

---

# PAC-Bayesian Domain Adaptation Bounds for Multiclass Learners (Supplementary material)

---

Anthony Sicilia<sup>1</sup>

Katherine Atwell<sup>2</sup>

Malihe Alikhani<sup>1,2</sup>

Seong Jae Hwang<sup>3</sup>

<sup>1</sup>Intelligent Systems Program, University of Pittsburgh, Pittsburgh, Pennsylvania, USA

<sup>2</sup>Department of Computer Science, University of Pittsburgh, Pittsburgh, Pennsylvania, USA

<sup>3</sup>Department of Artificial Intelligence, Yonsei University, Seoul, South Korea

## A PROOFS

### A.1 THEOREM 1

*Proof.* This is Thm. 2 of Ben-David et al. [2010a] with added bound on  $\mathbf{R}_S(h) - \mathbf{R}_S(h)$  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.  $\square$

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

**Theorem A.** [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  $\mathcal{H}$ ,

$$\mathbf{R}_T(\mathbb{Q}) \leq \omega'(\mathbf{R}_S(\mathbb{Q}) + |\mathbf{d}_S(\mathbb{Q}) - \mathbf{d}_T(\mathbb{Q})|) + |\mathbf{e}_S(\mathbb{Q}) - \mathbf{e}_T(\mathbb{Q})| + 2\omega \frac{\text{KL}(\mathbb{Q}||\mathbb{P}) - \ln(\delta/3)}{m\omega'} + 2(\omega' - 1) \quad (25)$$

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

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

In comparison to Thm. 1, the absolute difference in disagreement  $\mathbf{d}$  is most similar to the  $\mathcal{H}\Delta\mathcal{H}$ -divergence and the absolute difference in joint-error  $\mathbf{e}$  is most similar to the adaptability  $\lambda$  [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.* As noted, 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  $\omega > 0$ .  $\square$

### 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  $h$

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

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  $\Delta$ . The intended meaning is intuitive:

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

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  $\Delta_{\mathbb{Q}}(\mathbb{S}, \mathbb{T})$  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:

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

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 et al., 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. [2010a] 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, 2010a]. 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  $(x, y) \in \mathcal{X} \times \mathcal{Y}$ ,

$$1[h(x) \neq y] \leq 1[h(x) \neq h'(x)] + 1[h'(x) \neq y] \quad (30)$$

and

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

*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.  $\square$

Note, one observation is that the function  $\tilde{d}(y, y') \stackrel{\text{def}}{=} 1[y' \neq y]$  for any arguments  $y, y' \in \mathcal{Y}$  is identical to a well-known function called the trivial metric or the discrete metric. As implied by the name, the tuple  $(\mathcal{Y}, \tilde{d})$  forms a *metric space*, and subsequently, Lemma 1 above is a simple consequence of this fact. Nonetheless, we maintain the proof above to keep our discussion relatively self-contained.

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

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

where  $\mathcal{C}_h = \mathcal{H}\Delta\mathcal{H}$  or  $\mathcal{C}_h = h\Delta\mathcal{H}$  and  $(\mathbb{D}_i)_X$  is the  $\mathcal{X}$ -marginal of  $\mathbb{D}_i$ .<sup>1</sup>

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

*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  $(x, y) \in \mathcal{X} \times \mathcal{Y}$

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

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

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

where

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

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

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

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

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

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.  $\square$

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

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

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  $T$ :

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

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

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

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

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

$$\Pr\left(\forall \mathbb{Q} : \Delta_{\mathbb{Q}}(T, \mathbb{T}) \leq \sqrt{\frac{\text{KL}(\mathbb{Q}||\mathbb{P}) + \ln\sqrt{4m} - \ln(\delta)}{2m}}\right) \geq 1 - \delta. \quad (41)$$

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

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

where the ‘‘little’’  $\text{kl}$  is the KL-divergence between Bernoulli distributions parameterized by its arguments. The above bound implies the stated result by application of Pinsker’s Inequality.  $\square$

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

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

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

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

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

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

□

### 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  $\Delta$ . By design, this notation exhibits a triangle-inequality.

