PAC-Bayesian Domain Adaptation Bounds for Multiclass Learners

Abstract

Multiclass neural networks are a common tool in modern unsupervised domain adaptation, yet an appropriate theoretical description for their non-uniform sample complexity is lacking in the adaptation literature. To fill this gap, we propose the first PAC-Bayesian adaptation bounds for multiclass learners. We facilitate practical use of our bounds by also proposing the first approximation techniques for the multiclass distribution divergences we consider. For divergences dependent on a Gibbs predictor, we propose additional PAC-Bayesian adaptation bounds which remove the need for inefficient Monte-Carlo estimation. Empirically, we test the efficacy of our proposed approximation techniques as well as some novel design-concepts which we include in our bounds. Finally, we apply our bounds to analyze a common adaptation algorithm that uses neural networks.

1 INTRODUCTION

Multiclass neural networks are frequently used in implementation of many unsupervised domain adaptation algorithms. For example, neural networks are often employed for invariant feature learning algorithms [Ganin and Lempitsky, 2015, Long et al., 2017, 2018, Zhang et al., 2019], importance weighting algorithms [Lipton et al., 2018], or combinations of both techniques [Tachet des Combes et al., 2020]. While most of these adaptation algorithms are motivated by theoretical bounds, recent literature has paid close attention to the assumptions and failure-cases of some techniques [Zhao et al., 2019, Wu et al., 2019, Johansson et al., 2019]. Namely, some learning algorithms *ignore* key terms in the adaptation bounds on which they are based, and as a result, may output solutions (i.e., learned models) that violate assumptions and are *guaranteed* to fail at the adaptation task [Zhao et al., 2019, Wu et al., 2019]. Still, the story here is not totally complete. In particular, there has not been much discussion of the non-uniform sample complexity of these modern adaptation algorithms. Sample complexity, in fact, contributes an additional "ignored" term in the theoretical bounds on which modern adaptation algorithms are based.

In this paper, we propose the first multiclass adaptation bounds which allow us to study this non-uniform sample complexity. Studying sample complexity is important to our understanding of adaptation algorithms because it describes how "data-hungry" an algorithm is. When this sample complexity is non-uniform across an algorithm's solution space, it allows us to study properties of a solution as a function of its "data-hunger." This is especially important for adaptation algorithms, which as mentioned, can inadvertently output poor solutions. Identifying a dynamic relationship between the properties of solutions and their non-uniform sample-complexity can provide insight on how to prevent these failure-cases in practice (e.g., by collecting sufficient data for an algorithm). Non-uniform sample complexity (rather than uniform complexity) can also help us to better quantify implicit regularization inherent to our algorithm [Dziugaite and Roy, 2017, Nagarajan and Kolter, 2019]. Accurately describing implicit regularization is especially important when using neural networks [Neyshabur et al., 2014, Neyshabur, 2017, Keskar et al., 2017, Zhang et al., 2017], since similar learning algorithms can lead to solutions with distinct generalization performance and implicit regularization is believed to be the cause of this phenomena.

Despite the importance of studying non-uniform sample complexity in modern adaptation contexts, we are not aware of any multiclass adaptation bounds with this capability. To fill this gap, we contribute the first PAC-Bayesian adaptation bound for multiclass learners (Thm. 2). While PAC-Bayesian bounds actually control error for *stochastic* models, we choose this framework for its demonstrated empirical accuracy in describing neural network sample complexity [Dziugaite and Roy, 2017, Zhou et al., 2018, Jiang et al., 2019, Dziugaite et al., 2020, 2021, Pérez-Ortiz et al., 2021].

Submitted to the 38th Conference on Uncertainty in Artificial Intelligence (UAI 2022). **To be used for reviewing only**.

Compared to existing bounds, we design our proposals to be more sensitive to the solution output by our learning algorithm as well as the data sample available for estimating key quantities. The former is vital in studying non-uniform complexity of adaptation algorithms (as discussed), while the latter is important for facilitating empirical study. To make our bounds useful in practice, we also propose the first approximation techniques for the divergence terms in our bound. In one case, this involves proposal of a novel surrogate for optimizing 01-loss (Thm. 4). In another, we show a standard technique for computing divergence fails to generalize to the mutliclass setting without additional constraints (Thm. 5). Working in the PAC-Bayesian framework, some divergences we study are also expressed as expectations with no known analytic solution. For these, we propose additional bounds (Thm. 6, Cor. 1) which allow us to avoid inefficient Monte-Carlo estimation by introduction of a new flatness assumption related to the well-known flat-minima hypothesis [Hochreiter and Schmidhuber, 1997]. To conclude, we conduct extensive empirical study of more than 12K models learned across 5 diverse adaptation datasets.

2 BACKGROUND

2.1 NOTATION AND ASSUMPTIONS

Consider the space $\mathcal{X} \times \mathcal{Y}$ for some finite $\mathcal{Y}$ with $|\mathcal{Y}| > 2$ unless otherwise noted. Colloquially, we call $\mathcal{X}$ the feature space and $\mathcal{Y}$ the label space. For a distribution $\mathbb{D}$ over $\mathcal{X} \times \mathcal{Y}$, we are interested in the risk functional $\mathbf{R}_{\mathbb{D}} : \mathcal{Y}^{\mathcal{X}} \to [0, 1]$

$$\mathbf{R}_{\mathbb{D}}(h) \stackrel{\text{def}}{=} \mathbf{Pr}(h(X) \neq Y); \qquad (X, Y) \sim \mathbb{D} \tag{1}$$

applied to some hypothesis (i.e., model) $h \in \mathcal{H} \subseteq \mathcal{Y}^{\mathcal{X}}$. The risk functional $\mathbf{R}_{\mathbb{D}}$ precisely gives the error rate of the hypothesis h when tasked with modelling the relationship between $\mathcal{X}$ and $\mathcal{Y}$ described by $\mathbb{D}$. In PAC-Bayes, we also consider the risk of stochastic (Gibbs) predictors. For a distribution $\mathbb{Q}$ over $\mathcal{H} \subseteq \mathcal{Y}^{\mathcal{X}}$, the Gibbs risk is the expectation

$$\mathbf{R}_{\mathbb{D}}(\mathbb{Q}) \stackrel{\text{def}}{=} \mathbf{E}[\mathbf{R}_{\mathbb{D}}(H)]; \qquad H \sim \mathbb{Q}. \tag{2}$$

For neural networks, a common stochastic formulation is to sample weights from the distribution $\mathbb{Q}$ before inference – e.g., the Bayesian neural networks of Blundell et al. [2015].

Throughout this paper, we assume a source distribution $\mathbb{S}$ over $\mathcal{X} \times \mathcal{Y}$ and a target distribution $\mathbb{T}$ over $\mathcal{X} \times \mathcal{Y}$. We assume observation of an i.i.d. random sample $S \sim \mathbb{S}^n$ and an i.i.d. random sample $T_X \sim \mathbb{T}_X^m$ where the subscript X denotes the $\mathcal{X}$-marginal of a distribution. In this context, an algorithm for the unsupervised adaptation problem we study is a function $(S, T_X) \mapsto h \in \mathcal{H}$. We are interested in bounds on $\mathbf{R}_{\mathbb{T}}(h)$ for such algorithms.

Interchangeably, we think of the sample S as both a random variable with distribution $\mathbb{S}^n$ and a (random) distribution itself, since any observation of a sample S uniquely defines its own empirical distribution over $\mathcal{X} \times \mathcal{Y}$ by the pmf

$$(x, y) \mapsto n^{-1} \sum\nolimits_{i=1}^{n} \mathbf{1}_{\{(x_i, y_i)\}}\{(x, y)\} \tag{3}$$

where $\mathbf{1}$ is the indicator function. So, $\mathbf{R}_S$ is well-defined by this identification. $\mathbf{R}_S(\mathbb{Q}) = \mathbf{E}_{H \sim \mathbb{Q}}[\mathbf{R}_S(H)]$ is also defined – the observation of S is used, not integrated out.

Finally, we also use distribution divergences based on the $\mathcal{H}$-divergence proposed by Ben-David et al. [2007]. This divergence is a specification of the $\mathcal{A}$-distance [Kifer et al., 2004] which relaxes the total variation distance by considering only a subset $\mathcal{A}$ of measurable sets when taking the supremum. In particular, the $\mathcal{H}$-divergence considers sets identifiable by a class $\mathcal{H} \subseteq \{0, 1\}^{\mathcal{X}}$

$$\mathbf{d}_{\mathcal{H}}(\mathbb{D}_1, \mathbb{D}_2) \stackrel{\text{def}}{=} \sup\nolimits_{h \in \mathcal{H}} \left| \mathbf{E}[h(X_1)] - \mathbf{E}[h(X_2)] \right| \tag{4}$$

where $X_i \sim \mathbb{D}_i$. While it is typically defined with a factor of 2, we omit this for convenience. Given a class $\mathcal{H} \subseteq \mathcal{Y}^{\mathcal{X}}$, we first study the $\mathcal{H}\Delta\mathcal{H}$-divergence based on the class

$$\mathcal{H}\Delta\mathcal{H} \stackrel{\text{def}}{=} \left\{ x \mapsto 1 - \mathbf{1}_{\{h(x)\}}\{h'(x)\} \mid (h, h') \in \mathcal{H}^2 \right\}. \tag{5}$$

This is a multiclass generalization, which simplifies to the original binary definition of Ben-David et al. when $|\mathcal{Y}| = 2$.

2.2 SOME EXISTING ADAPTATION BOUNDS

In this section, we discuss two adaptation bounds. More detailed knowledge of these bounds will be useful later for comparison with our proposed bounds. First, we discuss the seminal uniform convergence bound proposed by Ben-David et al. [2007, 2010]. Second, we discuss a PAC-Bayesian bound proposed by Germain et al. [2020].

2.2.1 Adaptation Based on Uniform Convergence

Theorem 1. *[Ben-David et al., 2010] Let $\mathcal{Y}$ be binary. For all $\delta > 0$, w.p. at least $1 - \delta$, for all $h \in \mathcal{H}$*

$$\begin{aligned} \mathbf{R}_{\mathbb{T}}(h) &\leq \lambda + \mathbf{R}_S(h) + \mathbf{d}_{\mathcal{H}\Delta\mathcal{H}}(S_X, T_X) \\ &+ 4\sqrt{\tfrac{4\nu \ln(2m) - \ln(\delta/4)}{m}} + 2\sqrt{\tfrac{8\nu \ln(em/\nu) - 2\ln(\delta/4)}{m}} \end{aligned} \tag{6}$$

where $\lambda = \min_{\eta \in \mathcal{H}} \mathbf{R}_{\mathbb{S}}(\eta) + \mathbf{R}_{\mathbb{T}}(\eta)$ and $\nu = \mathrm{VCDim}(\mathcal{H})$.

The seminal result above is the standard adaptation bound on which many newer results are based. Still, this uniform convergence bound is not well-suited for every application. We discuss some limitations below.

Uniform Sample Complexity Simply put, uniform convergence is too conservative: it assigns the same sample complexity to each outcome of our learning algorithm, regardless of the solution quality. As discussed in Section 1, this prevents us from studying important properties of a model as a function of its sample complexity.

Model-Independent Divergence In general, divergence is meant to characterize the similarity in feature distributions under the source $\mathbb{S}$ and the target $\mathbb{T}$. Similar to above, independence of the divergence $\mathbf{d}_{\mathcal{H}\Delta\mathcal{H}}$ and the model h is overly conservative and makes this term insensitive to changes in the outcome of our learning algorithm. For example, when $\mathcal{H}$ is fixed, this divergence cannot distinguish between a random initialization and a carefully trained solution.

Sample-Independent Adaptability The term λ is often called the *adaptability*. It is a measure of similarity in the labeling functions of $\mathbb{S}$ and $\mathbb{T}$, characterizing the extent to which one hypothesis in $\mathcal{H}$ can do well on *both* of these distributions. When no such hypothesis exists, it is unclear how a learner could successfully adapt by minimizing risk on the source distribution [Ben-David et al., 2010]. Importantly, this term has been central to the theoretical discussion of failure-cases in widely used DA algorithms [Johansson et al., 2019, Zhao et al., 2019]. Meanwhile, estimation of λ remains an under-studied research area [Redko et al., 2020]. One problem, which we observe, is independence of λ from the samples S and T. In particular, one cannot directly compute the population statistic λ in typical circumstances. Instead, one might estimate using $\min_\eta \mathbf{R}_S(\eta) + \mathbf{R}_T(\eta)$, but this requires verifying generalization of a learned model $h^* \in \arg\min_\eta \mathbf{R}_S(\eta) + \mathbf{R}_T(\eta)$ using a holdout set or some other descriptor of generalization performance (e.g., such as a learning bound). This is undesirable when, as in this paper, we wish to study adaptability in an empirical context. As we show in later experiments (Section 4), the extra generalization requirement typically inflates our estimation of λ, and subsequently, mars the results we would like to interpret.

Binary Label Space It is also important to note that this bound was designed for binary learners. Computation of the $\mathcal{H}\Delta\mathcal{H}$-divergence is the most concerning issue, since existing algorithms for computation rely on symmetry of $\mathcal{H}$ and ERM over the class $\mathcal{H}\Delta\mathcal{H}$. In Section 3.2, we discuss these issues in detail and present some solutions.

2.2.2 A PAC-Bayesian Bound for Binary Learners

In this section, we consider Thm. 7 in Appendix A.2, which is a result of Germain et al. [2020]. While Germain et al. [2013, 2016, 2020] propose other adaptation bounds, we focus on this bound because it is easiest to compare to the proposal of Ben-David et al. [2010]. Our discussion below is also applicable to the other bounds of Germain et al.

Benefits Compared to Thm. 1 One benefit of Thm. 7 is that the divergence employed in this bound is **model-dependent** (rather than independent); namely, it depends on the Gibbs predictor $\mathbb{Q}$, whose target error we bound. As mentioned, model-independence is an overly conservative quality and Germain et al. [2020] show this formally by proving their divergence actually lowerbounds $\mathbf{d}_{\mathcal{H}\Delta\mathcal{H}}$ for all $\mathbb{Q}$ and $\mathcal{H}$. Another primary benefit is that Thm. 7 employs a **non-uniform** sample complexity. Specifically, complexity is measured through a KL-divergence $\mathrm{KL}(\mathbb{Q} \mid\mid \mathbb{P})$, which explicitly depends on the outcome of the learning algorithm $\mathbb{Q}$. Simply put, a model is complex if it deviates much from our prior knowledge, which is captured in the prior $\mathbb{P}$.

Weaknesses Shared with Thm. 1 Despite its benefits over Thm. 1, Thm. 7 also shares some weaknesses. First, the proposed adaptability term is also **sample-independent**. Second, the bound is still designed for a **binary label space** $\mathcal{Y}$. Unlike the case of Thm. 1, it is not computation of the bound that causes concern, but the *validity* of the bound in mutliclass settings. In particular, the problem arises because the proof of Thm. 7 relies on a decomposition of the risk which assumes $|\mathcal{Y}| = 2$. This decomposition does not hold, in general, when $\mathcal{Y}$ is larger. In fact, Germain et al. [2020], themselves, observe Thm. 7 is not easily extended to multiclass settings, leaving the investigation of such PAC-Bayes bounds as an open problem. For some additional empirical study of Thm. 7, see Appendix D.3.

2.3 OTHER RELATED WORKS

Besides those works discussed above, there are some additional works, which propose alternate theories of adaptation. Some theories of adaptation use distinct integral probability metrics in place of the $\mathcal{H}$-divergence [Redko et al., 2017, Shen et al., 2018, Johansson et al., 2019], while others have sought to generalize and modify the $\mathcal{H}$-divergence [Mansour et al., 2009, Kuroki et al., 2019, Zhang et al., 2019]. Meanwhile, others focus on assumptions distinct from small adaptability. These include covariate shift [Sugiyama et al., 2007, You et al., 2019], label shift [Lipton et al., 2018] and *generalized* label shift [Tachet des Combes et al., 2020]. The DA problem can also be modeled through causal graphs [Zhang et al., 2015, Magliacane et al., 2018] and some extensions to DA consider a meta-distribution over targets [Blanchard et al., 2021, Albuquerque et al., 2020, Deng et al., 2020]. Notably, most assumptions are untestable in practice and not many works consider such testing, even in controlled research settings where it might be possible. As we are aware, we are the first to use a sample-dependent adaptability, which improves estimation in empirical study.

In adaptation, PAC-Bayesian results are almost exclusively due to Germain et al. [2013, 2016, 2020]. Albeit, in transfer learning some work does exist [Li and Bilmes, 2007, McNamara and Balcan, 2017]. Most directly, our work employs the PAC-Bayes bound of Maurer [2004] in proofs as well as some techniques of Langford and Caruana [2001], Dziugaite and Roy [2017], and Pérez-Ortiz et al. [2021] in empirical study. Most notably, ours is the only PAC-Bayesian work to propose multiclass adaptation bounds. A more in depth coverage of relevant literature – for adaptation and PAC-Bayes – is available in Appendix B.

3 PROPOSED BOUNDS

In this section, we give the proposed adaptation bounds for multiclass learners. We also provide novel algorithms for computing two multiclass divergence terms and compare these to existing approaches. Lastly, we give a second adaptation bound which removes the need for inefficient Monte-Carlo estimation of divergences dependent on a Gibbs predictor. Proof of all results is given in Appendix A.

3.1 A PAC-BAYESIAN ADAPTATION BOUND FOR MULTICLASS LEARNERS

We begin by introducing a model-dependent class of hypotheses similar to $\mathcal{H}\Delta\mathcal{H}$. Precisely, for $h \in \mathcal{H}$,

$$h\Delta\mathcal{H} \stackrel{\text{def}}{=} \left\{x \mapsto 1 - \mathbf{1}_{\{h(x)\}}\{h'(x)\} \mid h' \in \mathcal{H}\right\}. \tag{7}$$

With it, we propose to use the $h\Delta\mathcal{H}$-divergence $\mathbf{d}_{h\Delta\mathcal{H}}$ in our adaptation bounds. This divergence is a **model-dependent** extension of the $\mathcal{H}\Delta\mathcal{H}$-divergence, applicable in multiclass settings. It is easy to observe from the definitions that this new divergence lowerbounds the $\mathcal{H}\Delta\mathcal{H}$-divergence for all $\mathcal{H}$ and all h. Both Zhang et al. [2019] and Kuroki et al. [2019] study similar divergences for bounding 01-loss in the binary case,[1] but we are first to use this divergence with non-uniform sample complexity, and also, the first to use this divergence in an adaptation bound for multiclass learners. Our full proposal requires novel technique and theoretical study to compute this divergence (see Section 3.2).

