark datasets, the sub-category labels are not available. Hence, we manually created such labels as follows. First, we converted the original multi-class classification problems into classification problems with fewer classes by integrating some classes together. Then, the original class labels are regarded as the subcategories. In this way, we converted the satimage, letter and MNIST datasets into binary classification problems, and 20newsgroups into a 7-class classifica-

[7] A sub-category (Ristin et al., 2015) is a *refined* category of a class label, e.g., a "flu" label contains three *sub-categories*: types A, B, and C flu.

[8] To gain more insight on the methods, we also reported all the metrics in terms of the surrogate loss in Appendix K.

Table 1: Experimental comparisons of the three methods w.r.t. the estimated ordinary risk and the estimated structural adversarial risk *using the 0-1 loss (%)*. The lower these values are, the better the performance of the method is. The KL divergence is used and distribution shift is assumed to be (a) class prior change and (b) sub-category prior change. Mean and standard deviation over 50 random train-test splits were reported. The best method and comparable ones based on the t-test at the significance level 1% are highlighted in boldface.

(a) Class prior change.

Dataset	Estimated ordinary risk			Estimated structural adversarial risk		
	ERM	AERM	Structural AERM	ERM	AERM	Structural AERM
blood	**22.4 (0.7)**	26.7 (5.0)	33.4 (2.0)	62.3 (2.4)	53.5 (10.6)	**36.7 (1.9)**
adult	**15.3 (0.2)**	15.4 (0.2)	18.7 (0.2)	30.4 (0.4)	30.4 (0.5)	**19.1 (0.3)**
fourclass	**24.0 (1.4)**	25.7 (2.6)	27.2 (1.4)	36.9 (2.2)	38.0 (6.2)	**29.5 (2.0)**
phishing	**6.1 (0.2)**	6.4 (0.2)	**6.0 (0.2)**	7.6 (0.4)	8.2 (0.4)	**6.5 (0.3)**
20news	**28.9 (0.4)**	30.6 (0.4)	34.7 (0.4)	44.1 (0.5)	45.0 (0.5)	**40.0 (0.6)**
satimage	**25.2 (0.3)**	30.8 (0.3)	32.2 (0.4)	**39.5 (0.6)**	47.9 (0.5)	**39.6 (0.6)**
letter	**14.3 (0.5)**	15.2 (0.6)	19.3 (0.8)	36.6 (1.5)	34.7 (1.7)	**22.5 (1.0)**
mnist	**10.0 (0.1)**	11.5 (0.1)	12.7 (0.1)	14.4 (0.2)	15.9 (0.2)	**13.8 (0.2)**

(b) Sub-category prior change.

Dataset	Estimated ordinary risk			Estimated structural adversarial risk		
	ERM	AERM	Structural AERM	ERM	AERM	Structural AERM
20news	**19.0 (0.3)**	20.6 (0.4)	23.3 (0.5)	35.6 (0.6)	37.8 (0.9)	**31.1 (0.5)**
satimage	**36.4 (0.3)**	44.2 (2.5)	40.7 (0.9)	62.1 (0.6)	66.2 (4.2)	**50.5 (0.6)**
letter	**17.5 (0.4)**	28.0 (5.4)	34.2 (2.0)	52.1 (0.6)	60.1 (5.8)	**43.0 (1.7)**
mnist	**13.3 (0.1)**	13.7 (0.1)	17.3 (0.2)	22.6 (0.2)	23.2 (0.2)	**19.9 (0.2)**

tion. Appendix J details how we grouped the class labels.

For all the methods, we used linear models with softmax output for the prediction function $g_\theta(x)$. The cross-entropy loss with ℓ_2 regularization was adopted. The regularization hyper-parameter λ was selected from $\{1.0, 0.1, 0.01, 0.001, 0.0001\}$ via 5-fold cross validation.

We used the two f-divergences (the KL and PE divergences) and set $\delta = 0.5$ for AERM and structural AERM. The same δ and f-divergence were used for estimating the structural adversarial risk. At the end of this section, we discuss how we can choose δ in practice.

Results: In Table 1, we report experimental results on the classification tasks when the KL divergence is used. Refer to Appendix L for the results when the PE divergence is used, which showed similar tendency.

We see from the left half of Table 1 that ordinary ERM achieved lower estimated risks as expected. On the other hand, we see from the entire Table 1 that AERM, which does not incorporate any structural assumptions on distribution shift, performed poorly in terms of both of two evaluation metrics; hence, it also performed poorly in terms of the adversarial risk (see Theorem 1). This may be because AERM was excessively sensitive to outliers as implied by Theorem 3. We see from the right half of Table 1 that structural AERM achieved significantly lower estimated structural adversarial risks. Although this was expected, our experiments confirmed that structural AERM indeed obtained classifiers robust against the *structural* distribution shift.[9]

Discussion: Here we provide an insight for users to determine δ in our DRSL (structural ARM and AERM). We see from Eq. (18) that the structural adversarial risk can be decomposed into the sum of the ordinary risk and the sensitivity term, where δ acts as a trade-off hyper-parameter between the two terms. In practice, users of our DRSL may want to have good balance between the two terms, i.e., the learned classifier should achieve high accuracy on the training distribution while being robust to specified distribution shift. Since both terms in Eq. (18) can be estimated by cross validation, the users can adjust δ of AERM at training time to best trade-off the two terms for their purposes, e.g., increasing δ during training to decrease the sensitivity term at the expense of a slight increase of the risk term.