*Proof.* Observe,

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

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. [2010a]. 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

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

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

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

The first equality follows by definition of absolute value, the second by law of complements, 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  $\varphi$  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<sup>2</sup>

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

From this and the fact that  $\tau$  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

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

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

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

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 proper 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.  $\square$

## 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. As a result, we observe the need to add additional constraints on the labeling function for the classification problem. Specifically, we introduce the class  $\Upsilon$  for use in our reduction. Careful attention is paid to show the 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  $\varphi$ . 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:

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

<sup>2</sup>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.

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

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

The inequality follows by the monotonicity of probability and the fact

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

Meanwhile, setting

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

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

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

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

$$\forall \varphi \in \mathcal{H} : \Pr(h(X) = \varphi(X)) + \Pr(h(\tilde{X}) \neq \varphi(\tilde{X})) = \min_{\bar{h} \in \Upsilon} \left\{ \Pr(\bar{h}(X) \neq \varphi(X)) + \Pr(h(\tilde{X}) \neq \varphi(\tilde{X})) \right\}. \quad (56)$$

To see this, it's easiest to use the definition of a set's min element as that which attains the greatest lower bound; i.e., the inf or infimum. Then, Eq. (52) implies the min upperbounds the LHS of Eq. (56), and Eq. (55) implies the min lowerbounds the LHS of Eq. (56). In combination, these bounds prove equality.

Note, an identical argument also gives,

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

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

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

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

## 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 discussed in proof of Thm. 2.

*Proof.* Observe, for any  $h_*$ ,

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

Use Lemma 3 and Lemma 4, respectively, to bound the latter two terms. Application of Boole's Inequality and selection of  $h_* = \mu$  gives the result.  $\square$

## 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 pushforwards 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  $\mu$  is the composition of a classifier  $c_{\mu}$  and a feature extractor  $f_{\mu}$ , we have

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

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  $f_{\mu} : \mathcal{X} \rightarrow \mathcal{Z}$  and  $c_{\mu} : \mathcal{Z} \rightarrow \mathcal{Y}$ , Eq. (60) follows because

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

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  $\mathcal{Q} \subseteq \mathcal{Y}^{\mathcal{Z}}$ , any function  $p : \mathcal{X} \rightarrow \mathcal{Z}$ , and any distributions  $\mathbb{S}$  and  $\mathbb{T}$  over  $\mathcal{X} \times \mathcal{Y}$ , we use the expansion below:

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

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  $\mathcal{Q} \circ p^{-1} = [\mu\Delta\mathcal{H}]_{\mu}$ .  $\square$

## 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, which in turn, bounds the 01-loss we consider. As noted, the primary difference between our work and the work of Zhang et al. [2019] is the use of uniform sample-complexity in the latter. Possibly, bounds in the latter could be extended to PAC-Bayesian contexts as well, but our choice of divergences allows us to work directly with 01-loss and avoid any loosening of the bound via the margin penalty.

**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 et al. [2010b] 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 0-1-loss. In experiments, we use data-dependent priors, perhaps first conceptualized by Ambroladze et al. [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  $28 \times 28$  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 likelihood 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-words 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 et al., 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.<sup>3</sup> 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., 2019]. 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 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 et al., 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 et al. [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 similar 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  $\lambda$ . 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.

---

<sup>3</sup>We only experiment *within* each representation type and do not attempt to transfer *across* different representations.

**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  $1 \times 10^{-2}$  for 100 epochs. Then, we train for another 50 epochs using a learning rate of  $1 \times 10^{-3}$ . If a model ever achieves a training error lower than  $5 \times 10^{-4}$ , 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  $1 \times 10^{-3}$  for 112 epochs (75% of 150) and  $5 \times 10^{-4}$  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  $360^\circ$ . 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  $28 \times 28$  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 4

Each datum in Figure 4 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  $\lambda$ , 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  $\lambda$**  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  $S_{\text{tr}}$  and  $T_{\text{tr}}$ , respectively. Denote the test splits for  $S$  and  $T$  by  $S_{\text{ho}}$  and  $T_{\text{ho}}$ , respectively. We train a model to minimize summed risks on  $S_{\text{tr}}$  and  $T_{\text{tr}}$  using the approach described in Section C.2. If this approach returns the hypothesis  $h$ , we then report the quantity

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

where  $m = |T_{\text{ho}}|$ ,  $n = |S_{\text{ho}}|$ , and  $\delta = 0.05$ . This is a valid upperbound for  $\lambda$  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 datum 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 3. Also, the mean and standard deviation reported in the main text are computed using all data in the histogram.