Theorem 2. *For any $\mathbb{P}$ over $\mathcal{H}$, all $\delta > 0$, w.p. at least $1-\delta$, for all $\mathbb{Q}$ over $\mathcal{H}$*

$$\begin{aligned}\mathbf{R}_{\mathbb{T}}(\mathbb{Q}) \leq \tilde{\lambda}_{S,T} + \mathbf{R}_S(\mathbb{Q}) + \mathbf{E}_{H\sim\mathbb{Q}}[\mathbf{d}_{\mathcal{C}_H}(S_X, T_X)] \\ + \sqrt{\tfrac{\mathrm{KL}(\mathbb{Q}\|\mathbb{P}) + \ln\sqrt{4m} - \ln(\delta)}{2m}}\end{aligned} \tag{8}$$

where $\tilde{\lambda}_{S,T} = \min_{\eta\in\mathcal{H}} \mathbf{R}_S(\eta) + \mathbf{R}_T(\eta)$ and we may choose either $\mathcal{C}_h = \mathcal{H}\Delta\mathcal{H}$ for all h as before or $\mathcal{C}_h = h\Delta\mathcal{H}$.

Comparison to Thms. 1 and 7 By design, the bound proposed above resolves the weaknesses mentioned in Section 2.2. First and foremost, we remove that requirement that $\mathcal{Y}$ is binary. Second, we use a **non-uniform** notion of sample complexity; i.e., $\mathrm{KL}(\mathbb{Q} \,||\, \mathbb{P})$. Third, Thm. 2 allows for *either* a **model-dependent** or **model-independent** notion of data-distribution divergence. While model-independent divergences do have some weaknesses, we retain them in our bound since, as discussed later, they can be more efficient. Lastly, Thm. 2 employs a **sample-dependent** notion of adaptability. Compared to λ, the new adaptability $\tilde{\lambda}_{S,T}$ is the smallest error achievable on the *samples*. In research contexts wherein we assume access to target labels for purpose of studying our assumptions, we later show that this quantity is fairly easy to empirically bound.

[1] We discuss a multiclass proposal of Zhang et al. [2019] later, but it is based on margin loss and used for uniform convergence.

3.2 APPROXIMATING MULTICLASS DIVERGENCE

3.2.1 The Multiclass $\mathcal{H}\Delta\mathcal{H}$-divergence

First, since the $\mathcal{H}\Delta\mathcal{H}$-divergence is model-independent, the expectation with respect to the Gibbs predictor $\mathbb{Q}$ simplifies significantly. In particular, if $\mathcal{C}_h = \mathcal{H}\Delta\mathcal{H}$, we have

$$\mathbf{E}_{H\sim\mathbb{Q}}[\mathbf{d}_{\mathcal{C}_H}(S_X, T_X)] = \mathbf{d}_{\mathcal{H}\Delta\mathcal{H}}(S_X, T_X). \tag{9}$$

Thus, computation of this divergence simplifies to computing the $\mathcal{H}\Delta\mathcal{H}$-divergence for multiclass learners.

Summary of Approach In general, we take inspiration from the proposal of Ben-David et al. [2010] who compute $\mathcal{H}\Delta\mathcal{H}$-divergence when models in $\mathcal{H}$ have binary output. Namely, we frame computation as minimization of error in a specific classification problem. To adapt this strategy to the multiclass setting, we do two primary things. First, we remove the assumption that $\mathcal{H}$ is symmetric. This is important for multiclass settings since we have no reason to believe $\mathcal{H}\Delta\mathcal{H}$ is typically symmetric. We replace the symmetry in $\mathcal{H}\Delta\mathcal{H}$ with a symmetry in our classification problems. Second, for score-based classifiers such as neural networks, we give an optimization procedure for approximating ERM over this class based on a surrogate loss function. As we are aware, this is the first algorithm for approximating ERM over $\mathcal{H}\Delta\mathcal{H}$ when models in $\mathcal{H}$ have multiclass output.

Reduction to ERM

Theorem 3. *Let $\mathcal{C} = \mathcal{H}\Delta\mathcal{H}$. Almost surely,*

$$\mathbf{d}_{\mathcal{C}}(S_X, T_X) = \max\left\{\begin{matrix} 1 - \min_{\varphi\in\mathcal{C}} \mathbf{R}_P(\varphi) + \mathbf{R}_Q(\varphi), \\ 1 - \min_{\varphi\in\mathcal{C}} \mathbf{R}_U(\varphi) + \mathbf{R}_V(\varphi) \end{matrix}\right\} \tag{10}$$

where

$$\begin{aligned} P = \big((X_i, 1) \mid X_i \in S_X\big),\ Q = \big((\tilde{X}_i, 0) \mid \tilde{X}_i \in T_X\big), \\ U = \big((X_i, 0) \mid X_i \in S_X\big),\ V = \big((\tilde{X}_i, 1) \mid \tilde{X}_i \in T_X\big). \end{aligned} \tag{11}$$

Notice, pooled samples $P +_{\mathrm{c}} Q$ and $U +_{\mathrm{c}} V$ define binary classification problems ($+_{\mathrm{c}}$ is concatenation). Namely, they represent an identification problem wherein the learner must distinguish between the samples S_X and T_X. To compute divergence as above, we need only select φ to minimize the sum of class-conditional error rates for these problems. Even in simple cases, risk-minimization can be computationally hard [Shalev-Shwartz and Ben-David, 2014]. Thus, we instead select φ by optimizing a surrogate loss.

Approximate Minimization via Surrogate WLOG, $\mathcal{Y} = \{1, \ldots, C\}$. We consider a score-based class $\mathcal{S}$ written

$$\mathcal{S} \stackrel{\text{def}}{=} \left\{\Psi_{\mathbf{f}} \mid \mathbf{f} \in \mathcal{F}\right\}; \quad \Psi_{\mathbf{f}}(x) \stackrel{\text{def}}{=} \arg\max_{\ell\in[C]} \mathbf{f}_\ell(x) \tag{12}$$

with $\mathcal{F} \subseteq \{\mathbf{f} \mid \mathbf{f}_\ell : \mathcal{X} \to \mathbb{R},\ \ell \in [C]\}$ a class of scoring-functions. In case of ties, suppose $\arg\max$ returns the least

label. Using the naïve definition in Eq. (5),

$$\mathcal{S}\Delta\mathcal{S} \stackrel{\text{def}}{=} \{x \mapsto 1 - \mathbf{1}_{\{\Psi_{\mathbf{f}}(x)\}}\{\Psi_{\mathbf{g}}(x)\} \mid (\mathbf{f}, \mathbf{g}) \in \mathcal{F}^2\}. \quad (13)$$

At first glance, it is unclear how to pick $\varphi \in \mathcal{S}\Delta\mathcal{S}$ to minimize error on a given dataset. So, in place of this obscure definition, the following result gives a surrogate loss which upperbounds the 01-loss on the original problem. Thus, we indirectly reduce the error by minimizing the surrogate.

Theorem 4. *Suppose $\tau : \mathbb{R} \to \mathbb{R}_{\geq 0}$ is differentiable and monotone increasing. Let $\mathbf{A} = \big(\tau \circ \mathbf{g}(x)\big) \cdot \big(\tau \circ \mathbf{f}(x)^{\mathrm{T}}\big)$ with τ applied element-wise and $\mathbf{f}, \mathbf{g} \in \mathcal{F}$. Set*

$$\begin{aligned} z(x) &\stackrel{\text{def}}{=} \max_{(j,k)\in[C]^2} \mathbf{A}_{jk} - \max_{i\in[C]} \mathbf{A}_{ii}, \\ \mathcal{L}(z, y) &\stackrel{\text{def}}{=} \ln(1 + \exp(-(2y-1)\cdot z))/\ln(2). \end{aligned} \quad (14)$$

Then, if $\varphi(x) = 1 - \mathbf{1}_{\{\Psi_{\mathbf{f}}(x)\}}\{\Psi_{\mathbf{g}}(x)\}$, we have

$$\mathbf{R}_{\mathbb{D}}(\varphi) \leq \mathbf{E}_{(X,Y)\sim\mathbb{D}}\, \mathcal{L}(z(X), Y) \quad (15)$$

for any distribution $\mathbb{D}$ s.t. $\mathbf{f}(X)$ has no repeated scores and $\mathbf{g}(X)$ has no repeated scores, almost surely.[2]

We point out the log loss $\mathcal{L}(z, y)$ is differentiable with respect to z and $z(x)$ is differentiable with respect to $\mathbf{f}$ and $\mathbf{g}$. In practice, functions in $\mathcal{F}$ – such as $\mathbf{f}$ and $\mathbf{g}$ – are typically differentiable with respect to a real-parameter vector, which also defines the function. For example, this is precisely the case for neural networks. In these contexts, since composition preserves differentiability, the output of the surrogate $\mathcal{L}$ is differentiable with respect to the real-parameter vector. So, the RHS of Eq. (15) may be approximately minimized using batch SGD. At this point, the proposed algorithm should be familiar to the typical practioner. It is equivalent to the manner in which we usually optimize a neural network, except for the new surrogate $(\mathbf{f}, \mathbf{g}, x, y) \mapsto \mathcal{L}(z(x), y)$.

3.2.2 The Multiclass $h\Delta\mathcal{H}$-divergence

When $\mathcal{C}_h = h\Delta\mathcal{H}$ the divergence term is model-dependent and the expectation with respect to the Gibbs predictor $\mathbb{Q}$ becomes a challenge. For neural networks, even the Gibbs risk $\mathbf{R}_S(\mathbb{Q})$ does not have a known analytic solution. Instead, it is common to approximate using Monte-Carlo sampling [Langford and Caruana, 2001, Dziugaite and Roy, 2017, Dziugaite et al., 2021, Pérez-Ortiz et al., 2021]. By Hoeffding's Inequality, w.p. at least $1 - \delta$, we approximate

$$\mathop{\mathbf{E}}_{H\sim\mathbb{Q}}[\mathbf{d}_{\mathcal{C}_H}(S_X, T_X)] \leq \frac{1}{k}\sum_{i=1}^{k} \mathbf{d}_{\mathcal{C}_{H_i}}(S_X, T_X) + \sqrt{\tfrac{\ln 2/\delta}{2k}} \quad (16)$$

where $(H_i)_{i=1}^k \sim \mathbb{Q}^k$. Using the RHS as an approximation, our computation reduces to computing $\mathbf{d}_{\mathcal{C}_h}$ for any deterministic $h \in \mathcal{H}$. Upon sampling from $\mathbb{Q}$, we can apply the algorithm for computing $\mathbf{d}_{\mathcal{C}_h}$ to each point in the sample. In light of this, the next part focuses on computing $\mathbf{d}_{\mathcal{C}_h}$ for deterministic h. We proceed as before, reducing computation to risk-minimization for a specific classification problem.

Reduction to ERM with Constrained Labeling Function

Theorem 5. *Almost surely, for all $h \in \mathcal{H}$*

$$\mathbf{d}_{\mathcal{C}_h}(S_X, T_X) = \max \left\{ \begin{array}{l} 1 - \min\limits_{\substack{\varphi\in\mathcal{H},\\ \bar{h}\in\Upsilon}} \mathbf{R}_P(\varphi) + \mathbf{R}_Q(\varphi), \\ 1 - \min\limits_{\substack{\varphi\in\mathcal{H},\\ \bar{h}\in\Upsilon}} \mathbf{R}_U(\varphi) + \mathbf{R}_V(\varphi) \end{array} \right\} \quad (17)$$

where $\mathcal{C}_h = h\Delta\mathcal{H}$ and

$$\begin{aligned} P(\bar{h}) &= \big((X_i, \bar{h}(X_i)) \mid X_i \in S_X\big), \\ Q &= \big((\tilde{X}_i, h(\tilde{X}_i)) \mid \tilde{X}_i \in T_X\big), \\ U &= \big((X_i, h(X_i)) \mid X_i \in S_X\big), \\ V(\bar{h}) &= \big((\tilde{X}_i, \bar{h}(\tilde{X}_i)) \mid \tilde{X}_i \in T_X\big) \end{aligned} \quad (18)$$

and $\Upsilon = \{\bar{h} \in \mathcal{Y}^{\mathcal{X}} \mid \bar{h}(x) \neq h(x),\ \forall x \in \mathcal{X}\}$.

As before, the result describes two classification problems. This time, the learner's goal is to agree with h on one sample, while disagreeing with h (in the way specified by $\bar{h}$) on the other sample. We minimize the class-conditional error rates by selecting from the class $\mathcal{H}$ used for the original prediction task, rather than $\mathcal{H}\Delta\mathcal{H}$. So, we make the obvious proposal: re-use whichever approximation technique we used to select h in the first place. In our experiments, since $\mathcal{H}$ is a space of neural networks, we use batch SGD on an NLL loss.

A Heuristic for Selecting from Υ Reduction to ERM in the multiclass setting also requires specification of $\bar{h} \in \Upsilon$. Specifically, $\bar{h}$ should aid in minimizing the class-conditional error rates. In our experiments, we found a simple strategy to be fairly effective. Namely, we specify $\bar{h}$ by always picking the label with the second-highest confidence according to h. So, $\bar{h}$ disagrees with h on all of $\mathcal{X}$, but does so in the "most reasonable" way according to the probabilities output by h. This approach uses the probabilities output by h to rank similarity of labeling functions and supposes the most "similiar" labeling function in Υ will be easiest for another hypothesis in $\mathcal{H}$ to simultaneously learn. Mathematically, our solution satisfies

$$\bar{h} \in \arg\max_{\upsilon\in\Upsilon} \sum\nolimits_{x\in\mathcal{X}} h_{\upsilon(x)}(x) \quad (19)$$

where $h_\ell(x)$ is the score assigned to the label ℓ. This heuristic is a practical solution that avoids search over Υ, which will typically be unknown, unless we inefficiently enumerate using membership constraints. As we are aware, there is no known algorithm to efficiently select minimizers from Υ and $\mathcal{H}$, simultaneously, as called for by Thm. 5. Besides the heuristic, we leave this problem as future work.

Comparison to Some Related Approaches Considering a binary label space $\mathcal{Y}$, Kuroki et al. [2019] propose a similar

[2] This stipulation on $\mathbb{D}$ is not too strict. It only assumes ties in the scores of $\mathbf{f}$ or $\mathbf{g}$ are *very* unlikely, so these ties can be ignored.

algorithm. The multiclass setting we consider does require some differences, primarily, related to the distinct degrees of freedom in multiclass and binary classification. First, our proposal removes the requirement that $\mathcal{H}$ is symmetric since, as we are aware, this concept is not well-defined for hypotheses with multiclass output. Similar to before, we replace the symmetry required of $\mathcal{H}$ with symmetry in our classification problems. Second, in multiclass settings, the reduction strategy necessitates a new parameter to optimize: the labeling function $\bar{h}$. Besides our proposed heuristic for optimization, proof of this fact is *not* a straightforward extension of the work of Kuroki et al. [2019]. In fact, it requires a different proof-technique (see Appendix A.6). In a multiclass setting, Zhang et al. [2019] also propose an approach for approximating a mutliclass divergence. In general, our two techniques require different consideration because their divergence is based on margin loss, rather than 01-loss. Notably, the multiclass bounds of Zhang et al. [2019] use uniform sample complexity, unlike our proposed non-uniform approach.

3.3 EFFICIENCY THROUGH FLAT-MINIMA

In full view of Section 3.2.2, the reader may rightfully be concerned about the efficiency of the proposed technique for approximating $\mathbf{E}_H[\mathbf{d}_{\mathcal{C}_H}(\cdot,\cdot)]$. In particular, the suggestion requires training k distinct neural networks: one for each $H_i \sim \mathbb{Q}$. Typically, k will be large – e.g., larger than 100 – to control the size of the upperbound in Eq. (16), so this is not computationally feasible for practical applications. This problem is not totally unique to the model-dependent $h\Delta\mathcal{H}$-divergence, either. Common invariant feature-learning algorithms – e.g., DANN [Ganin and Lempitsky, 2015] – actually modify the feature distribution over which the classifier learns. In these cases, even the $\mathcal{H}\Delta\mathcal{H}$-divergence becomes dependent on the model [Johansson et al., 2019]. In particular, supposing every model $h \in \mathcal{H}$ is the composition $c_h \circ f_h$ of a classifier c_h and a feature extractor f_h, the modified $\mathcal{H}\Delta\mathcal{H}$-divergence results from the following restriction

$$[\mathcal{H}\Delta\mathcal{H}]_h \stackrel{\text{def}}{=} \{1-\mathbf{1}_{\{c_p\circ f_h(\cdot)\}}\{c_q\circ f_h(\cdot)\} \mid (p,q)\in\mathcal{H}^2\}. \quad (20)$$

A similar restriction can be defined for the class $h\Delta\mathcal{H}$

$$[h\Delta\mathcal{H}]_h \stackrel{\text{def}}{=} \{1-\mathbf{1}_{\{c_h\circ f_h(\cdot)\}}\{c_q\circ f_h(\cdot)\} \mid q\in\mathcal{H}\}. \quad (21)$$

In both cases, due to the dependence on h, the expectation over $\mathbb{Q}$ cannot be avoided as in Section 3.2.1. To resolve this frequent issue, we propose a new adaptation bound, which relies on an assumption related to flatness of the 01-loss over a (weighted) region in parameter space defined by $\mathbb{Q}$. Flatness assumptions are not unusual in PAC-Bayes and we develop this connection next.