7. Conclusion

In this paper, we theoretically analyzed DRSL with f-divergences applied to classification. We showed that the DRSL ends up giving a classifier optimal for the *training distribution*, which is too pessimistic in terms of the original motivation of distributionally robust classification. To rectify this, we presented simple DRSL that gives a robust classifier based on structural assumptions on distribution shift. We derived efficient optimization algorithms for our DRSL and empirically demonstrated its effectiveness.

[9] When we used the surrogate loss to evaluate the methods (which is not the case in ordinary classification), we confirmed that the methods indeed achieved the best performance in terms of the metrics they optimized for, i.e., ERM, AERM, and structural AERM performed the best in terms of the ordinary risk, adversarial risk and structural adversarial risk, respectively. See Appendix K for the actual experimental results.

Acknowledgement

We thank anonymous reviewers for their constructive feedback. WH was supported by JSPS KAKENHI 18J22289. MS was supported by CREST JPMJCR1403.

References

Bagnell, J. A. Robust supervised learning. In *Proceedings of Association for the Advancement of Artificial Intelligence*, volume 20, pp. 714–719, 2005.

Bartlett, P. L., Jordan, M. I., and McAuliffe, J. D. Convexity, classification, and risk bounds. *Journal of the American Statistical Association*, 101(473):138–156, 2006.

Ben-Tal, A., Den Hertog, D., De Waegenaere, A., Melenberg, B., and Rennen, G. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357, 2013.

Blake, C. and Merz, C. J. UCI repository of machine learning databases. http://archive.ics.uci.edu/ml/index.html, 1998.

Blanchet, J., Kang, Y., and Murthy, K. Robust wasserstein profile inference and applications to machine learning. *arXiv preprint arXiv:1610.05627*, 2016.

Bonnans, J. F. and Shapiro, A. Optimization problems with perturbations, a guided tour. *SIAM Review*, 40(2):228–264, 1998.

Chung, K. L. *A Course in Probability Theory*. Academic Press, 1968.

Ciszar, I. Information-type measures of difference of probability distributions and indirect observations. *Studia Sci. Math. Hungar.*, 2:299–318, 1967.

Danskin, J. M. The theory of max-min, with applications. *SIAM Journal on Applied Mathematics*, 14(4):641–664, 1966.

Duchi, J., Glynn, P., and Namkoong, H. Statistics of robust optimization: A generalized empirical likelihood approach. *arXiv preprint arXiv:1610.03425*, 2016.

Esfahani, P. M. and Kuhn, D. Data-driven distributionally robust optimization using the wasserstein metric: Performance guarantees and tractable reformulations. *arXiv preprint arXiv:1505.05116*, 2015.

Globerson, A. and Roweis, S. Nightmare at test time: robust learning by feature deletion. In *Proceedings of International Conference on Machine learning*, pp. 353–360, 2006.

Hashimoto, T. B., Srivastava, M., Namkoong, H., and Liang, P. Fairness without demographics in repeated loss minimization. In *Proceedings of International Conference on Machine learning*, 2018.

Lang, K. Newsweeder: Learning to filter netnews. In *Proceedings of International Conference on Machine Learning*, pp. 331–339, 1995.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, pp. 2278–2324, 1998.

Liu, A. and Ziebart, B. Robust classification under sample selection bias. In *Advances in Neural Information Processing Systems*, pp. 37–45, 2014.

Namkoong, H. and Duchi, J. C. Stochastic gradient methods for distributionally robust optimization with f-divergences. In *Advances in Neural Information Processing Systems*, pp. 2208–2216, 2016.

Namkoong, H. and Duchi, J. C. Variance-based regularization with convex objectives. In *Advances in Neural Information Processing Systems*, pp. 2975–2984, 2017.

Quionero-Candela, J., Sugiyama, M., Schwaighofer, A., and Lawrence, N. D. *Dataset shift in machine learning*. The MIT Press, 2009.

Ristin, M., Gall, J., Guillaumin, M., and Van Gool, L. From categories to subcategories: large-scale image classification with partial class label refinement. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 231–239, 2015.

Robinson, S. M. A characterization of stability in linear programming. *Operations Research*, 25:435–447, 1977.

Saerens, M., Latinne, P., and Decaestecker, C. Adjusting the outputs of a classifier to new a priori probabilities: a simple procedure. *Neural Computation*, 14(1):21–41, 2002.

Sinha, A., Namkoong, H., and Duchi, J. Certifiable distributional robustness with principled adversarial training. *arXiv preprint arXiv:1710.10571*, 2017.

Storkey, A. J. and Sugiyama, M. Mixture regression for covariate shift. In *Advances in Neural Information Processing Systems*, pp. 1337–1344, 2007.

Tewari, A. and Bartlett, P. L. On the consistency of multiclass classification methods. *Journal of Machine Learning Research*, 8(May):1007–1025, 2007.

Vapnik, V. N. *Statistical Learning Theory*. Wiley New York, 1998.