## 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 5. We use the same aggregate data as Figure 4.

## D.2 TRANSFER ERROR OF TRAINED AND RANDOM HYPOTHESES

In Figure 6, 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 6 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 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 et al. [2020] achieve an error rate of  $\approx 0.38$

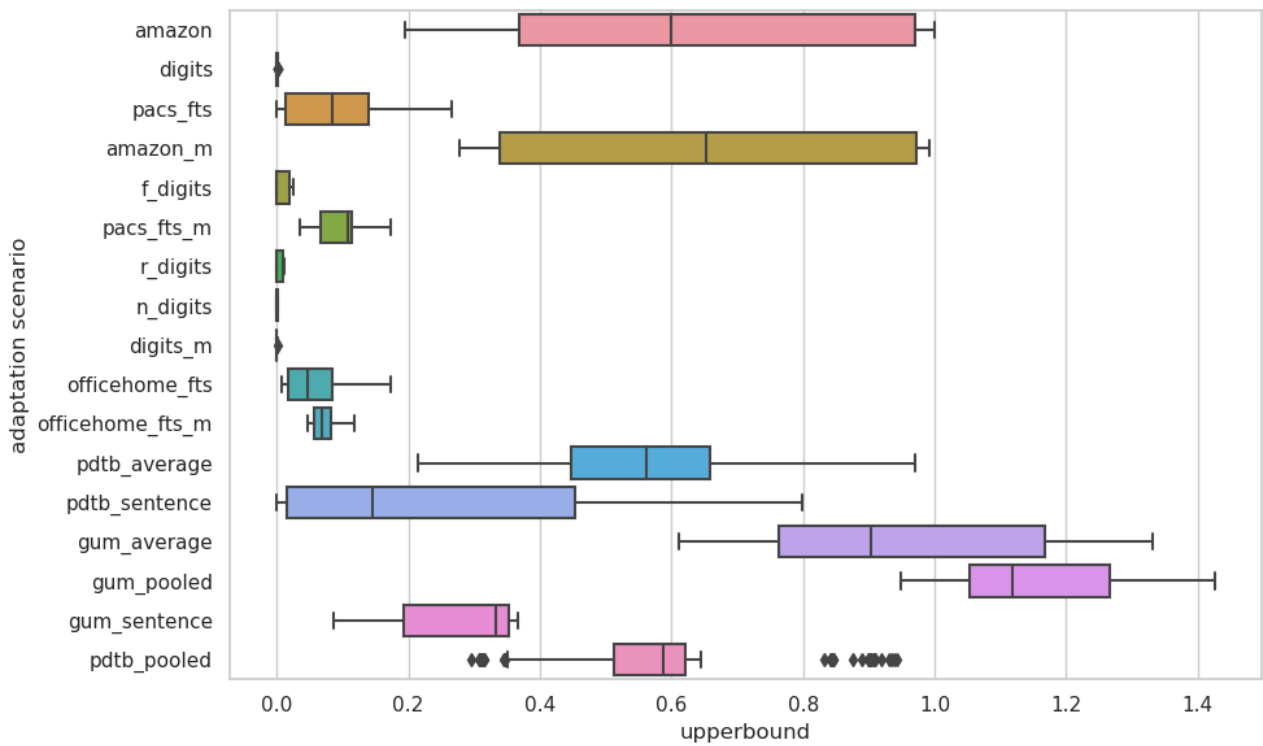


Figure 5: Boxplots for upperbounds on  $\tilde{\lambda}$  for individual datasets. The appendage `_m` denotes the **multi-source** setup, otherwise it is **single-source**. For digits, we prepend `r_`, `n_`, or `f_` to denote transfer to rotated, noisy, or randomly generated (fake) data as discussed 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.