3.3.1 Flat-Minima and PAC-Bayes

An SGD solution lies in a flat-minimum if its parameters are robust to perturbation: changing the parameters (slightly) does not change the trained network's already low error rate. So, flatness here is an absence of "elevation" in error as our model moves about some region of parameter space. Hochreiter and Schmidhuber [1997] first discussed "flatness" as it relates to neural network generalization, hypothesizing that models lying in a large flat-minimum generalize well. More recently, the idea has been validated empirically at large scale. In particular, notions of the sharpness of minima are often good empirical descriptors of an SGD-trained neural network's generalization performance [Jiang et al., 2019, Dziugaite et al., 2020]. The motivation for using PAC-Bayes bounds is very often based on the hypothesis that flat-minima generalize well because PAC-Bayes bounds implicitly encode the existence of flat-minima [Neyshabur et al., 2017, Dziugaite and Roy, 2017]. For unimodal $\mathbb{Q}$, a region of non-zero probability around the mode necessitates models in this region have identically low error (form a flat-minimum). Otherwise, the Gibbs risk would be large. As the prior $\mathbb{P}$ has some variance, we know these regions should exist near $\mathbb{Q}$, or else $\mathrm{KL}(\mathbb{Q} \mid\mid \mathbb{P})$ would be large. In this sense, PAC-Bayes bounds and flat-minima go hand-in-hand. If the former is small, we know the latter exists.

3.3.2 A More Efficient Adaptation Bound

Definition 1. *A Gibbs predictor $\mathbb{Q}$ is ρ-flat on the distribution $\mathbb{D}$ if $|\mathbf{E}_{H\sim\mathbb{Q}}\,\mathbf{R}_{\mathbb{D}}(H) - \mathbf{R}_{\mathbb{D}}(\mathbf{E}_{H\sim\mathbb{Q}}\,H)| \le \rho$.*

Often, as in the definition above, we will refer to the "flatness" of a Gibbs predictor $\mathbb{Q}$ when it would be more precise to refer to the flatness of the (weighted) region in parameter space that this predictor defines. In this sense, the above definition quantifies the flatness of a region in parameter space by the ability of the error in this region to be represented by a single hypothesis from that region. Intuitively, this echoes physical properties of flatness: a topographic map requires many more numbers to describe a mountainous terrain than a flat prairie. Next, we give the proposed bound.

Theorem 6. *For any $\mathbb{P}$ over $\mathcal{H}$, all $\delta > 0$, w.p. at least $1-\delta$, for all $\mathbb{Q}$ over $\mathcal{H}$ s.t. $\mathbb{Q}$ is ρ_S-flat on S and ρ_T-flat on T*

$$\begin{aligned}\mathbf{R}_{\mathbb{T}}(\mathbb{Q}) \le \rho + \tilde{\lambda}_{S,T} + \mathbf{R}_S(\mathbb{Q}) + \mathbf{d}_{\mathcal{C}_\mu}(S_X, T_X) \\ + \sqrt{\tfrac{\mathrm{KL}(\mathbb{Q}||\mathbb{P}) + \ln\sqrt{4m} - \ln(\delta)}{2m}}\end{aligned} \quad (22)$$

where $\mu = \mathbf{E}_{H\sim\mathbb{Q}}\,H$, $\rho = \rho_S + \rho_T$, and $\mathcal{C}_\mu = \mu\Delta\mathcal{H}$.

Corollary 1. *To study algorithms like DANN, we can instead choose $\mathcal{C}_\mu = [\mathcal{H}\Delta\mathcal{H}]_\mu$ or $\mathcal{C}_\mu = [\mu\Delta\mathcal{H}]_\mu$ in Thm. 6. The adaptability $\tilde{\lambda}_{S,T}$ is then dependent on μ as below*

$$\tilde{\lambda}^{\mu}_{S,T} = \min_{g\in\mathcal{H}} \mathbf{R}_S(c_g\circ f_\mu) + \mathbf{R}_T(c_g\circ f_\mu). \quad (23)$$

The main bound is identical to Thm. 2 except that we assume $\mathbb{Q}$ is flat on both S and T, then use this assumption to introduce a deterministic reference μ in the divergence.

This deterministic reference replaces the expectation over $\mathbb{Q}$ whose estimation was inefficient, but the new cost is inflation of the bound by ρ. Unfortunately, similar to adaptability terms, we cannot expect to compute ρ outside of controlled research contexts, since labels are required to estimate the flatness of $\mathbb{Q}$ on T (according to Def. 1). Instead, for the bound to be practically useful, we propose to assume ρ is small. This, for example, is often the suggestion when it comes to adaptability as well. Albeit, the caveats of carelessly making assumptions on adaptation problems should be noted [Zhao et al., 2019, Johansson et al., 2019].

We argue the assumption of small ρ is not an overly strong (or careless) assumption to make. To begin with, PAC-Bayes bounds and flat-minima are already related. The only addition we make to the usual connection (see Section 3.3.1) is that flat regions *remain flat* when we transfer across data distributions (or, samples). Note, we do not even require the transferred region to remain a minimum, since the size of ρ is only dictated by the *difference* in the Gibbs risk and the reference risk: the Gibbs risk can be high on T as long as the reference risk is as well. Thus, if one is willing to accept the usual assumptions, then our additional assumption merely begs the question: *Do flat regions transfer?*

In the next section, empirically, we test this question along with the other proposals given in this text.

4 EXPERIMENTS

4.1 SETUP

Datasets We use a wide-array of datasets from vision and NLP: **Digits** [Ganin and Lempitsky, 2015], **PACS** [Li et al., 2017], **Office-Home** [Venkateswara et al., 2017], **Amazon Reviews** [Blitzer et al., 2007], and **Discourse** sense classification datasets [Prasad et al., 2008, Ramesh and Yu, 2010, Zeyrek et al., 2020]. For **Digits**, we use the image as feature, while for **Amazon Reviews**, we use uni-gram and bi-gram features. For other datasets, we use pre-trained ResNet-50 [He et al., 2016] or BERT [Devlin et al.] features.

Models For **Digits**, we use a 4-layer CNN. For all other datasets, we use both a linear model and a 4-layer fully-connected network. For simplicity, our larger scale experiments use a simple adaptation algorithm (**SA**) which optimizes models to minimize risk on S. On **Digits**, we also study the **DANN** algorithm proposed by Ganin and Lempitsky [2015], modified to train Gibbs predictors with varied regularization using **PBB** [Pérez-Ortiz et al., 2021]. We pick **Digits**, specifically, because it exhibits shift in the marginal label distributions, which can cause **DANN** to fail [Zhao et al., 2019]. More training details are given in Appendix C.

Experiments The data points in our results are each individual experiments done on a source dataset S and target dataset T using a classifier h or Gibbs predictor $\mathbb{Q}$. The pair S and T are taken from a set of data splits using the datasets discussed above (details in Appendix C). Across these splits, we consider various scenarios including: **single-source**, **multi-source**, and **within-distribution** adaptation (i.e., $\mathbb{S} = \mathbb{T}$) using multiple random data splits. On **Digits**, we also consider **natural shifts** (i.e., noise and rotation) and **unnatural shifts** (i.e., transfer to random data). In general, we restrict the pair (S, T) to have a common label space.

Table 1: Correlation of h-dependent and -independent divergence with $|\mathbf{R}_h(S) - \mathbf{R}_h(T)|$. Columns delineate data subsets. Each datum describes unique (S, T, h). See Appendix C.6 for details.

	All	**Digi.**	**Disc.**	**PACS+OH**	**Amaz.**
model-ind.	0.54	0.15	0.70	0.41	-0.05
model-dep.	0.58	0.23	0.78	0.14	0.41

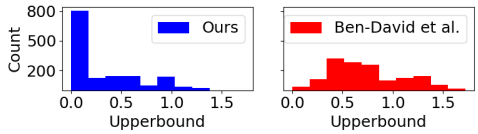

Figure 1: Sample-dependent (left) and independent adaptability. Each datum is for unique $(S, T, \mathcal{H})$. See Appendix C.5 for details.

4.2 RESULTS

Sample-Dependent Adaptability As mentioned, estimation of sample-independent adaptability (e.g., λ in Thm. 1) requires verification of generalization. In particular, to estimate λ one can learn $\eta \in \mathcal{H}$ which has small sum of risks over the observed samples S and T. In our experiments, we do so using batch SGD on a weighted NLL loss – a common surrogate. Because λ is a population statistic for $\mathbb{S}$ and $\mathbb{T}$, we cannot directly report the errors on S and T – this is incorrect, like using training error as a validation

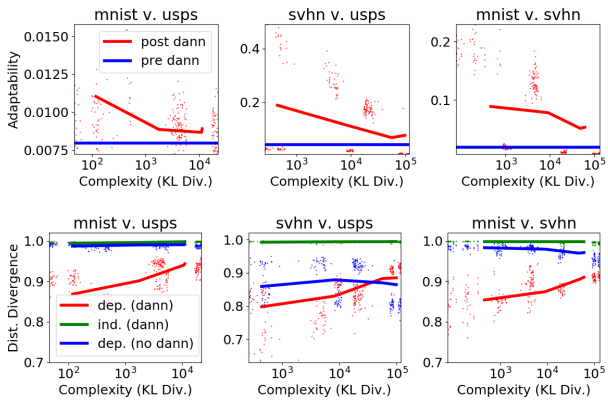

Figure 2: Adaptability and dependent/independent divergences for **DANN** on **Digits**. Solid line is median. Scatter describes unique $(S, T, \mathbb{Q})$, limited to 95% or more data to filter extreme values.

metric. Instead, we should check the performance of η on a heldout data subset (for example). This is the strategy we take in Figure 1, using Hoeffding's Inequality to produce a valid upperbound on λ. Comparably, estimating sample-dependent adaptability is much easier. By design, we *can* report error on the samples S and T used for training η. Doing so, produces a valid upperbound:

$$\forall \eta \in \mathcal{H} \ : \ \tilde{\lambda}_{S,T} \leq \mathbf{R}_S(\eta) + \mathbf{R}_T(\eta) \quad \text{(by definition).} \tag{24}$$

As is visible in Figure 1, this strategy for estimating $\tilde{\lambda}$ is much more effective than the sample-independent strategy in revealing important information. We see from the histogram of upperbounds on $\tilde{\lambda}$ that adaptability is very often small and concentrated near 0, although this is not always the case. Comparatively, upperbounds for λ are spread out with notable mass at large values; we miss out on the interpretation that adaptability very often is small (as we might like to assume, in practice). In the rest of our discussion, all adaptability will be sample-dependent. Note, additional experiments on adaptability are available in Appendix D.1.

Divergence and Approximation In Table 1, we give results for our approximation techniques applied to the model-dependent $h\Delta\mathcal{H}$-divergence and the model-independent $\mathcal{H}\Delta\mathcal{H}$-divergence. The models used in these experiments are trained using **SA**. Since there is actually no ground-truth to compare too, we report performance of our approximations on a ranking task. That is, we compare our approximations to absolute difference in risks on the source and target and compute the Spearman rank correlation. According to our adaptation bounds, smaller divergence should predict smaller difference in risk and larger divergence should predict larger difference in risk as in the ranking task we study. Any effective approximation of divergence should also mimic this behavior, allowing us to conduct an indirect evaluation. In aggregate, we observe both divergences are capable of ranking performance similarity on the source and target, which validates our approximations to some extent. For reference, a recent statistic designed for shift-detection [Rabanser et al., 2019] achieves correlation **0.29** on all data. We also observe the model-dependent divergence typically ranks "better" than the model-independent divergence. This, also, is to be expected according to our theory, since the model-independent divergence does not account for variation in h and should thus perform worse. Overall, the nuanced agreement of our approximations with our theoretical expectations is suggestive that these techniques are effective.

Do Flat Regions Transfer? As noted, one stipulation of practical use for Thm. 6 is a small flatness value ρ. This is not unlike the common assumption that λ is small and, as discussed, is related to the flat-minma hypothesis. To estimate ρ and test our assumption, we select ρ_S and ρ_T to be the smallest values so that Def. 1 is satisfied on S and T using a Monte-Carlo estimate for the Gibbs Risk.[3] We train $\mathbb{Q}$ using a variant of **SA** based on the technique of Pérez-Ortiz et al. [2021]. Our results indicate ρ is typically small as desired with mean **0.007** and SD **0.01** across 4K+ experiments. See Appendix D Figure 5 for a histogram.

Analysis of Assumptions after DANN Our results in Figure 2 show an interesting relationship between the sample complexity of $\mathbb{Q}$ – as measured by $\mathrm{KL}(\mathbb{Q} \,||\, \mathbb{P})$ – and our assumption on adaptability. Namely, we can be more confident in the assumption $\tilde{\lambda}$ is small when the sample complexity of our solution increases. A similar observation holds for the flatness term ρ (see Appendix D.4 Figure 7). Our analysis suggests **DANN** may be a data-hungry algorithm, since solutions with properties we desire have large sample complexity. The practical suggestion is to use large quantities of unlabeled data when applying **DANN**, which is not unreasonable since unlabeled data can be "cheap" to acquire.

Analysis of Divergence after DANN In Figure 2, according to the (more sensitive) model-dependent divergence, **DANN** reduces data-distribution divergence as it is designed to do. Still, it does not reduce divergence to the degree one might expect and, as the sample complexity of the solution increases, the gap between divergences – before and after **DANN** – begins to wane. This is interesting because it shows reduction of divergence and reduction of adaptability / flatness may be competing objectives. Further, this finding echoes theoretical hypotheses in recent literature [Zhao et al., 2019, Wu et al., 2019, Johansson et al., 2019], while also revealing the role of sample complexity in this story. To meet our assumptions when using **DANN**, we should use large amounts of unlabeled data and allow an unconstrained solution, but to ensure **DANN** reduces distribution divergence significantly, we should instead constrain our solution to lower complexity (e.g., via regularization). Depending on problem context, there may be some optimum between these extremes, but in any case, these opposing relationships are an interesting take-away from the application of our theory.

5 CONCLUSION

In this work, we proposed the first adaptation bounds capable of studying the non-uniform sample complexity of adaptation algorithms using multiclass neural networks. Empirically, we validated the novel design-concepts in our adaptation bounds and showed our approximation techniques for some multiclass divergences were effective. In culmination, we applied our bounds to study sample complexity of a common domain-invariant learning algorithm. Our findings revealed unexpected relationships between sample complexity and important properties of the algorithm we studied. Upon acceptance, we plan to make the code for reproducing our experiments publicly available.

[3] A penalty based on Hoeffding Inequality could be added to this estimate to create a valid upperbound. We do not consider this since a penalty is also added if we use the strategy in Eq. (16).

References

Isabela Albuquerque, João Monteiro, Tiago H Falk, and Ioannis Mitliagkas. Adversarial target-invariant representation learning for domain generalization. *arXiv:1911.00804*, 2020.

Shai Ben-David, John Blitzer, Koby Crammer, Fernando Pereira, et al. Analysis of representations for domain adaptation. *NeurIPS*, 19:137, 2007.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1):151–175, 2010.

Gilles Blanchard, Aniket Anand Deshmukh, Ürün Dogan, Gyemin Lee, and Clayton Scott. Domain generalization by marginal transfer learning. *JMLR*, 22:2–1, 2021.

John Blitzer, Mark Dredze, and Fernando Pereira. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *ACL*, pages 440–447, 2007.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *ICML*, pages 1613–1622. PMLR, 2015.

Zhun Deng, Frances Ding, Cynthia Dwork, Rachel Hong, Giovanni Parmigiani, Prasad Patil, and Pragya Sur. Representation via representations: Domain generalization via adversarially learned invariant representations. *arXiv:2006.11478*, 2020.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, pages 4171–4186. ACL.

Gintare Karolina Dziugaite and Daniel M Roy. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. *arXiv:1703.11008*, 2017.

Gintare Karolina Dziugaite, Alexandre Drouin, Brady Neal, Nitarshan Rajkumar, Ethan Caballero, Linbo Wang, Ioannis Mitliagkas, and Daniel M Roy. In search of robust measures of generalization. *NeurIPS*, 33, 2020.

Gintare Karolina Dziugaite, Kyle Hsu, Waseem Gharbieh, Gabriel Arpino, and Daniel Roy. On the role of data in pac-bayes. In *AISTATS*, pages 604–612. PMLR, 2021.

Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, pages 1180–1189. PMLR, 2015.

Pascal Germain, Amaury Habrard, François Laviolette, and Emilie Morvant. A pac-bayesian approach for domain adaptation with specialization to linear classifiers. In *ICML*, pages 738–746. PMLR, 2013.

Pascal Germain, Amaury Habrard, François Laviolette, and Emilie Morvant. A new pac-bayesian perspective on domain adaptation. In *ICML*, pages 859–868. PMLR, 2016.

Pascal Germain, Amaury Habrard, François Laviolette, and Emilie Morvant. Pac-bayes and domain adaptation. *Neurocomputing*, 379:379–397, 2020.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

Sepp Hochreiter and Jürgen Schmidhuber. Flat minima. *Neural computation*, 9(1):1–42, 1997.

Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio. Fantastic generalization measures and where to find them. In *ICLR*, 2019.

Fredrik D Johansson, David Sontag, and Rajesh Ranganath. Support and invertibility in domain-invariant representations. In *AISTATS*, pages 527–536. PMLR, 2019.

Nitish Shirish Keskar, Jorge Nocedal, Ping Tak Peter Tang, Dheevatsa Mudigere, and Mikhail Smelyanskiy. On large-batch training for deep learning: Generalization gap and sharp minima. In *ICLR 2017*, 2017.

Daniel Kifer, Shai Ben-David, and Johannes Gehrke. Detecting change in data streams. In *VLDB*, volume 4, pages 180–191, 2004.

Seiichi Kuroki, Nontawat Charoenphakdee, Han Bao, Junya Honda, Issei Sato, and Masashi Sugiyama. Unsupervised domain adaptation based on source-guided discrepancy. In *AAAI*, volume 33, pages 4122–4129, 2019.

John Langford and Rich Caruana. (not) bounding the true error. *NeurIPS*, 14, 2001.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *IEEE ICCV*, pages 5542–5550, 2017.

Xiao Li and Jeff Bilmes. A bayesian divergence prior for classiffier adaptation. In *AISTATS*, pages 275–282. PMLR, 2007.

Zachary Lipton, Yu-Xiang Wang, and Alexander Smola. Detecting and correcting for label shift with black box predictors. In *ICML*, pages 3122–3130. PMLR, 2018.

Mingsheng Long, Han Zhu, Jianmin Wang, and Michael I Jordan. Deep transfer learning with joint adaptation networks. In *ICML*, pages 2208–2217. PMLR, 2017.

Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. In *NeurIPS*, pages 1647–1657, 2018.