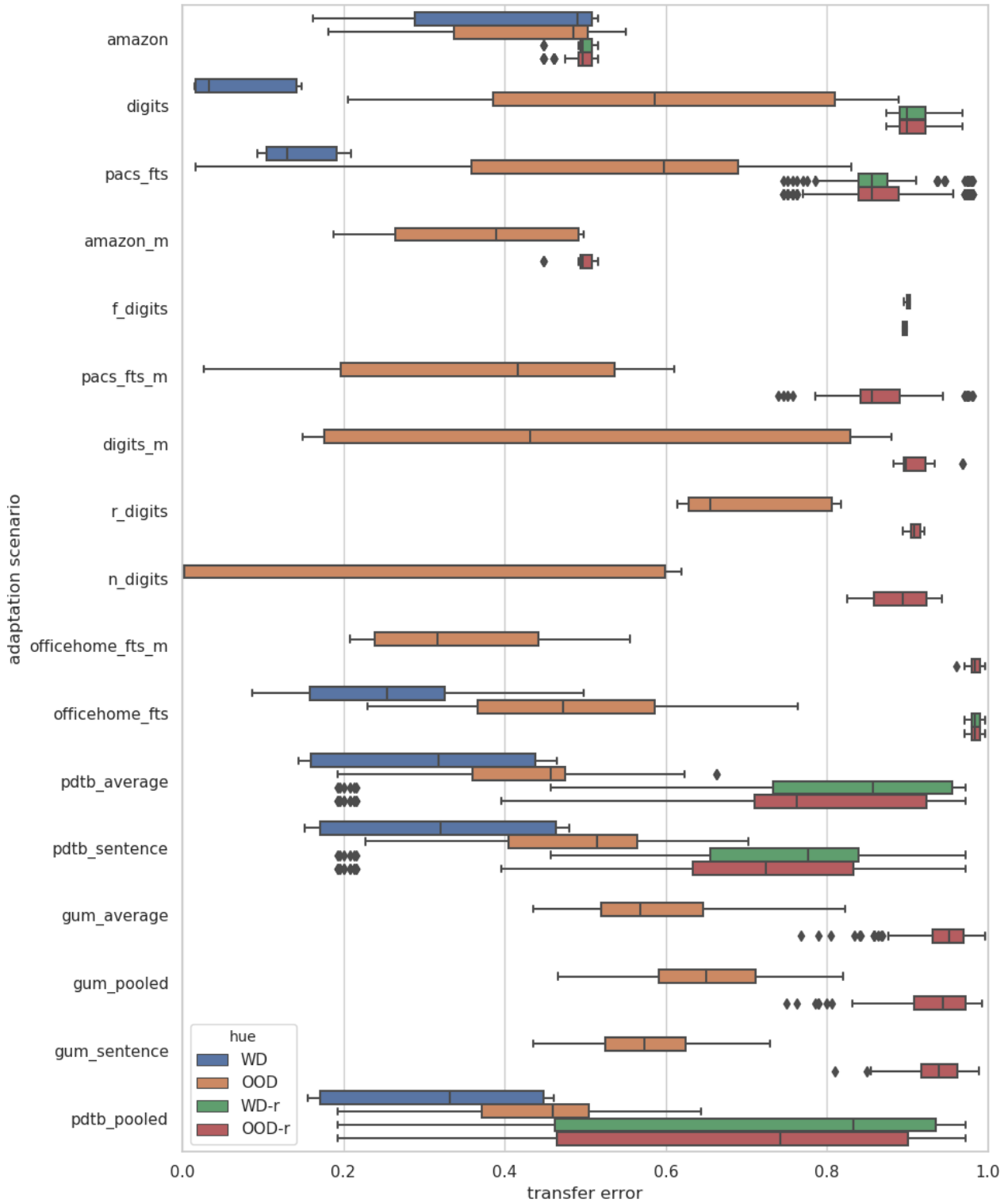


Figure 6: Dataset names are as in Figure 5. Hues correspond to within-distribution **WD** and out-of-distribution **OOD** based on whether the target  $T$  is drawn from the same component domain as the source  $S$ . Notice, **WD** is not available for multi-source setups and certain Digits setups. This 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.