Sara Magliacane, Thijs van Ommen, Tom Claassen, Stephan Bongers, Philip Versteeg, and Joris M Mooij. Domain adaptation by using causal inference to predict invariant conditional distributions. In *NeurIPS*, 2018.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation: Learning bounds and algorithms. *arXiv:0902.3430*, 2009.

Andreas Maurer. A note on the pac bayesian theorem. *arXiv cs/0411099*, 2004.

Daniel McNamara and Maria-Florina Balcan. Risk bounds for transferring representations with and without fine-tuning. In *ICML*, pages 2373–2381. PMLR, 2017.

Vaishnavh Nagarajan and J Zico Kolter. Uniform convergence may be unable to explain generalization in deep learning. *NeurIPS*, 32, 2019.

Behnam Neyshabur. Implicit regularization in deep learning. *arXiv:1709.01953*, 2017.

Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. In search of the real inductive bias: On the role of implicit regularization in deep learning. *arXiv:1412.6614*, 2014.

Behnam Neyshabur, Srinadh Bhojanapalli, David Mcallester, and Nati Srebro. Exploring generalization in deep learning. *NeurIPS*, 30:5947–5956, 2017.

Marıa Pérez-Ortiz, Omar Rivasplata, John Shawe-Taylor, and Csaba Szepesvári. Tighter risk certificates for neural networks. *JMLR*, 22, 2021.

Rashmi Prasad, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind K Joshi, and Bonnie L Webber. The penn discourse treebank 2.0. In *LREC*. Citeseer, 2008.

Stephan Rabanser, Stephan Günnemann, and Zachary C Lipton. Failing loudly: an empirical study of methods for detecting dataset shift. In *NeurIPS*, pages 1396–1408, 2019.

Balaji Polepalli Ramesh and Hong Yu. Identifying discourse connectives in biomedical text. In *AMIA Annual Symposium Proceedings*, volume 2010, page 657. American Medical Informatics Association, 2010.

Ievgen Redko, Amaury Habrard, and Marc Sebban. Theoretical analysis of domain adaptation with optimal transport. In *ECML PKDD*, pages 737–753. Springer, 2017.

Ievgen Redko, Emilie Morvant, Amaury Habrard, Marc Sebban, and Younès Bennani. A survey on domain adaptation theory. *ArXiv*, abs/2004.11829, 2020.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. Wasserstein distance guided representation learning for domain adaptation. In *AAAI*, 2018.

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

Remi Tachet des Combes, Han Zhao, Yu-Xiang Wang, and Geoffrey J Gordon. Domain adaptation with conditional distribution matching and generalized label shift. *NeurIPS*, 33, 2020.

Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. In *IEEE CVPR*, pages 5018–5027, 2017.

Yifan Wu, Ezra Winston, Divyansh Kaushik, and Zachary Lipton. Domain adaptation with asymmetrically-relaxed distribution alignment. In *ICML*, pages 6872–6881. PMLR, 2019.

Kaichao You, Ximei Wang, Mingsheng Long, and Michael Jordan. Towards accurate model selection in deep unsupervised domain adaptation. In *ICML*, pages 7124–7133. PMLR, 2019.

Deniz Zeyrek, Amália Mendes, Yulia Grishina, Murathan Kurfalı, Samuel Gibbon, and Maciej Ogrodniczuk. Ted multilingual discourse bank (ted-mdb): a parallel corpus annotated in the pdtb style. *LREC*, 54(2):587–613, 2020.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. In *ICLR 2017*. OpenReview.net, 2017.

Kun Zhang, Mingming Gong, and Bernhard Schölkopf. Multi-source domain adaptation: A causal view. In *AAAI*, 2015.

Yuchen Zhang, Tianle Liu, Mingsheng Long, and Michael Jordan. Bridging theory and algorithm for domain adaptation. In *ICML*, pages 7404–7413. PMLR, 2019.

Han Zhao, Remi Tachet Des Combes, Kun Zhang, and Geoffrey Gordon. On learning invariant representations for domain adaptation. In *ICML*, pages 7523–7532. PMLR, 2019.

Wenda Zhou, Victor Veitch, Morgane Austern, Ryan P Adams, and Peter Orbanz. Non-vacuous generalization bounds at the imagenet scale: a pac-bayesian compression approach. In *ICLR*, 2018.

A PROOFS

A.1 THEOREM 1

Proof. This is Thm. 2 of Ben-David et al. [2010] with added bound on $\mathbf{R}_S(h) - \mathbf{R}_{\mathbb{S}}(h)$ by standard uniform convergence arguments; e.g., Ch. 28.1 of Shalev-Shwartz and Ben-David [2014]. Boole's Inequality is used to combine bounds. □

A.2 THEOREM 7 OF GERMAIN ET AL. [2020]

Theorem 7. *[Germain et al., 2020] Let $\mathcal{Y}$ be binary, $\mathbb{P}$ any distribution over $\mathcal{H}$, and $\omega > 0$. For all $\delta > 0$, w.p. at least $1 - \delta$, for all distributions $\mathbb{Q}$ over $\mathcal{H}$,*

$$\mathbf{R}_{\mathbb{T}}(\mathbb{Q}) \leq \omega'(\mathbf{R}_S(\mathbb{Q}) + |\mathrm{d}_S(\mathbb{Q}) - \mathrm{d}_T(\mathbb{Q})|) + |\mathrm{e}_{\mathbb{S}}(\mathbb{Q}) - \mathrm{e}_{\mathbb{T}}(\mathbb{Q})| + 2\omega\frac{\mathrm{KL}(\mathbb{Q}||\mathbb{P}) - \ln(\delta/3)}{m\omega'} + 2(\omega' - 1) \tag{25}$$

where $\omega' = 2\omega/(1 - \exp(-2\omega))$ and for $H_i \sim (\mathbb{Q})_i$, $(X, Y) \sim \mathbb{S}$ we have

$$\begin{aligned} \mathrm{e}_{\mathbb{S}}(\mathbb{Q}) &\stackrel{\text{def}}{=} \mathbf{E}[(1 - \mathbf{1}_{\{H_1(X)\}}\{Y\})(1 - \mathbf{1}_{\{H_2(X)\}}\{Y\})], \\ \mathrm{d}_{\mathbb{S}}(\mathbb{Q}) &\stackrel{\text{def}}{=} \mathbf{E}[1 - \mathbf{1}_{\{H_1(X)\}}\{H_2(X)\}]. \end{aligned} \tag{26}$$

In comparison to Thm. 1, the absolute difference in disagreement d is most similar to the $\mathcal{H}\Delta\mathcal{H}$-divergence and the absolute difference in joint-error e is most similar to the adaptability λ [Germain et al., 2020]. For this reason, in our discussion in Section 2, we refer to the former as the "divergence" and the latter as the "adaptability".

Proof. This is a simplification of Thm. 7 of Germain et al. [2020]. We set $\omega = a$ in the original notation and use the fact that $\omega/(1 - \exp(-\omega))$ is increasing for $\omega > 0$. □

A.3 THEOREM 2

Before diving into the proof, we setup some helpful notation and Lemmas.

A.3.1 Notation

Frequently in our proofs, we use the *error gap*, defined for any distributions $\mathbb{S}, \mathbb{T}$ and hypothesis h

$$\Delta_h(\mathbb{S}, \mathbb{T}) \stackrel{\text{def}}{=} |\mathbf{R}_{\mathbb{S}}(h) - \mathbf{R}_{\mathbb{T}}(h)|. \tag{27}$$

By the identification in Eq. (3), we observe that $\Delta_h(S, T)$ is also well-defined for any random samples S and T. Also, using the usual definition of the Gibbs risk, $\Delta_{\mathbb{Q}}(\mathbb{S}, \mathbb{T})$ is well-defined for any distribution $\mathbb{Q}$ over a hypothesis space $\mathcal{H}$. Occasionally, we also use two-subscripts on the error-gap Δ. The intended meaning is intuitive:

$$\Delta_{q,p}(\mathbb{S}, \mathbb{T}) \stackrel{\text{def}}{=} |\mathbf{R}_{\mathbb{S}}(q) - \mathbf{R}_{\mathbb{T}}(p)|. \tag{28}$$

This notation will be especially useful in proofs since $\Delta_{q,p}(\mathbb{S}, \mathbb{T})$ obeys a triangle-inequality with respect to the subscripts and arguments. Further, any bound on $\Delta_{\mathbb{Q}}(S, \mathbb{T})$ trivially yields a PAC-Bayesian adaptation bound for the Gibbs predictor $\mathbb{Q}$ by definition of the absolute value.

As another short-hand in proofs, we frequently use the following more evocative expressions for the indicator function:

$$1[a = b] \stackrel{\text{def}}{=} \mathbf{1}_{\{a\}}\{b\}; \qquad 1[a \neq b] \stackrel{\text{def}}{=} 1 - \mathbf{1}_{\{a\}}\{b\}. \tag{29}$$

Now, we can proceed with the employed Lemmas.

A.3.2 Lemmas

In this section, we build to the proof of Theorem 2. These results consist of most of the "real" work in proving the result. They range in degree of novelty and we provide some exposition on this point here. Lemma 1 is an adaptation of the triangle-inequality for 01-loss [Crammer, Kearns and Wortman, 2007, Ben-David et al., 2007] to the multiclass setting. Similarly, Lemma 2 is an adaptation of the main inequality of Ben-David et al. [2010] to the multiclass setting. The former requires some work to verify the logic, while our overall strategy for the latter is similar to the binary case. Next, Lemma 3 uses the identification in Eq. (3) to apply Lemma 2 to the random samples S and T. While it is a simple insight, it is extremely important, since it enables us to introduce the sample-dependent adaptability $\tilde{\lambda}$. The next result, Lemma 4, is well-known in PAC-Bayes. Meanwhile, the final result, Lemma 5, is a new result which allows us to apply Lemma 3 to Gibbs predictors. When broken down in this manner, as is our intention, the individual pieces that build to our bound may appear simple. Still, it is important to remember that PAC-Bayesian bounds have never previously been combined with multiclass variants of the results of Ben-David et al. [2007, 2010]. After some trial and error, we've found our primary innovations – the use of sample-independent adaptability, along with Lemma 5 – are vital to introducing the desired non-uniform notion of sample complexity. In any case, we now proceed by stating and proving each of the discussed Lemmas.

Lemma 1. *For any $(h, h') \in \mathcal{H}^2$ and any $(x, y) \in \mathcal{X} \times \mathcal{Y}$,*

$$1[h(x) \neq y] \leq 1[h(x) \neq h'(x)] + 1[h'(x) \neq y] \tag{30}$$

and

$$1[h(x) \neq h'(x)] \leq 1[h'(x) \neq y] + 1[y \neq h(x)]. \tag{31}$$

Proof. We begin with Eq. (30). We use proof by exhaustion. If $h(x) = y$, then the LHS is 0 and the RHS will always be non-negative so the equation is true. If $h(x) \neq y$ and $h(x) \neq h'(x)$, then the equation evaluates to $1 \leq 1 + c$ for $c \geq 0$ which is true. If $h(x) \neq y$ and $h(x) = h'(x)$, then $h'(x) \neq y$, and $1 \leq 1$ which is true. This concludes the argument.

Next, we consider Eq. (31). Again, we use proof by exhaustion. If $h(x) = h'(x)$, the LHS is 0. If $h(x) \neq h'(x)$ and $h(x) = y$, we have $h'(x) \neq y$ and the equation evaluates to $1 \leq 1$ which is true. If $h(x) \neq h'(x)$ and $h(x) \neq y$, it evaluates to $1 \leq 1 + c$ for $c \geq 0$ which is true and concludes the argument. □

Lemma 2. *For any distributions $\mathbb{D}_1$ and $\mathbb{D}_2$ over $\mathcal{X} \times \mathcal{Y}$, for any $h \in \mathcal{H}$*

$$\mathbf{R}_{\mathbb{D}_1}(h) \leq \mathbf{R}_{\mathbb{D}_2}(h) + \mathbf{d}_{\mathcal{C}_h}((\mathbb{D}_1)_X, (\mathbb{D}_2)_X) + \min_{\eta \in \mathcal{H}} \mathbf{R}_{\mathbb{D}_1}(\eta) + \mathbf{R}_{\mathbb{D}_2}(\eta) \tag{32}$$

where $\mathcal{C}_h = \mathcal{H}\Delta\mathcal{H}$ and $(\mathbb{D}_i)_X$ is the $\mathcal{X}$-marginal of $\mathbb{D}_i$.[4]

Proof. Let $\mathbb{D}_1$, $\mathbb{D}_2$, and h as assumed.

Recall by Lemma 1 Eq. (30), for any h' in $\mathcal{H}$ and any $(x, y) \in \mathcal{X} \times \mathcal{Y}$

$$1[h(x) \neq y] \leq 1[h(x) \neq h'(x)] + 1[h'(x) \neq y]. \tag{33}$$

Then, by monotonicity and linearity of the expectation, for any choice of h',

$$\begin{aligned} \mathbf{R}_{\mathbb{D}_1}(h) &\leq \mathbf{E}[1[h(X_1) \neq h'(X_1)]] + \mathbf{R}_{\mathbb{D}_1}(h'); \qquad X_1 \sim (\mathbb{D}_1)_X \\ &\leq \mathbf{E}[1[h(X_2) \neq h'(X_2)]] + \mathbf{R}_{\mathbb{D}_1}(h') + \xi; \qquad X_2 \sim (\mathbb{D}_2)_X \end{aligned} \tag{34}$$

where

$$\begin{aligned} \xi &= \left|\mathbf{E}[1[h(X_2) \neq h'(X_2)]] - \mathbf{E}[1[h(X_1) \neq h'(X_1)]]\right| \\ &\leq \mathbf{d}_{\mathcal{C}_h}((\mathbb{D}_1)_X, (\mathbb{D}_2)_X)) \qquad \text{(by definition of supremum, for either choice of } \mathcal{C}_h). \end{aligned} \tag{35}$$

Alternatively, by Lemma 1 Eq. (31), for any choice of h', x, y,

$$1[h(x) \neq h'(x)] \leq 1[h'(x) \neq y] + 1[y \neq h(x)]. \tag{36}$$

[4] In a formal sense, $(\mathbb{D}_i)_X$ is the pushforward distribution $\mathbb{D}_i \circ \pi^{-1}$ of the projection $\pi : \mathcal{X} \times \mathcal{Y} \to \mathcal{X}$ defined $\pi(x, y) = x$.

Using monotonicty and linearity of the expectation as before, we have

$$\mathbf{E}[1[h(X_2) \neq h'(X_2)]] \leq \mathbf{R}_{\mathbb{D}_2}(h') + \mathbf{R}_{\mathbb{D}_2}(h); \qquad X_2 \sim (\mathbb{D}_2)_X. \tag{37}$$

As the above holds for any $h' \in \mathcal{H}$, select h' to be minimizer of the quantity $\mathbf{R}_{\mathbb{D}_1}(h') + \mathbf{R}_{\mathbb{D}_2}(h')$.

This yields the desired result. □

Lemma 3. *Almost surely, w.r.t samples S and T,*

$$\forall h \in \mathcal{H} \ : \ \Delta_h(S,T) \leq \tilde{\lambda} + \mathbf{d}_{\mathcal{C}_h}(S_X, T_X) \tag{38}$$

where $\tilde{\lambda} \stackrel{\text{def}}{=} \min_{h \in \mathcal{H}} \mathbf{R}_S(h) + \mathbf{R}_T(h)$ and the bound holds for both $\mathcal{C}_h = \mathcal{H}\Delta\mathcal{H}$ and $\mathcal{C}_h = h\Delta\mathcal{H}$.

Proof. The statement asserts the following holds with probability 1 according to the random draws of S and T:

$$\forall h \in \mathcal{H} \ : \ \Delta_h(S,T) \leq \tilde{\lambda} + \mathbf{d}_{\mathcal{C}_h}(S_X, T_X) \tag{39}$$

It is sufficient to show the statement holds for any realization of S and T. Recall, for any realization, S and T themselves define distributions by the identification in Eq. (3). So, Lemma 2 may be applied. Doing so twice and interchanging the roles of S and T gives

$$\forall h \in \mathcal{H} \ : \ \mathbf{R}_h(S) - \mathbf{R}_h(T) \leq \tilde{\lambda} + \mathbf{d}_{\mathcal{C}_h}(S_X, T_X) \qquad \text{and} \qquad \mathbf{R}_h(T) - \mathbf{R}_h(S) \leq \tilde{\lambda} + \mathbf{d}_{\mathcal{C}_h}(S_X, T_X). \tag{40}$$

So, the absolute difference between $\mathbf{R}_h(S)$ and $\mathbf{R}_h(T)$ is also bounded and we have our result. □

Lemma 4. *[Maurer, 2004] For any distribution $\mathbb{P}$ over $\mathcal{H}$, for any $\delta > 0$,*

$$\mathbf{Pr}\Big(\forall\, \mathbb{Q} \ : \ \Delta_{\mathbb{Q}}(T, \mathbb{T}) \leq \sqrt{\tfrac{\mathrm{KL}(\mathbb{Q}||\mathbb{P}) + \ln\sqrt{4m} - \ln(\delta)}{2m}}\ \Big) \geq 1 - \delta. \tag{41}$$

Proof. This is the result of Maurer [2004] given below

$$\mathbf{Pr}\left(\mathrm{kl}(\mathbf{R}_T(\mathbb{Q}) \,||\, \mathbf{R}_{\mathbb{T}}(\mathbb{Q})) \leq \frac{\mathrm{KL}(\mathbb{Q}||\mathbb{P}) - \ln\delta + \ln\sqrt{4m}}{m}\right) \geq 1 - \delta, \tag{42}$$

where the "little" kl is the KL-divergence between Binomial distributions parameterized by its arguments. The above bound implies the stated result by application of Pinsker's Inequality. □

Lemma 5. *For any distribution $\mathbb{Q}$, almost surely w.r.t samples S and T,*

$$\Delta_{\mathbb{Q}}(S,T) \leq \tilde{\lambda} + \mathbf{E}_H[\mathbf{d}_{\mathcal{C}_H}(S_X, T_X)] \tag{43}$$

where $\tilde{\lambda}$ and $\mathcal{C}_H$ are defined as in Lemma 3.