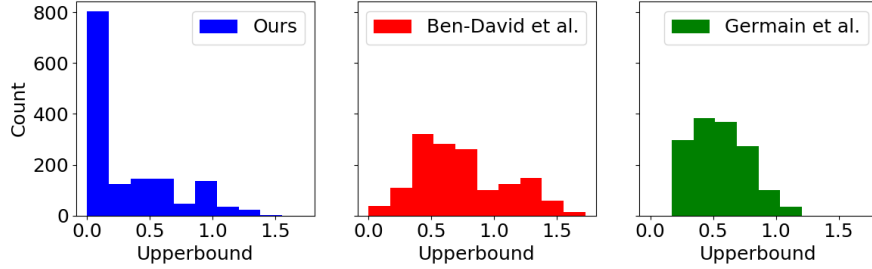


Figure 7: Upperbounds for sample-dependent (left), sample-independent (center), and binary PAC-Bayes variant (right) of adaptability. Each datum describes unique  $(S, T, \mathcal{H})$ .

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 7, we present a sample-dependent variation of the adaptability proposed by Germain et al.<sup>4</sup> 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 histogram in Figure 7 shows a histogram of upperbounds on a sample-dependent variation of the adaptability of Germain et al. [2020] given in Thm. A. 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  $(H_{i,1}, H_{i,2})_i \sim (\mathbb{Q} \times \mathbb{Q})^k$

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

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.

**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  $\Delta$  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 8, we show the effect of **DANN** on  $\rho$ . This confirms our takeaway in the main text that, as a function of sample complexity,  $\rho$  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 9, we show the target (transfer) error of solutions

<sup>4</sup>Some proof-techniques we employ in Thm. 2 can be applied to derive a sample-dependent variation of Thm. A, instead.

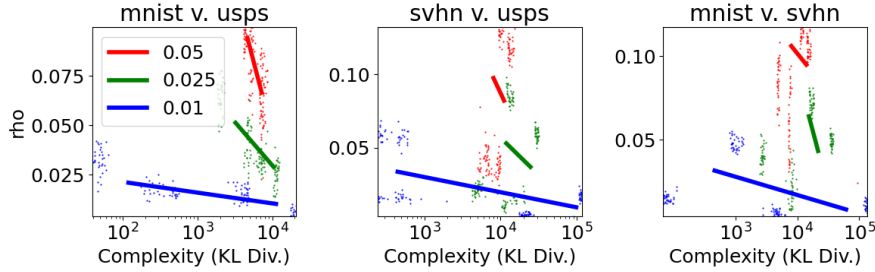


Figure 8: Estimates for  $\rho$  while using **DANN** for various choices of the prior variance parameter  $\sigma$ . Solid line shows median, while scatter shows 95% or more of data. Each datum describes unique  $(S, T, \mathbb{Q})$ . As a function of complexity, we expect  $\rho$  to be smaller for more complex solutions. For example, in the formula for KL-divergence between Gaussian distributions, similarly concentrated distributions will have high KL-divergence as their variances decrease (i.e., holding all else constant). Sensibly, smaller variance gives more concentrated  $\mathbb{Q}$ , which helps to ensure small  $\rho$  as well. This relationship (between  $\rho$  and complexity) is observed in the above and is similar to our findings in the main text on adaptability after **DANN**.

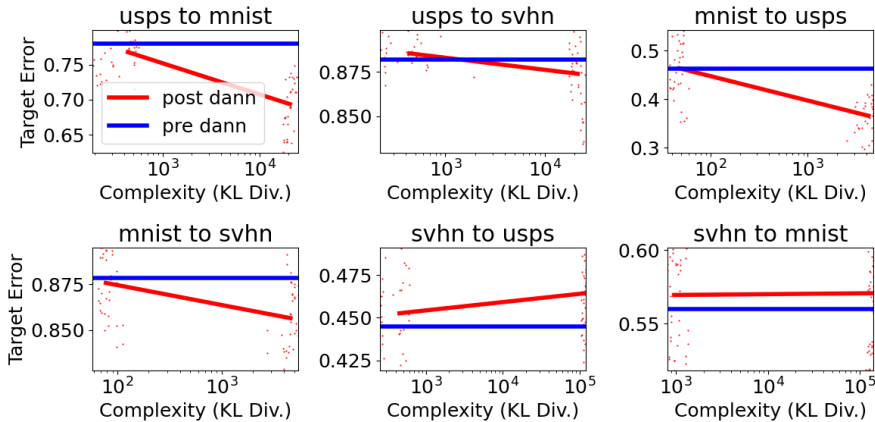


Figure 9: Estimates of target error  $\mathbf{R}_T(\mathbb{Q})$  (after **DANN**) compared to prior error  $\mathbf{R}_T(\mathbb{P})$  (before **DANN**, i.e. using **SA**). Solid line shows median, while scatter shows 95% or more of data. Each datum describes unique  $(S, T, \mathbb{Q})$ . More complex solutions achieve lower error as expected. **DANN** is effective at reducing target error in many cases.