Proof. We apply Lemma 3. By Jensen's Inequality, monotonicity of $\mathbf{E}$, and linearity of $\mathbf{E}$, we have

$$\Delta_{\mathbb{Q}}(S,T) \leq \mathbf{E}_H[\Delta_H(S,T)] \leq \tilde{\lambda} + \mathbf{E}_H[\mathbf{d}_{\mathcal{C}_H}(S_X, T_X)] \tag{44}$$

almost surely. In more details, for any realization of S and T,

$$\begin{aligned}
\Delta_{\mathbb{Q}}(S,T) &= \big|\mathbf{R}_{\mathbb{Q}}(S) - \mathbf{R}_{\mathbb{Q}}(T)\big| \\
&= \big|\mathbf{E}[\mathbf{R}_S(H)] - \mathbf{E}[\mathbf{R}_T(H)]\big| && (H \sim \mathbb{Q},\ S \text{ fixed},\ T \text{ fixed}) \\
&= \Big|\mathbf{E}\Big[\mathbf{R}_S(H) - \mathbf{R}_T(H)\Big]\Big| && (\text{Linearity of } \mathbf{E}) \\
&\leq \mathbf{E}\big[\Delta_H(S,T)\big] && (\text{Jensen's Inequality}) \\
&\leq \mathbf{E}\big[\tilde{\lambda} + \mathbf{d}_{\mathcal{C}_H}(S_X, T_X)\big] && (\text{Lemma 3 and monotonicity of } \mathbf{E}) \\
&\leq \tilde{\lambda} + \mathbf{E}[\mathbf{d}_{\mathcal{C}_H}(S_X, T_X)] && (\text{Linearity of } \mathbf{E}).
\end{aligned}$$

□

A.3.3 Proof

We give the final proof of Theorem 2 below. Admittedly, it is a bit underwhelming, since most of the work has gone into the Lemmas above. The remaining component we rely on is our notation for the error-gap Δ. By design, this notation exhibits a triangle-inequality.

Proof. Observe,

$$\Delta_{\mathbb{Q}}(S, \mathbb{T}) \leq \Delta_{\mathbb{Q}}(S, T) + \Delta_{\mathbb{Q}}(T, \mathbb{T}). \tag{45}$$

To bound the former, we use Lemma 5. To bound the latter, we use Lemma 4. We use Boole's Inequality to combine to the desired result. □

A.4 THEOREM 3

As noted in the main text, we employ the overall strategy of Ben-David et al. [2010]. The main distinction in our result below is the removal of any symmetry assumption on $\mathcal{H}$.

Proof. As before, we show the statement holds for any realization of S_X and T_X.

Let $\mathcal{C} = \mathcal{H}\Delta\mathcal{H}$ and expand the divergence as below

$$\mathbf{d}_{\mathcal{C}}(S_X, T_X) = \max_{\varphi \in \mathcal{H}\Delta\mathcal{H}} \left|\mathbf{E}[\varphi(X)] - \mathbf{E}[\varphi(\tilde{X})]\right| = \max_{\varphi \in \mathcal{H}\Delta\mathcal{H}} \left|\mathbf{Pr}(\varphi(X) = 1) - \mathbf{Pr}(\varphi(\tilde{X}) = 1)\right| \tag{46}$$

where $X \sim S_X$, $\tilde{X} \sim T_X$. Note, we substitute $\max$ for $\sup$ because both S_X and T_X are finitely supported, and thus, some $\varphi \in \mathcal{C}$ does achieve the maximum. Then, we have

$$\begin{aligned}
&\max_{\varphi \in \mathcal{H}\Delta\mathcal{H}} \left|\mathbf{Pr}(\varphi(X) = 1) - \mathbf{Pr}(\varphi(\tilde{X}) = 1)\right| \\
&= \max_{\varphi \in \mathcal{H}\Delta\mathcal{H}} \max \left\{ \begin{matrix} \mathbf{Pr}(\varphi(X) = 1) - \mathbf{Pr}(\varphi(\tilde{X}) = 1), \\ \mathbf{Pr}(\varphi(\tilde{X}) = 1) - \mathbf{Pr}(\varphi(X) = 1) \end{matrix} \right\} \\
&= \max_{\varphi \in \mathcal{H}\Delta\mathcal{H}} \max \left\{ \begin{matrix} 1 - \mathbf{Pr}(\varphi(X) = 0) - \mathbf{Pr}(\varphi(\tilde{X}) = 1), \\ 1 - \mathbf{Pr}(\varphi(\tilde{X}) = 0) - \mathbf{Pr}(\varphi(X) = 1) \end{matrix} \right\} \\
&= \max \left\{ \begin{matrix} 1 - \min_{\varphi \in \mathcal{H}\Delta\mathcal{H}} \mathbf{Pr}(\varphi(X) = 0) + \mathbf{Pr}(\varphi(\tilde{X}) = 1), \\ 1 - \min_{\varphi \in \mathcal{H}\Delta\mathcal{H}} \mathbf{Pr}(\varphi(\tilde{X}) = 0) + \mathbf{Pr}(\varphi(X) = 1) \end{matrix} \right\}.
\end{aligned} \tag{47}$$

The first equality follows by definition of absolute value, the second by law of compliments, and last because consecutive applications of the $\max$ operation may be interchanged. Taking P, Q, U, V as assumed, the result follows by the definition of risk (i.e., Eq. (1)). □

A.5 THEOREM 4

As we are aware, Theorem 4 is the first proposal for approximation of ERM over the class $\mathcal{H}\Delta\mathcal{H}$ when $\mathcal{H}$ has multiclass output. Our strategy is to identify an appropriate score-based surrogate expression for any $\varphi \in \mathcal{S}\Delta\mathcal{S}$; i.e., which is positive where φ returns 1 and negative otherwise. Upon doing so, we can use standard techniques for giving smooth upperbounds to the 01-loss.

Proof. Let $x \in \mathcal{X}$, $\mathbf{f}, \mathbf{g} \in \mathcal{F}$ and suppose $\mathbf{f}(x)$ and $\mathbf{g}(x)$ have no repeated entries. Recall, for any two sets of non-negative numbers S_1 and S_2 the following equality holds[5]

$$\max\{a \cdot b \mid a \in S_1, b \in S_2\} = (\max S_1) \cdot (\max S_2). \tag{48}$$

[5]Suppose not. Then, WLOG $\max\{a \cdot b\} = d \cdot e > (\max S_1) \cdot (\max S_1)$ for some $d \neq \max S_1$ or some $e \neq \max S_2$. But, we also have $d \cdot e \leq d \cdot \max S_2 \leq (\max S_1) \cdot (\max S_2)$, a contradiction.

From this and the fact that τ is non-negative and order-preserving, we know $\mathbf{A}_{ii} \geq \mathbf{A}_{jk}$ for some $i \in [C]$ and all $(j,k) \in [C]^2$ if and only if

$$i = \arg\max_{\ell \in C} \mathbf{f}_\ell(x) = \arg\max_{\ell \in C} \mathbf{g}_\ell(x). \tag{49}$$

Notice, ties are impossible due to the assumed uniqueness of the scores. So, by this same logic, we observe

$$\begin{aligned} & \arg\max_{\ell \in C} \mathbf{f}_\ell(x) \neq \arg\max_{\ell \in C} \mathbf{g}_\ell(x) \\ \text{iff} \quad & \forall\, i \in [C],\ \exists\, (j,k) \in [C]^2 \ : \mathbf{A}_{ii} < \mathbf{A}_{jk} \\ \text{iff} \quad & \max_{i \in [C]} \mathbf{A}_{ii} < \max_{(j,k) \in [C]^2} \mathbf{A}_{jk} \\ \text{iff} \quad & 0 < \max_{(j,k) \in [C]^2} \mathbf{A}_{jk} - \max_{i \in [C]} \mathbf{A}_{ii} = z(x) \end{aligned} \tag{50}$$

So, under the current assumptions, the score $z(x)$ is positive if and only if $\hat{y} = 1 - \mathbf{1}_{\{\Psi_{\mathbf{f}}(x)\}}\{\Psi_{\mathbf{g}}(x)\} = 1$. Using this fact, it is easy to verify $\mathcal{L}(z(x), y) \geq 1[\hat{y} \neq y]$ for each case $(\hat{y}, y) \in \{(0,0), (0,1), (1,0), (1,1)\}$. The loss $\mathcal{L}$ is actually a standard surrogate – i.e., the cross-entropy – multiplied by a constant factor as in Dziugaite and Roy [2017] to turn it into a propper upperbound on the 01-loss. The main novelty here comes from defining $z(x)$ to be positive whenever $\hat{y}$ is.

Notice, the inequality holds on all but a set of measure 0, according to $\mathbb{D}$. Thus, monotonicity of $\mathbf{E}$ gives the result. □

A.6 THEOREM 5

As noted in the main text, Theorem 5 is conceptually similar to a result – in the binary case – given by Kuroki et al. [2019]. Unfortunately, their strategy does not simply extend to the multiclass case: there is a loss of precision due to the increased degrees of freedom in multiclass classification. To remedy this, we introduce the class Υ for use in our reduction. As a result, we observe the need to add additional constraints on the labeling function for the classification problem. Careful attention is paid show this constrained labeling function can be independent of the classifier we wish to learn $\varphi \in \mathcal{H}$, which enables our appeal to a simple heuristic that is also independent of φ. Otherwise, in simpler formulations, this dependence produces a more complicated minimization problem.

Proof. We show the statement holds for any realization of S_X and T_X.

Let $h \in \mathcal{H}$ arbitrarily and let $\mathcal{C} = h\Delta\mathcal{H}$. We proceed by expanding the divergence:

$$\begin{aligned} \mathbf{d}_{\mathcal{C}}(S_X, T_X) &= \max_{\nu \in h\Delta\mathcal{H}} \left| \mathbf{Pr}(\nu(X) = 1) - \mathbf{Pr}(\nu(\tilde{X}) = 1) \right| \\ &= \max \left\{ \begin{array}{l} 1 - \min_{\nu \in h\Delta\mathcal{H}} \mathbf{Pr}(\nu(X) = 0) + \mathbf{Pr}(\nu(\tilde{X}) = 1), \\ 1 - \min_{\nu \in h\Delta\mathcal{H}} \mathbf{Pr}(\nu(\tilde{X}) = 0) + \mathbf{Pr}(\nu(X) = 1) \end{array} \right\} \\ &= \max \left\{ \begin{array}{l} 1 - \min_{\varphi \in \mathcal{H}} \mathbf{Pr}(h(X) = \varphi(X)) + \mathbf{Pr}(h(\tilde{X}) \neq \varphi(\tilde{X})), \\ 1 - \min_{\varphi \in \mathcal{H}} \mathbf{Pr}(h(\tilde{X}) = \varphi(\tilde{X})) + \mathbf{Pr}(h(X) \neq \varphi(X)) \end{array} \right\} \end{aligned} \tag{51}$$

where $X \sim S_X$ and $\tilde{X} \sim T_X$. The first and second lines follow from an identical expansion as in the proof of Theorem 3. The last follows by definition of $h\Delta\mathcal{H}$.

Next, we observe

$$\forall \varphi \in \mathcal{H},\ \forall \bar{h} \in \Upsilon \ :\ \mathbf{Pr}(h(X) = \varphi(X)) + \mathbf{Pr}(h(\tilde{X}) \neq \varphi(\tilde{X})) \leq \mathbf{Pr}(\bar{h}(X) \neq \varphi(X)) + \mathbf{Pr}(h(\tilde{X}) \neq \varphi(\tilde{X})). \tag{52}$$

The inequality follows by the monotonicity of probability and the fact

$$\{x \mid h(x) = \varphi(x)\} \subseteq \{x \mid \bar{h}(x) \neq \varphi(x)\} \qquad \text{(by definition of } \bar{h}). \tag{53}$$

Meanwhile, setting

$$\bar{h}^*_\varphi(x) \stackrel{\text{def}}{=} \begin{cases} \varphi(x), & \text{if } \varphi(x) \neq h(x) \\ \max\{\ell \in [C] \mid \ell \neq \varphi(x)\}, & \text{else} \end{cases} \tag{54}$$

implies $\bar{h}^*_\varphi \in \Upsilon$ and $\{x \mid h(x) = \varphi(x)\} = \{x \mid \bar{h}^*_\varphi(x) \neq \varphi(x)\}$. So, we also have

$$\forall \varphi \in \mathcal{H} \ : \ \mathbf{Pr}(h(X) = \varphi(X)) + \mathbf{Pr}(h(\tilde{X}) \neq \varphi(\tilde{X})) = \mathbf{Pr}(\bar{h}^*_\varphi(X) \neq \varphi(X)) + \mathbf{Pr}(h(\tilde{X}) \neq \varphi(\tilde{X})). \tag{55}$$

Considering that $\bar{h}^*_\varphi \in \Upsilon$, Eq. (52) and Eq. (55) in combination tell us

$$\forall \varphi \in \mathcal{H} \ : \ \mathbf{Pr}(h(X) = \varphi(X)) + \mathbf{Pr}(h(\tilde{X}) \neq \varphi(\tilde{X})) = \min_{\bar{h} \in \Upsilon} \mathbf{Pr}(\bar{h}(X) \neq \varphi(X)) + \mathbf{Pr}(h(\tilde{X}) \neq \varphi(\tilde{X})) \tag{56}$$

since the lowerbound in Eq. (52) is always attained by the upperbound, for some $\bar{h} \in \Upsilon$; i.e., the $\min$ is sandwiched.

Note, an identical argument also gives,

$$\forall \varphi \in \mathcal{H} \ : \ \mathbf{Pr}(h(\tilde{X}) = \varphi(\tilde{X})) + \mathbf{Pr}(h(X) \neq \varphi(X)) = \min_{\bar{h} \in \Upsilon} \mathbf{Pr}(\bar{h}(\tilde{X}) \neq \varphi(\tilde{X})) + \mathbf{Pr}(h(X) \neq \varphi(X)). \tag{57}$$

To continue, we apply Eq. (56) and Eq. (57) to Eq. (51). Specifically, we have

$$\begin{aligned} &\max \left\{ \begin{array}{l} 1 - \min\limits_{\varphi \in \mathcal{H}} \mathbf{Pr}(h(X) = \varphi(X)) + \mathbf{Pr}(h(\tilde{X}) \neq \varphi(\tilde{X})), \\ 1 - \min\limits_{\varphi \in \mathcal{H}} \mathbf{Pr}(h(\tilde{X}) = \varphi(\tilde{X})) + \mathbf{Pr}(h(X) \neq \varphi(X)) \end{array} \right\} \\ &= \max \left\{ \begin{array}{l} 1 - \min\limits_{\substack{\varphi \in \mathcal{H}, \\ \bar{h} \in \Upsilon}} \mathbf{Pr}(\bar{h}(X) \neq \varphi(X)) + \mathbf{Pr}(h(\tilde{X}) \neq \varphi(\tilde{X})), \\ 1 - \min\limits_{\substack{\varphi \in \mathcal{H}, \\ \bar{h} \in \Upsilon}} \mathbf{Pr}(\bar{h}(\tilde{X}) \neq \varphi(\tilde{X})) + \mathbf{Pr}(h(X) \neq \varphi(X)) \end{array} \right\}. \end{aligned} \tag{58}$$

Taking P, Q, U, V as assumed, the desired result follows by the definition of risk in Eq. (1). □

A.7 THEOREM 6

As noted in the main text, this result introduces a deterministic reference to avoid costly Monte-Carlo estimation. It is the consequence of a series of triangle-inequalities and some of the Lemmas disucssed in proof of Thm. 2.

Proof. Observe, for any h_*,

$$\begin{aligned} \Delta_{\mathbb{Q}}(S, \mathbb{T}) &\leq \Delta_{\mathbb{Q}, h_*}(S, T) + \Delta_{h_*, \mathbb{Q}}(T, \mathbb{T}) \\ &\leq \Delta_{h_*}(S, T) + \Delta_{\mathbb{Q}, h_*}(S, S) + \Delta_{\mathbb{Q}}(T, \mathbb{T}) + \Delta_{\mathbb{Q}, h_*}(T, T) \\ &\leq \rho + \Delta_{h_*}(S, T) + \Delta_{\mathbb{Q}}(T, \mathbb{T}) \end{aligned} \tag{59}$$

Use Lemma 3 and Lemma 4, respectively, to bound the latter two terms. Application of Boole's Inequality and selection of $h_* = \mathbf{E}_{H \sim \mathbb{Q}}[H]$ gives the result. □

A.8 COROLLARY 1

Conceptually, this result relies on the same proof-technique as Theorem 6, but the proof is still a bit more technically involved than a typical "Corollary" because it requires the measure-theoretic notion of a pushforward. We consider pushfowards of empirical distributions, which are finitely supported, so there is no need to discuss issues of measurability.

Proof. Following the proof of Theorem 6, we have $\Delta_{\mathbb{Q}}(S, \mathbb{T}) \leq \rho + \Delta_\mu(S, T) + \Delta_{\mathbb{Q}}(T, \mathbb{T})$. Now, recalling μ is the composition of a classifier c_μ and a feature extractor f_μ, we have

$$\Delta_\mu(S, T) = \Delta_{c_\mu}(S \circ f_\mu^{-1}, T \circ f_\mu^{-1}) \tag{60}$$

where we abuse notation and write $\mathbb{D} \circ g^{-1}$ for the pushforward of a distribution $\mathbb{D}$ on $\mathcal{X} \times \mathcal{Y}$ by the function $\Phi_g(x, y) = (g(x), y)$. In details, setting $\ell(h, (x, y)) = 1[h(x) \neq y]$ and assuming $f_\mu : \mathcal{X} \to \mathcal{Z}$ and $c_\mu : \mathcal{Z} \to \mathcal{Y}$, Eq. (60) follows because

$$\mathbf{R}_{\mathbb{D}}(\mu) = \int_{\mathcal{X} \times \mathcal{Y}} \ell(c_\mu \circ f_\mu, v) \mathbb{D}(\mathrm{d}v) = \int \ell(c_\mu, \Phi_{f_\mu}(v)) \mathbb{D}(\mathrm{d}v) = \int_{\mathcal{Z} \times \mathcal{Y}} \ell(c_\mu, w) \mathbb{D} \circ f_\mu^{-1}(\mathrm{d}w) = \mathbf{R}_{\mathbb{D} \circ f_\mu^{-1}}(c_\mu) \tag{61}$$

for any distribution $\mathbb{D}$ over $\mathcal{X} \times \mathcal{Y}$. After applying the equality in Eq. (60), we can conclude our argument as in the proof of Theorem 2 using Lemma 3. Although, it should be noted the adaptation problem has changed slightly, since we now consider the hypothesis space $\mathcal{W} = \{c_h \mid h \in \mathcal{H}\} \subseteq \mathcal{Y}^{\mathcal{Z}}$, the source distribution $\mathbb{S} \circ f_\mu^{-1}$ over $\mathcal{Z} \times \mathcal{Y}$, and the target distribution $\mathbb{T} \circ f_\mu^{-1}$ over $\mathcal{Z} \times \mathcal{Y}$. Of course, Lemma 3 still applies in this case, so this does not present an issue.

After this, to arrive at the result in the main text, we simplify terms to remove any discussion of pushforward distributions. For any risks, this is accomplished by reversing the steps in Eq. (61). For any divergences, a similar equality holds and can be applied. In particular, for any $\mathcal{Q} \subseteq \mathcal{Z}^{\mathcal{Y}}$, any function $p : \mathcal{X} \to \mathcal{Z}$, and any distributions $\mathbb{S}$ and $\mathbb{T}$ over $\mathcal{X} \times \mathcal{Y}$, we use the expansion below:

$$\begin{aligned}
\mathbf{d}_{\mathcal{Q}}((\mathbb{S} \circ p^{-1})_Z, (\mathbb{T} \circ p^{-1})_Z) &= \sup_{q \in \mathcal{Q}} |\mathbf{E}_{Z \sim (\mathbb{S} \circ p^{-1})_Z}[q(Z)] - \mathbf{E}_{Z \sim (\mathbb{T} \circ p^{-1})_Z}[q(Z)]| \\
&= \sup_{q \in \mathcal{Q}} |\mathbf{E}_{X \sim \mathbb{S}_X}[(q \circ p)(X)] - \mathbf{E}_{X \sim \mathbb{T}_X}[(q \circ p)(X)]| \qquad \text{(similiar to Eq. (61))} \\
&= \sup_{r \in \mathcal{Q} \circ p^{-1}} |\mathbf{E}_{X \sim \mathbb{S}_X}[r(X)] - \mathbf{E}_{X \sim \mathbb{T}_X}[r(X)]| \qquad (\mathcal{Q} \circ p^{-1} \stackrel{\text{def}}{=} \{q \circ p \mid q \in \mathcal{Q}\}) \\
&= \mathbf{d}_{\mathcal{Q} \circ p^{-1}}(S_X, T_X).
\end{aligned} \tag{62}$$

Taking $\mathcal{Q} = \{1 - \mathbf{1}_{\{c(\cdot)\}}\{c'(\cdot)\} \mid (c, c') \in \mathcal{W}\}$ and $p = f_\mu$, we end up with $\mathcal{Q} \circ p^{-1} = [\mathcal{H}\Delta\mathcal{H}]_\mu$ as defined in the main text. Likewise, taking $\mathcal{Q} = \{1 - \mathbf{1}_{\{c_\mu(\cdot)\}}\{c'(\cdot)\} \mid c' \in \mathcal{W}\}$ and $p = f_\mu$, we end up with $\mathcal{Q} \circ p^{-1} = [\mu\Delta\mathcal{H}]_\mu$. □

B EXTENDED RELATED WORKS

Here, we give an extended version of the related works (Section 2.3). First, we discuss theoretical adaptation work. We compartmentalize relevant contributions based on some key-terms common to adaptation bounds. Following this, we discuss related works in PAC-Bayes, in which, we give a more in depth history of these bounds.

Divergence Many bounds use a modified, or generalized, divergence term. Mansour et al. [2009] define divergence for *any* loss function (i.e., in addition to the 01-loss we consider). With some restrictions on hypothesis space, Redko et al. [2017] show a Wasserstein metric may be used to bound error. Shen et al. [2018] extend this to more general settings. As noted by Redko et al. [2020], bounds based on Wasserstein metric imply bounds based on MMD [Gretton et al., 2012] due to a general relationship between the two. Johansson et al. [2019] give another bound based on an integral probability metric. Note, none of these works consider approximation of divergences used to bound 01-loss in multiclass settings. In this regard, the closest work to ours is Zhang et al. [2019] who approximate a divergence used to bound a multiclass *margin* loss.

Adaptability Besides requiring small adaptability term, some theoretical DA works consider other possible assumptions. For example, a covariate shift assumption can be made: the marginal feature distributions disagree, but the feature-conditional label distributions are identical. This assumption is useful, for example, in designing model-selection algorithms [Sugiyama et al., 2007, You et al., 2019], but Ben-David, Lu, Luu and Pál [2010] show this assumption (on its own) is *not* enough for the general DA problem. Another frequent assumption is label-shift: the marginal label distributions disagree, but the label-conditional feature distributions remain the same. As mentioned, Zhao et al. [2019] show failure-cases in this context, while Lipton et al. [2018] propose techniques for detecting and correcting shift in this case. Similarly, Tachet des Combes et al. [2020] propose *generalized* label-shift and motivate new algorithms in this context. The DA problem can also be modeled through causal graphs [Zhang et al., 2015, Magliacane et al., 2018] and some extensions to DA consider a meta-distribution over targets [Blanchard et al., 2021, Albuquerque et al., 2020, Deng et al., 2020]. Notably, most assumptions are untestable in practice, but not many works consider this. As we are aware, we are the first work to use a sample-dependent adaptability term, which improves estimation in empirical study.

PAC-Bayes For completeness, besides what is discussed here, readers are directed to the work of Catoni [2007], McAllester [2013], Germain et al. [2009, 2015], and the primer by Guedj [2019]. While PAC-Bayes is often attributed to McAllester [1999] with early ideas by Shawe-Taylor and Williamson [1997], the particular bound we use is due to Maurer [2004]. A similar result was first shown by Langford and Seeger [2001] for 01-loss. In experiments, we use data-dependent priors, perhaps first conceptualized by Ambroladze, Parrado-Hernández and Shawe-Taylor [2007], Parrado-Hernández et al. [2012]. Besides the previously discussed work of Germain et al., PAC-Bayes has also been used in theories for transfer learning [Li and Bilmes, 2007, McNamara and Balcan, 2017]. As mentioned, our bounds are the first PAC-Bayesian multiclass adaptation bounds.

C EXPERIMENTAL DETAILS

C.1 DATASETS AND MODELS

As noted in the main text, we consider a collection of common adaptation datasets from both computer vision and NLP. Each dataset consists of a number of component *domains* which are themselves distinct datasets that all share a common label space. In this way, we can simulate transfer of some model from one domain to another. The datasets and models we consider are as follows:

1. **Digits**: Digits consists of collection of digit classification datasets including: USPS [Hull, 1994], MNIST [LeCun and Cortes, 2010], and SVHN [Netzer et al., 2011]. We use only the training sets. The number images in each is about 7K, 60K, and 70K, respectively. We select $\mathcal{X}$ to be the space of 28×28 grayscale images (i.e., the original feature space for MNIST). For USPS and SVHN, this is accomplished through image transformation. The label space $\mathcal{Y}$ consists of the digits 0-9. As we are aware, this collection was first used by Ganin and Lempitsky [2015]. For this task, we consider $\mathcal{H}$ to be a space of CNNs of a fixed 4-layer architecture.
2. **PACS**: PACS is an image-classification, domain generalization dataset where each domain has a different style. It was proposed by Li et al. [2017] to be a more challenging task compared to existing generalization datasets. The domains consist of images in style of: Photo, Art Painting, Cartoon, or Sketch. There are about 10K total labeled images with some slight imbalance in the liklihood of each style. The label space $\mathcal{Y}$ consists of 7 common object categories: dog, elephant, giraffe, guitar, horse, house, and person. The feature space $\mathcal{X}$ is selected to be space of real-vectors of dimension 2048; i.e., $\mathbb{R}^{2048}$. To map to the feature space from an image, we use the hidden-layer output of an image passed through a pre-trained ResNet-50 [He et al., 2016]. For this task, we consider $\mathcal{H}$ to be either the space of linear classifiers or the space of 4-layer FCNs of a fixed architecture (fully-connected networks).
3. **Office-Home**: Office-Home was originally proposed by Venkateswara et al. [2017], but we use the smaller preprocessed version given by Zhou et al. [2020]. The dataset is similar to PACS. It also contains 4 different styles as its component domains across about 15K total images: Art, Clipart, Product, and Real-World. Unlike PACS, it has a much larger number of classes. In particular, the label space $\mathcal{Y}$ contains 65 categories of different daily objects. Like PACS, we use the outputs of ResNet-50 to map to the real vector space $\mathcal{X}$. We let $\mathcal{H}$ be either a linear model or a 4-layer FCN as before.
4. **Amazon Reviews**: Amazon Reviews is a text-classification dataset introduced by Blitzer et al. [2007]. We use the Books, DVD, Electronics, and Kitchen domains preprocessed as in Blitzer et al. This totals about 4000 reviews which are labeled as having positive or negative sentiment. The feature space $\mathcal{X}$ is the space $\mathbb{N}^{4096}$; i.e., the space of bag-of-word representations. For each review, the non-zero vector components correspond to counts for words found within the review. Implicitly, this limits our vocabulary to the 4095 most frequent words and leaves one special token for out-of-vocabulary words. As noted, the label space $\mathcal{Y}$ is a binary space whose elements denote the sentiment of the review. We let $\mathcal{H}$ be either a linear model or a 4-layer FCN as before.
5. **Discourse A (PDTB Labels)**: The Penn Discourse Treebank (PDTB) [Prasad et al., 2008] is an NLP dataset containing a subset of Wall Street Journal articles from the Penn Treebank [Marcus, Santorini and Marcinkiewicz, 1993] which are tagged with shallow discourse coherence relations (i.e., relations that hold only between the argument pairs and do not have any hierarchy or graph structure). These coherence relations can be explicitly signaled by discourse *connectives* such as *and*, *so*, and *but*, or could require the insertion of an *implicit* connective. In this paper, we focus on the task of implicit discourse sense classification which is the most difficult task for discourse parsers. To form a DA dataset, we also used implicit relations from two parallel corpora: the TED-MDB [Zeyrek et al., 2020] which contains tagged TED talks and the BioDRB [Ramesh and Yu, 2010] which contains tagged scientific articles. These three mentioned datasets form our component domains. Our feature space $\mathcal{X}$ is selected to be space of real-vectors of dimension 728. Within this space, we try three different feature representations for the discourse relations.[6] In the first two cases, the feature is made up of the argument pairs which have been concatenated and encoded using a BERT model [Devlin et al.]. We use either the pooled output or the average of the hidden states. In the last case, we use Sentence-BERT [Reimers and Gurevych, 2019] to encode our features. Our label space $\mathcal{Y}$ consists of the 4 level 1 discourse sense classes contained in the Penn Discourse Treebank. We let $\mathcal{H}$ be either a linear model or a 4-layer FCN as before.
6. **Discourse B (GUM Labels)**: The GUM corpus [Zeldes, 2017] contains text documents from 8 different genres: Academic, Biography, Fiction, Interview, News, Reddit, How-To, and Travel. These genres form our component

[6] We only experiment *within* each representation type and do not attempt to transfer *across* different representations.

domains. Documents within the corpus are annotated using the discourse framework of Rhetorical Structure Theory [Mann and Thompson, 1987] in which discourse coherence relations are organized in a hierarchical tree structure. The sense hierarchy used for the GUM corpus is similar to that of the RST Discourse Treebank [Carlson, Marcu and Okurowski, 2003]. In order to focus on coherence relations only between two argument pairs (without the additional hierarchical structure), we removed all relations where one or both nodes was not a leaf node. To form the label space, we mapped the twenty GUM labels to the conventional RST discourse treebank top-level labels where only three GUM labels did not have an existing mapping encoded in the RST. We mapped these three in the following manner, following Braud, Coavoux and Søgaard [2017]: *preparation* to BACKGROUND, *justify* and *motivation* to EXPLANATION, and *solutionhood* to TOPIC-COMMENT. Given this mapping, our final label space $\mathcal{Y}$ consists of the 13 different RST discourse sense classes that were mapped to by the GUM corpus classes. Our features are encoded the same way as the PDTB features. We let $\mathcal{H}$ be either a linear model or a 4-layer FCN as before.

Random Data Splits To simulate variability due to sampling and also to consider more mild forms of dataset shift, we split each component domain within each dataset into two disjoint sets of (roughly) equal size. So, for a dataset with 4 component domains, the new number of component domains will be 8. Some of these component domains *should* now follow a fairly similiar distribution; i.e., splits coming from the same original component. This process is done randomly and all adaptation scenarios (see Section C.4) test 3 different seeds for this split.

C.2 MODEL TRAINING FOR DIVERGENCE APPROXIMATION, ADAPTABILITY UPPERBOUNDS, AND SIMPLE ALGORITHM (SA)

We train a number of deterministic models throughout our experimentation; e.g., for divergence approximation, to compute risks for the ranking task, and to compute upperbounds on λ. To avoid individual parameter selection for more than 12,000 models all trained in our experimentation, we use the optimization parameters given below in most cases. For Gibbs predictors, we use a slightly modified technique which is also discussed below. For the **DANN** algorithm, we use different parameters which were more carefully selected (details discussed in Section C.3). While this "one size fits all" approach is arguably simplistic, we found these settings worked well for a majority of cases in our preliminary experiments. For divergence approximation and upperbounds on adaptability this is visible in the main text results. In Appendix D.2, we also report statistics on the transfer (target) error of some hypotheses trained to minimize error on the source sample (i.e., the Simple Algorithm **SA**). These provide a sanity check that our simple optimization procedure is indeed selecting non-trivial hypotheses in a majority of cases. Notably, the point of this work is to study global trends rather than to achieve optimal performance on any one dataset. The "one size fits all" approach we take is reflective of this; it allows us to use our limited computational resources to study *more* datasets and models, rather than do rigorous parameter search on just a few.

Optimization Parameters All models are trained using SGD on an NLL loss with momentum set to 0.9. The NLL loss is sometimes weighted to correctly replicate the importance of multiple risks (e.g., when minimizing a sum of two risks). For example, if we have the objective $\min_h \mathbf{R}_S(h) + \mathbf{R}_T(h)$ and S has more samples than T, the NLL loss will weight examples in T higher to give them equal importance during optimization, as described by the objective. We start training with a learning of 1×10^{-2} for 100 epochs. Then, we train for another 50 epochs using a learning rate of 1×10^{-3}. If a model ever achieves a training error lower than 5×10^{-4}, we terminate training. In all cases, we use a batch size of 250.

Gibbs Predictors To learn a Gibbs predictor (stochastic model) $\mathbb{Q}$ we need to use a slightly different approach. In all cases, $\mathbb{Q}$ will be a multivariate normal distribution with diagonal covariance and we will minimize Gibbs risk on a source sample S with intention to transfer to a target sample T. We use PAC-Bayes-by-Backprop (PBB) to learn the parameters of our normal distribution. PBB is an SGD-based technique proposed by Pérez-Ortiz et al. [2021] to learn stochastic models that optimize PAC-Bayes bounds. The approach requires specification of a particular PAC-Bayes bound to use as the objective and a particular distribution $\mathbb{P}$ to use as the prior. For the former, we use the variational bound proposed by Dziugaite et al. [2021]. For the latter, we use a multivariate normal distribution: the mean is a (trained) deterministic model (i.e., its parameter vector) and the covariance matrix is $\sigma\mathbf{I}$ where $\sigma = 0.01$ and $\mathbf{I}$ is the identity matrix. We train the deterministic model to minimize the error on S using the same optimization parameters discussed previously. Note, this may seem taboo to one familiar with PAC-Bayes, since the prior $\mathbb{P}$ is typically required to be independent of the data used in the bound. Contrary to this, in our setting, it is *perfectly valid* to select $\mathbb{P}$ based on the data in (only) S. This is clear in the proofs of Theorem 2 and Theorem 6 because the prior $\mathbb{P}$ is only used to bound the generalization gap between T and $\mathbb{T}$. Thus, this choice is reflective of a realistic scenario where one wishes to compute a PAC-Bayes bound with $\mathbb{Q}$. The approach we describe essentially corresponds to the idea of using a data-dependent prior (see Section B). The prior $\mathbb{P}$ is learned using data that is not used in

any part of the bound which depends on $\mathbb{P}$. For optimization parameters of PBB not discussed here (e.g., learning rate), we default to the previously discussed choices.

C.3 DANN MODEL TRAINING

For **Digits**, we study a PAC-Bayes variant of the invariant feature learning algorithm **DANN** (Domain Adversarial Neural Network) proposed by Ganin and Lempitsky [2015]. The output of our variation is a Gibbs predictor $\mathbb{Q}$, so we again employ PBB as discussed above in Section C.2. While many parameters are similar to those used above – including the prior and the PAC-Bayes bound –, we highlight some differences here. Most notably, we re-weight the KL divergence in the PAC-Bayes bound by a dampening factor to reduce its regularizing impact during training. For example, if the KL divergence was 46K and the dampening factor was 0.1, the effective KL divergence during training is 4.6K. We explore a range of different dampening factors to get a breadth of different "complexities" for interpretation; i.e., this variability produces the movement along the horizontal axes of Figure 2. In our experiments, we let the dampening factor range in the set $\{0.1, 0.05, 0.01, 0\}$. We did not use any dampening factor to recover the original PAC-Bayes bound (i.e., 1) because we found this setting to be too restrictive in preliminary experiments. Due to the increased training time involved in this parameter sweep, we down-sampled the **Digits** dataset discussed above so that neither S nor T have more than 5K examples. We selected the learning rate by manual inspection, varying the learning rate (and number of epochs accordingly) until we did not observe frequent gradient explosion / vanishing. We ended up using an initial learning rate of 1×10^{-3} for 112 epochs (75% of 150) and 5×10^{-4} for the remaining 38 epochs. We also reduced the batch size to 128, but most other parameters remained the same. It is important to note that the instability we experienced (i.e., related to gradient explosion / vanishing) is somewhat common when training adversarial methods such as DANN. As an additional measure to combat this issue, we slowly eased in the adversarial loss by weighting it using the parameter $\beta_p = 2/(1 + \exp(-10p))$ where p is the progress ratio of the current epoch in training; e.g., $p = 0.1$ corresponds epoch 15 out of 150. This approach was first proposed by Ganin and Lempitsky [2015] for the same purpose. Besides the adversarial loss, we also multiply the KL divergence dampening factor by this weight to ease in the regularization component as well. As a final measure, we used multiple restarts with a new neural network initialization (up to 25 attempts), which proved to be the most effective measure. Among about 32K statistics computed during these experiments, only 21 statistics were unable to be computed due to instability. These consisted of restricted model-independent divergences (i.e., using class $[\mathcal{H}\Delta\mathcal{H}]_\mu$) and were ignored in plots. Roughly 5% of the data points still had "extreme" values that did not match any other trend, so we removed these in Figure 2 to help with visual interpretation.

C.4 ADAPTATION PAIRS

We now discuss the different adaptation scenarios we consider. Instances of each scenario produce the collection of (S, T) pairs we consider in our histograms in the main text. Recall, we randomly split each component domain into two halves (see Section C.1). This will be important for understanding our adaptation scenarios.

Single-Source For all datasets except **Discourse B**, we consider a single-source adaptation scenario: each component domain in a dataset is paired with each distinct component domain. So, for a dataset with 8 components, this forms 64 (S, T) pairs. For example, one pair might take S to be the first random half of SVHN and T to be the first random half of USPS. Another pair might take S to be the first random half of SVHN and T to be the *second* random half of SVHN. So, as we see from this example, this implies that components derived from the same domain (i.e., through our random splitting procedure) will be paired. Note, these should follow a fairly similar (or identical) distribution. This is purposeful and provides a number of instances of **within-distribution** shift. These milder forms of shift allow us to test a broader range of realistic scenarios. The random splitting procedure also allows us to test variability in the outcome of a transfer task due to sampling; e.g., the first and second random half of SVHN will both be paired with every other dataset.

Multi-Source We also consider multi-source scenarios for all datasets except **Discourse A**. In these cases, we group all but one component of the dataset into a single pooled sample. The single component which was left out is chosen to be the target. So, for a dataset with 4 components, this forms 4 (S, T) pairs. For example, for PACS, one pair might take S to be the union of Art, Cartoon, and Sketch while T consists of only Photo. **Importantly**, we only use *one* of the random splits from each component type; e.g., for PACS, although we split photo into two disjoint sets, we only use one of these two sets. Otherwise, in every (S, T) pair, S would contain some data coming from a similar distribution as T. Informally speaking, this likely to weaken the adaptation difficulty. Note, the adaptation bounds we give implicitly cover multi-source contexts, since we can view the single source $\mathbb{S}$ as a mixture distribution.

Digits-Specific Scenarios The **Digits** dataset is particular interesting because the feature space $\mathcal{X}$ is well-understood by humans. Thus, we can use our experience to design some natural distribution shifts. In particular, we consider the case where S is some component of **Digits** and T is the same sample except every image is randomly rotated up to $360°$. We also consider the case where S is some component of **Digits** and T is the same sample except every image is blurred with random white noise. We can also consider a very unnatural shift. In particular, we consider transfer to randomly generated data. Here, S is some component of **Digits** (as before) and every image in T is a 28×28 grid of randomly generated pixels which is assigned a random label. For these scenarios, we use the entirety of the components in the **Digits** sets without doing any random splitting.

C.5 DETAILS FOR FOR FIGURE 1

Each dataum in Figure 1 corresponds to an upperbound for one of the adaptation pairs described in Section C.4 using one of the compatible hypothesis spaces described in Section C.1. We describe the process for computing each type of upperbound below. In all cases, we compute the upperbound using 3 different random seeds; i.e., this will effect things like the model-training and subsequently the final bound. We report the smallest upperbound of these seeds. This is logical since the smallest upperbound is still a valid upperbound. In case of λ, this is actually overly optimistic since the confidence parameter should be changed to account for all 3 bounds.

Upperbound for $\tilde{\lambda}$ This bound is computed just as described in the main text. For each adaptation pair (S, T) and each hypothesis space $\mathcal{H}$ which is compatible with S and T, we train a model to minimize the summed risks on S and T using the approach described in Section C.2. If this approach returns the hypothesis h, we report $\mathbf{R}_S(h) + \mathbf{R}_T(h)$. As noted in the main text, this is a valid upperbound for $\tilde{\lambda}$.

Upperbound for λ For each adaptation pair (S, T) and each hypothesis space $\mathcal{H}$ which is compatible with S and T, we randomly split S and T using an 80/20 train/test split. Denote train splits for S and T by S_{tr} and T_{tr}, respectively. Denote the test splits for S and T by S_{ho} and T_{ho}, respectively. We train a model to minimize summed risks on S_{tr} and T_{tr} using the approach described in Section C.2. If this approach returns the hypothesis h, we then report the quantity

$$\mathbf{R}_{S_{\text{ho}}}(h) + \mathbf{R}_{T_{\text{ho}}}(h) + \sqrt{\ln(4/\delta)/(2m)} + \sqrt{\ln(4/\delta)/(2n)} \tag{63}$$

where $m = |T_{\text{ho}}|$, $n = |S_{\text{ho}}|$, and $\delta = 0.05$. This is a valid upperbound for λ which holds (i.e., prior to observing data) with probability $1 - \delta$. It is easily derived using Hoeffding's Inequality to bound both $\Delta_h(S_{\text{ho}}, \mathbb{S})$ and $\Delta_h(T_{\text{ho}}, \mathbb{T})$, and then, using Boole's Inequality to combine the bounds.

C.6 DETAILS FOR FOR TABLE 1

Each dataum used to compute the correlations in Table 1 corresponds to a divergence approximation for an (S, T, h) triple. The datasets S and T are given by one of the adaptation pairs described in Section C.4, and conceptually, we pick h to be the hypothesis whose error we would like to bound in the simulated transfer from S to T. Specifically, for each adaptation pair (S, T) and each compatible hypothesis space $\mathcal{H}$, we select h using **SA**. We then approximate either the $\mathcal{H}\Delta\mathcal{H}$- or $h\Delta\mathcal{H}$-divergence using the (appropriate) technique described in Section 3.2. The Spearman rank correlation we report compares the $\mathcal{H}\Delta\mathcal{H}$- and the $h\Delta\mathcal{H}$-divergence to the error-gap $\Delta_h(S, T)$ defined in Eq. (27) over all adaptation pairs (S, T) in Section C.4, all compatible hypothesis spaces $\mathcal{H}$ discussed in Section C.1, and all 3 seeds.

C.7 DETAILS FOR FOR ESTIMATION OF FLATNESS

Each datum discussed in the main text paragraph **Do Flat Regions Transfer?** corresponds to an estimate for an $(S, T, \mathbb{Q})$ triple. The datasets S and T are given by one of the adaptation pairs described above, and as before, we pick $\mathbb{Q}$ to be the Gibbs predictor whose error we would like to bound in the simulated transfer from S to T (i.e., using **SA**). We then estimate as described in the main text. For each (S, T) pair and compatible hypothesis space $\mathcal{H}$, we repeat this procedure with 3 seeds to control for variability in the selection of $\mathbb{Q}$. We report all seeds in the histogram in Figure 5. Also, the mean and standard deviation reported in the main text are computed using all data in the histogram.

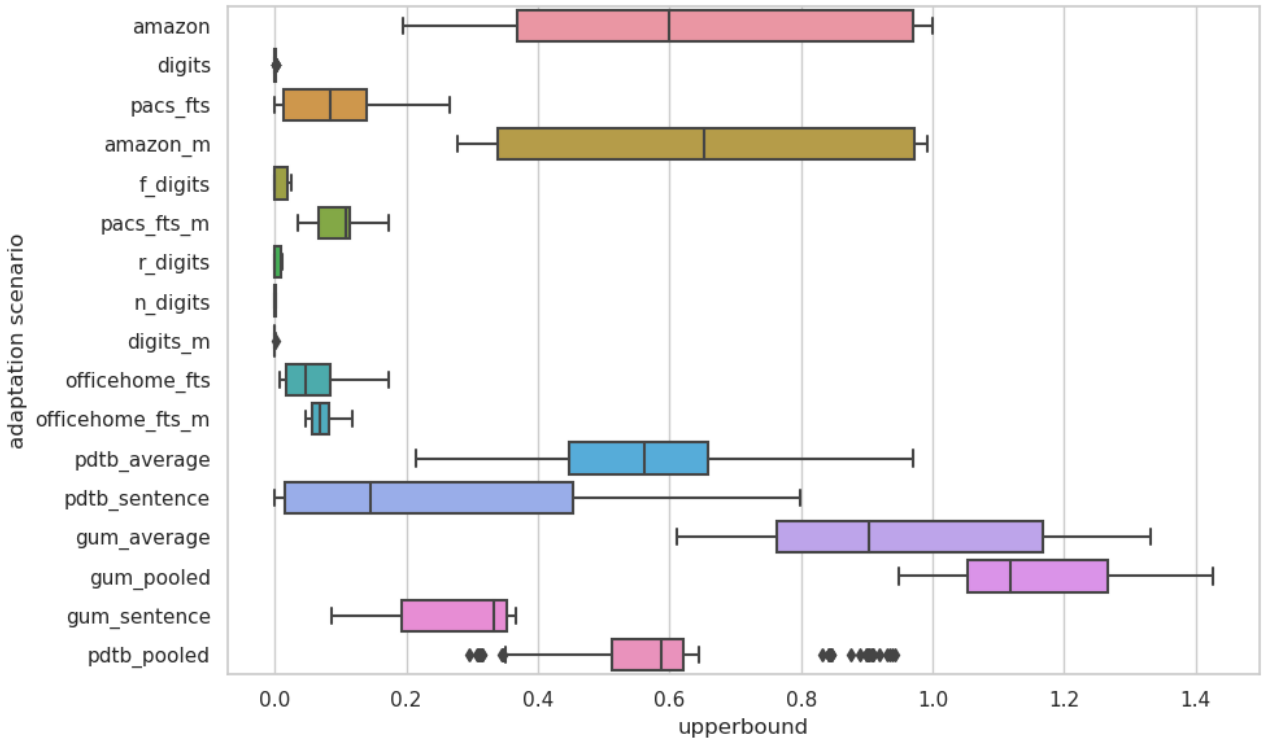

Figure 3: Boxplots for upperbounds on $\tilde{\lambda}$ for individual datasets. The appendage `_m` denotes the **multi-source** setup, otherwise it is **single-source**. For digits, we prepend `r_`, `n_`, or `f_` to denote transfer to rotated, noisy, or randomly generated (fake) data as discussed in Section C.4. As may be inferred, `pdtb` corresponds to **Discourse A** and `gum` corresponds to **Discourse B**. Appendages for these indicate the type of BERT features used.

C.8 DETAILS FOR FIGURE 2

Each datum in Figure 2 corresponds to statistics computed for an $(S, T, \mathbb{Q})$ triple. As noted, S and T are restricted to be samples from the **Digits** dataset. Further, we only consider out-of-distribution adaptation scenarios as indicated in each subfigure title. The Gibbs predictor $\mathbb{Q}$ is selected using the training details discussed in Section C.3 and the statistics are reported as described in Thm. 6 and Cor. 1.

D ADDITIONAL EXPERIMENTAL RESULTS

D.1 UPPERBOUNDS ON ADAPTABILITY (DATASET SPECIFIC)

We also found it interesting to consider our sample-dependent adaptability in a more problem-specific context. This reveals to us that **PACS** and **Office-Home** have the larger upperbounds of computer vision datasets. It also reveals that most upperbounds above about 0.3 are due to NLP datasets. Informally speaking, this is sensible as the NLP tasks we consider have higher uncertainty in the labeling functions. Results are shown in Figure 3. We use the same aggregate data as Figure 1.

D.2 TRANSFER ERROR OF TRAINED AND RANDOM HYPOTHESES

In Figure 4, we show transfer error (i.e., the error on T) of hypotheses trained using **SA**. These are precisely the hypotheses used to compute the error-gap $\Delta_h(S, T)$ when reporting Spearman rank correlation. As noted, we use a standardized optimization procedure to forego parameter selection on the more than 12,000 models we train. Thus, we are not interested in optimal performance in any case. Instead, Figure 4 primarily serves as a sanity check to make sure the hypotheses we use are somewhat reflective of those which might be used in a practical scenario. That is, we would like to confirm that these

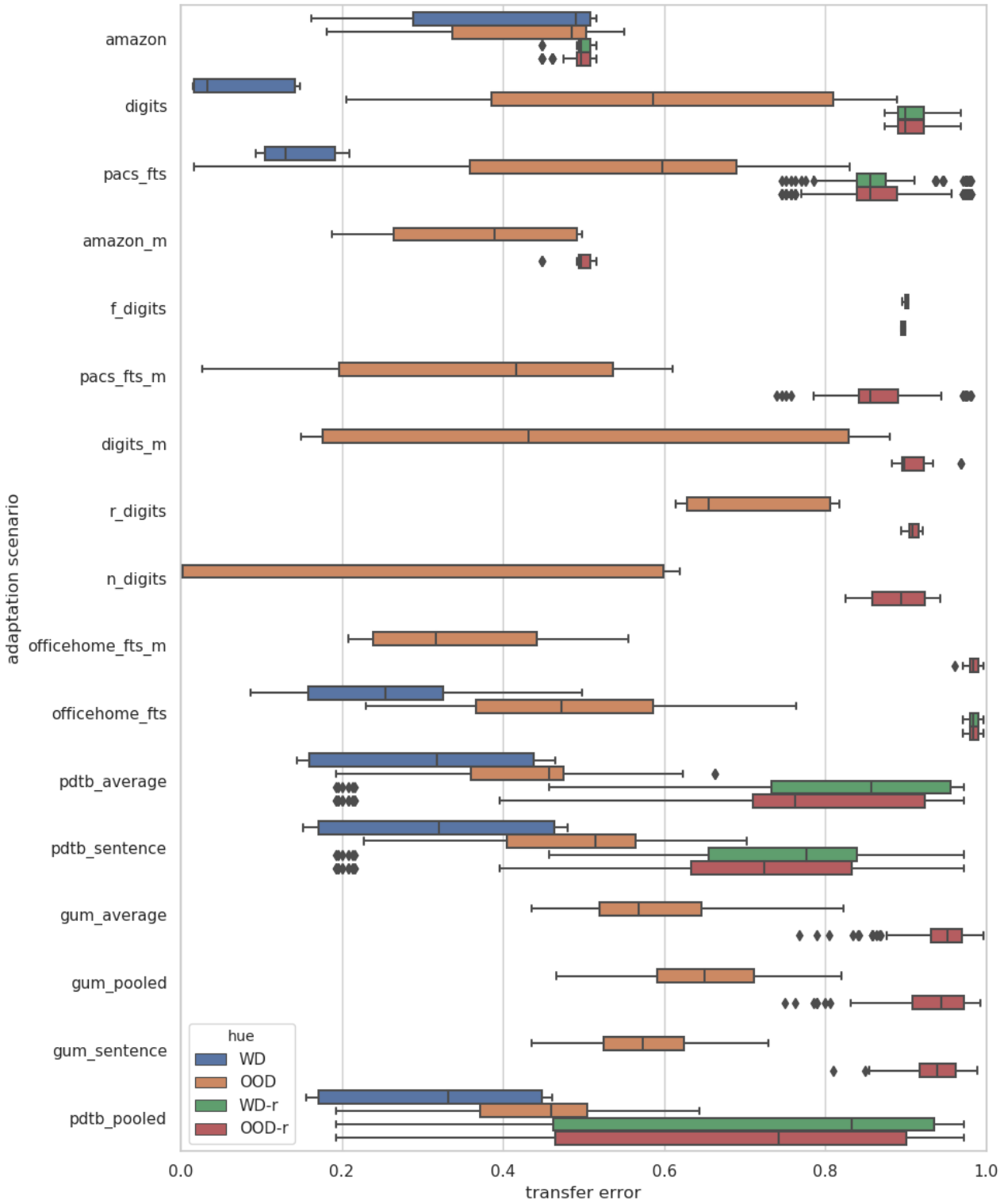

Figure 4: Dataset names are as in Figure 3. Hues correspond to within-distribution **WD** and out-of-distribution **OOD** based on whether the target T is drawn from the same component domain as the source S. Notice, **WD** is not available for multi-source setups and certain Digits setups. This is simply a property of these adaptation scenarios. In these cases, we still report the **OOD** error. The appendage `-r` denotes the hypothesis is randomly initialized and not trained. These experiments provide a point of reference for comparison.

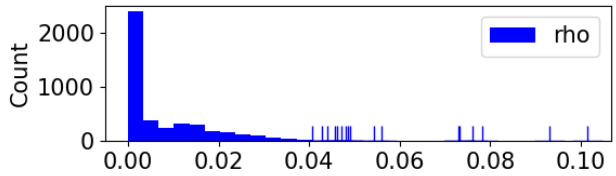

Figure 5: Estimates of ρ as described in main text and Section C.7. Each datum describes unique $(S, T, \mathbb{Q})$. Visualization confirms that ρ is very often small as discussed in the main text.

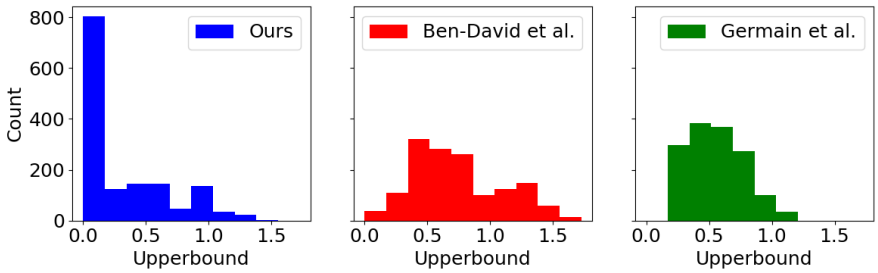

Figure 6: Upperbounds for sample-dependent (left), sample-independent (center), and binary PAC-Bayes variant (right) of adaptability. Each datum describes unique $(S, T, \mathcal{H})$.

hypotheses have learned something non-trivial (at least on the source domain). To illustrate this, for all datasets, we also report the error of randomly initialized hypotheses which have not been trained. This provides a point of reference. It is easy to see our trained hypotheses are typically far more effective than the untrained random initializations.

For the discourse datasets with PDTB labels, we observe the error of the random initializations is somewhat harder to interpret. For this reason we compare to a related work. In particular, Kishimoto, Murawaki and Kurohashi [2020] achieve an error rate of ≈ 0.38 on a comparable (within-distribution) discourse sense classification task. We observe our within-distribution PDTB results (Discourse A) are frequently better than this.

D.3 COMPARISON TO GERMAIN ET AL.

While we know, theoretically, the PAC-Bayes bound of Germain et al. [2020] is not valid for the multiclass setting, we also study this question empirically. In Figure 6, we present a sample-dependent variation of the adaptability proposed by Germain et al.[7] because of our previous (positive) results on sample-dependence. Even with this upgrade, the adaptability of Germain et al. is not able to capture the same useful information as our multiclass sample-dependent adaptability. Further, conducting a similar experiment as in Table 1, we find the divergence term of Germain et al. has low rank correlation (**0.27** on all data). These empirical results confirm the hypothesis of Germain et al. [2020] that their PAC-Bayesian theory of adaptation in binary settings is not easily extend to the multiclass setting. This is especially true in comparison to the positive outcomes observed under our theory. The experimental details for these results are provided below.

Upperbound for Adaptability of Germain et al. [2020] The historgram in Figure 6 shows a histogram of upperbounds on a sample-dependent variation of the adaptability of Germain et al. [2020] given in Thm. 7. We compute the upperbounds using the same setup as described in Appendix C.5. Because we do not have full access to $\mathbb{Q}$, we instead estimate this term with a finite sample $Q = (H_i)_i \sim \mathbb{Q}^k$. Using Linearity of $\mathbf{E}$ and Hoeffding's Inequality, we have the following bound on Germain et al.'s adaptability (our sample-dependent variant) with i.i.d. sample $(H_{i,1}, H_{i,2})_i \sim (\mathbb{Q} \times \mathbb{Q})^k$

$$\left| k^{-1} \sum_i \mathop{\mathbf{E}}_{X,Y} [1[H_{i,1}(X) \neq Y] \cdot 1[H_{i,2}(X) \neq Y]] - k^{-1} \sum_i \mathop{\mathbf{E}}_{\tilde{X},\tilde{Y}} [1[H_{i,1}(\tilde{X}) \neq \tilde{Y}] \cdot 1[H_{i,2}(\tilde{X}) \neq \tilde{Y}]] \right| + \sqrt{\frac{2\ln(2/\delta)}{k}}. \quad (64)$$

Here, we pick $\delta = 0.05$ as before and use $k = 100$. This gives a valid bound for which holds with probability $1 - \delta$ (i.e., prior to seeing the samples from $\mathbb{Q}$). We select the Gibbs predictor $\mathbb{Q}$ using **SA** as before.

[7] Some proof-techniques we employ in Thm. 2 can be applied to derive a sample-dependent variation of Thm. 7, instead.

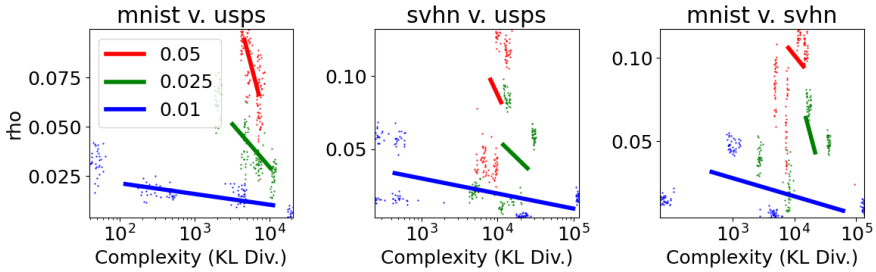

Figure 7: Estimates for ρ while using **DANN** for various choices of the prior variance parameter σ. Solid line shows median, while scatter shows 95% or more of data. Each datum describes unique $(S, T, \mathbb{Q})$. Sensibly, smaller prior variance helps to ensure small ρ. As a function of complexity, we can expect ρ to be smaller for more complex solutions. This is similar to our findings in the main text on adaptability after **DANN**.

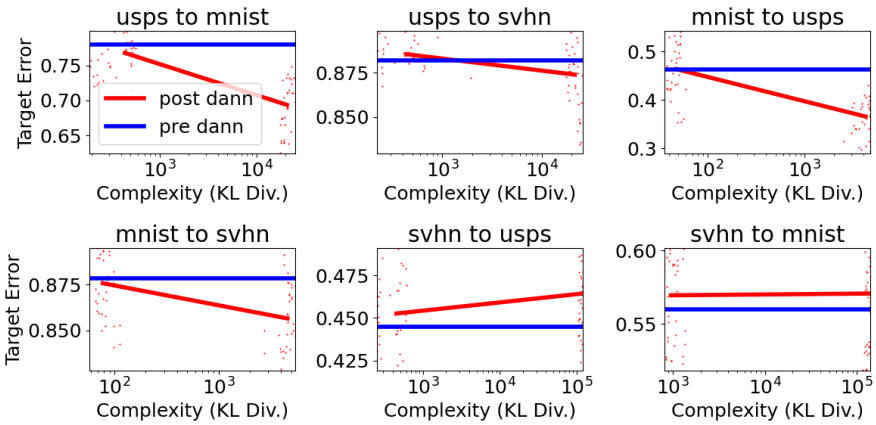

Figure 8: Estimates of target error $\mathbf{R}_T(\mathbb{Q})$ (after **DANN**) compared to prior error $\mathbf{R}_T(\mathbb{P})$ (before **DANN**, i.e. using **SA**). Solid line shows median, while scatter shows 95% or more of data. Each datum describes unique $(S, T, \mathbb{Q})$. More complex solutions achieve lower error as expected. **DANN** is effective at reducing target error in many cases.

Divergence of Germain et al. [2020] We also approximate the divergence of Germain et al. to compare to the $\mathcal{H}\Delta\mathcal{H}$- and $h\Delta\mathcal{H}$-divergence in terms of model selection. As noted, the comparison is made through Spearman rank correlation with error-gap Δ using the same experimental setup as in Appendix C.6. Since we are not aware of an analytic solution for this divergence (in case of neural networks), we approximate the divergence term of Germain et al. using a random sample $Q \sim \mathbb{Q}^k$ with $k = 100$. Here, $\mathbb{Q}$ is a distribution over $\mathcal{H}$ selected, again, using **SA**. We do this for each adaptation pair (S, T) and each compatible hypothesis space $\mathcal{H}$. The final reported correlation compares the approximated divergence to $\Delta_Q(S, T)$ over all adaptation pairs, all compatible hypothesis spaces, and all 3 seeds.

D.4 ADDITIONAL DANN RESULTS

In Figure 7, we show the effect of **DANN** on ρ. This confirms our takeaway in the main text that, as a function of sample complexity, ρ behaves like adaptability. Also, we see that increasing the prior variance makes flatness less likely, since this indirectly controls the variance of $\mathbb{Q}$, which is regularized to be similar to the prior via PBB (see Section C.3). Intuitively, it is easier to find small flat-minima than very large flat-minima. In Figure 8, we show the target (transfer) error of solutions trained using **DANN** compared to solutions trained using **SA**. The error rates may appear unusually high to familiar readers (e.g., compared to Ganin and Lempitsky [2015]), but this is likely a result of the down-sampling we do to save training time (see Section C.3). Since **DANN** is a more sophisticated adaptation algorithm, we expect it to learn more about the target than **SA**, and indeed, it does in many contexts. Thus, this result also serves to validate our empirical setup for applying **DANN**. Lastly, as before, we can interpret the target error as a function of sample complexity: unconstrained solutions are able to achieve lower error than constrained solutions. Due to the higher complexity, these solutions may not generalize well.

References

Albuquerque, Isabela, João Monteiro, Tiago H Falk and Ioannis Mitliagkas. 2020. "Adversarial target-invariant representation learning for domain generalization." *arXiv:1911.00804* .

Ambroladze, Amiran, Emilio Parrado-Hernández and John Shawe-Taylor. 2007. "Tighter pac-bayes bounds." *NeurIPS* 19:9.

Ben-David, Shai, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira and Jennifer Wortman Vaughan. 2010. "A theory of learning from different domains." *Machine learning* 79(1):151–175.

Ben-David, Shai, John Blitzer, Koby Crammer, Fernando Pereira et al. 2007. "Analysis of representations for domain adaptation." *NeurIPS* 19:137.

Ben-David, Shai, Tyler Lu, Teresa Luu and Dávid Pál. 2010. Impossibility theorems for domain adaptation. In *AISTATS*. JMLR Workshop and Conference Proceedings pp. 129–136.

Blanchard, Gilles, Aniket Anand Deshmukh, Ürün Dogan, Gyemin Lee and Clayton Scott. 2021. "Domain Generalization by Marginal Transfer Learning." *JMLR* 22:2–1.

Blitzer, John, Mark Dredze and Fernando Pereira. 2007. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *ACL*. pp. 440–447.

Braud, Chloé, Maximin Coavoux and Anders Søgaard. 2017. "Cross-lingual RST discourse parsing." *arXiv:1701.02946* .

Carlson, Lynn, Daniel Marcu and Mary Ellen Okurowski. 2003. Building a discourse-tagged corpus in the framework of rhetorical structure theory. In *Current and new directions in discourse and dialogue*. Springer pp. 85–112.

Catoni, Olivier. 2007. "PAC-Bayesian supervised classification: the thermodynamics of statistical learning." *arXiv:0712.0248v1* .

Crammer, Koby, Michael Kearns and Jennifer Wortman. 2007. Learning from multiple sources. In *NeurIPS*. pp. 321–328.

Deng, Zhun, Frances Ding, Cynthia Dwork, Rachel Hong, Giovanni Parmigiani, Prasad Patil and Pragya Sur. 2020. "Representation via representations: Domain generalization via adversarially learned invariant representations." *arXiv:2006.11478* .

Devlin, Jacob, Ming-Wei Chang, Kenton Lee and Kristina Toutanova. N.d. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*. ACL pp. 4171–4186.

Dziugaite, Gintare Karolina and Daniel M Roy. 2017. "Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data." *arXiv:1703.11008* .

Dziugaite, Gintare Karolina, Kyle Hsu, Waseem Gharbieh, Gabriel Arpino and Daniel Roy. 2021. On the role of data in PAC-Bayes. In *AISTATS*. PMLR pp. 604–612.

Ganin, Yaroslav and Victor Lempitsky. 2015. Unsupervised domain adaptation by backpropagation. In *ICML*. PMLR pp. 1180–1189.

Germain, Pascal, Alexandre Lacasse, Francois Laviolette, Mario March and Jean-Francis Roy. 2015. "Risk Bounds for the Majority Vote: From a PAC-Bayesian Analysis to a Learning Algorithm." *JMLR* 16:787–860.

Germain, Pascal, Alexandre Lacasse, François Laviolette and Mario Marchand. 2009. PAC-Bayesian learning of linear classifiers. In *ICML*.

Germain, Pascal, Amaury Habrard, François Laviolette and Emilie Morvant. 2020. "PAC-Bayes and domain adaptation." *Neurocomputing* 379:379–397.

Gretton, Arthur, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf and Alexander Smola. 2012. "A kernel two-sample test." *JMLR* 13(1):723–773.

Guedj, Benjamin. 2019. "A primer on PAC-Bayesian learning." *arXiv:1901.05353v3* .

He, Kaiming, Xiangyu Zhang, Shaoqing Ren and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*.

Hull, J. J. 1994. "A database for handwritten text recognition research." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 16(5):550–554.

Johansson, Fredrik D, David Sontag and Rajesh Ranganath. 2019. Support and invertibility in domain-invariant representations. In *AISTATS*. PMLR pp. 527–536.

Kishimoto, Yudai, Yugo Murawaki and Sadao Kurohashi. 2020. Adapting BERT to Implicit Discourse Relation Classification with a Focus on Discourse Connectives. In *LREC*. Marseille, France: European Language Resources Association pp. 1152–1158.
URL: *https://aclanthology.org/2020.lrec-1.145*

Kuroki, Seiichi, Nontawat Charoenphakdee, Han Bao, Junya Honda, Issei Sato and Masashi Sugiyama. 2019. Unsupervised domain adaptation based on source-guided discrepancy. In *AAAI*. Vol. 33 pp. 4122–4129.

Langford, John and Matthias Seeger. 2001. "Bounds for averaging classifiers.".

LeCun, Yann and Corinna Cortes. 2010. "MNIST handwritten digit database.".
URL: *http://yann.lecun.com/exdb/mnist/*

Li, Da, Yongxin Yang, Yi-Zhe Song and Timothy M Hospedales. 2017. Deeper, broader and artier domain generalization. In *IEEE ICCV*. pp. 5542–5550.

Li, Xiao and Jeff Bilmes. 2007. A bayesian divergence prior for classiffier adaptation. In *AISTATS*. PMLR pp. 275–282.

Lipton, Zachary, Yu-Xiang Wang and Alexander Smola. 2018. Detecting and correcting for label shift with black box predictors. In *ICML*. PMLR pp. 3122–3130.

Magliacane, Sara, Thijs van Ommen, Tom Claassen, Stephan Bongers, Philip Versteeg and Joris M Mooij. 2018. Domain Adaptation by Using Causal Inference to Predict Invariant Conditional Distributions. In *NeurIPS*.

Mann, William C and Sandra A Thompson. 1987. *Rhetorical structure theory: A theory of text organization*. University of Southern California, Information Sciences Institute Los Angeles.

Mansour, Yishay, Mehryar Mohri and Afshin Rostamizadeh. 2009. "Domain adaptation: Learning bounds and algorithms." *arXiv:0902.3430* .

Marcus, Mitchell, Beatrice Santorini and Mary Ann Marcinkiewicz. 1993. "Building a large annotated corpus of English: The Penn Treebank.".

Maurer, Andreas. 2004. "A note on the PAC Bayesian theorem." *arXiv cs/0411099* .

McAllester, David. 2013. "A PAC-Bayesian tutorial with a dropout bound." *arXiv:1307.2118v1* .

McAllester, David A. 1999. "Some PAC-Bayesian theorems." *Machine Learning* 37:355–363.

McNamara, Daniel and Maria-Florina Balcan. 2017. Risk bounds for transferring representations with and without fine-tuning. In *ICML*. PMLR pp. 2373–2381.

Netzer, Yuval, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu and Andrew Y Ng. 2011. "Reading digits in natural images with unsupervised feature learning.".

Parrado-Hernández, Emilio, Amiran Ambroladze, John Shawe-Taylor and Shiliang Sun. 2012. "PAC-Bayes bounds with data dependent priors." *JMLR* 13:3507–3531.

Pérez-Ortiz, Marıa, Omar Rivasplata, John Shawe-Taylor and Csaba Szepesvári. 2021. "Tighter risk certificates for neural networks." *JMLR* 22.

Prasad, Rashmi, Nikhil Dinesh, Alan Lee, Eleni Miltsakaki, Livio Robaldo, Aravind K Joshi and Bonnie L Webber. 2008. The Penn Discourse TreeBank 2.0. In *LREC*. Citeseer.

Ramesh, Balaji Polepalli and Hong Yu. 2010. Identifying discourse connectives in biomedical text. In *AMIA Annual Symposium Proceedings*. Vol. 2010 American Medical Informatics Association p. 657.

Redko, Ievgen, Amaury Habrard and Marc Sebban. 2017. Theoretical analysis of domain adaptation with optimal transport. In *ECML PKDD*. Springer pp. 737–753.

Redko, Ievgen, Emilie Morvant, Amaury Habrard, Marc Sebban and Younès Bennani. 2020. "A survey on domain adaptation theory." *ArXiv* abs/2004.11829.

Reimers, Nils and Iryna Gurevych. 2019. "Sentence-bert: Sentence embeddings using siamese bert-networks." *arXiv:1908.10084* .

Shalev-Shwartz, Shai and Shai Ben-David. 2014. *Understanding machine learning: From theory to algorithms*. Cambridge university press.

Shawe-Taylor, John and Robert C Williamson. 1997. A PAC analysis of a Bayesian estimator. In *COLT*.

Shen, Jian, Yanru Qu, Weinan Zhang and Yong Yu. 2018. Wasserstein distance guided representation learning for domain adaptation. In *AAAI*.

Sugiyama, Masashi, Matthias Krauledat and Klaus-Robert Müller. 2007. "Covariate shift adaptation by importance weighted cross validation." *JMLR* 8(5).

Tachet des Combes, Remi, Han Zhao, Yu-Xiang Wang and Geoffrey J Gordon. 2020. "Domain adaptation with conditional distribution matching and generalized label shift." *NeurIPS* 33.

Venkateswara, Hemanth, Jose Eusebio, Shayok Chakraborty and Sethuraman Panchanathan. 2017. Deep hashing network for unsupervised domain adaptation. In *IEEE CVPR*. pp. 5018–5027.

You, Kaichao, Ximei Wang, Mingsheng Long and Michael Jordan. 2019. Towards accurate model selection in deep unsupervised domain adaptation. In *ICML*. PMLR pp. 7124–7133.

Zeldes, Amir. 2017. "The GUM corpus: Creating multilayer resources in the classroom." *LREC* 51(3):581–612.

Zeyrek, Deniz, Amália Mendes, Yulia Grishina, Murathan Kurfalı, Samuel Gibbon and Maciej Ogrodniczuk. 2020. "TED Multilingual Discourse Bank (TED-MDB): a parallel corpus annotated in the PDTB style." *LREC* 54(2):587–613.

Zhang, Kun, Mingming Gong and Bernhard Schölkopf. 2015. Multi-source domain adaptation: A causal view. In *AAAI*.

Zhang, Yuchen, Tianle Liu, Mingsheng Long and Michael Jordan. 2019. Bridging theory and algorithm for domain adaptation. In *ICML*. PMLR pp. 7404–7413.

Zhao, Han, Remi Tachet Des Combes, Kun Zhang and Geoffrey Gordon. 2019. On learning invariant representations for domain adaptation. In *ICML*. PMLR pp. 7523–7532.

Zhou, Kaiyang, Yongxin Yang, Timothy Hospedales and Tao Xiang. 2020. "Deep Domain-Adversarial Image Generation for Domain Generalisation." *arXiv:2003.06054* .