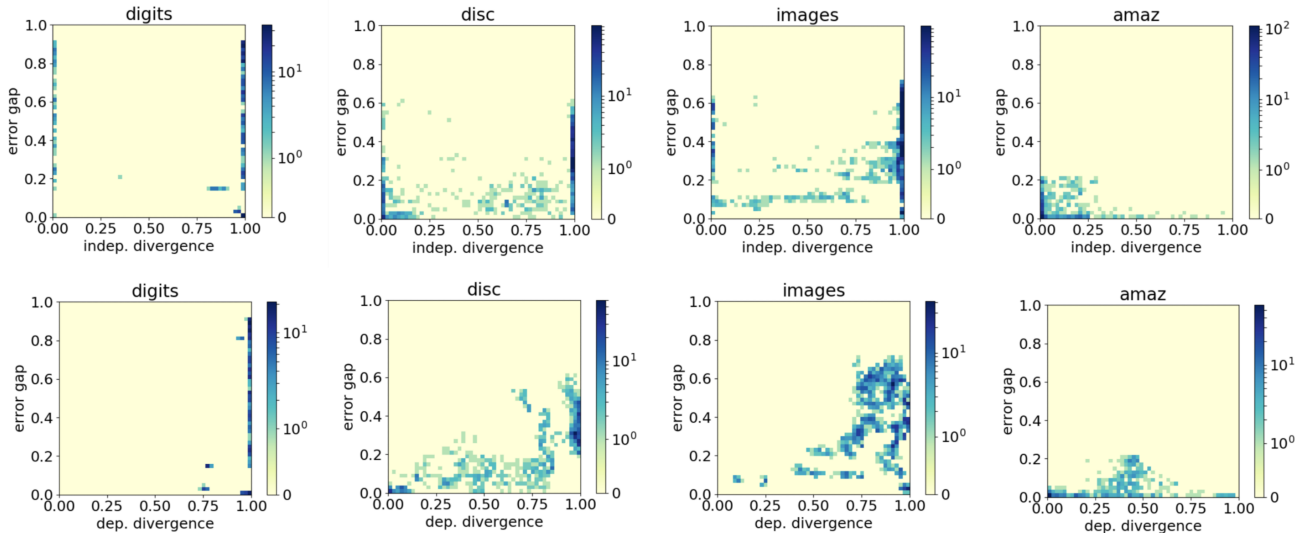


Figure 10: Heatmap (i.e., 2d histogram) showing counts for data used to compute correlations in Table 1. **images** corresponds to the **PACS+OH** dataset.

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.

## D.5 DETAILED VISUALIZATION OF DATA IN TABLE 1

In some instances, we observe poor correlation of the proposed divergence terms with the error-gap. For example, in Table 1, poor correlation is observed on the **Digits** dataset. Poor correlation is also observed on the **PACS+OH** dataset for the model-dependent divergence. To study and understand these errors in detail, we visualize heatmaps (i.e., 2d histograms) in Figure 10. Histogram counts illustrate counts of the individual data pairs used to compute correlation in Table 1; i.e., between a particular approximation of divergence and the corresponding error-gap. Please, see Section C.6 for details on the data used for Table 1.

Results show the poor performance on the **Digits** dataset is likely due to insensitivity of the divergence approximation to changes in data sample and hypothesis. In particular, there is significant concentration of the divergence approximations near 1. In the case of the model-independent divergence, we also observe some artificially low approximations (i.e., near 0) compared to the error-gap; this illustrates poor approximation. On the **PACS+OH** dataset, the poor performance of the model-dependent divergence is best explained by comparing to the data-points for the model-independent divergence. While the model-dependent divergence is more variable and sensitive to data/model changes as one would expect, we see a high density of anti-correlated measurements on the **PACS+OH** data. Specifically, there is a cluster of cases where the divergence is near 1 with absolute change in error about 0.4 and another cluster of cases where the divergence is only about 0.8 with absolute change in error higher at 0.6. Aply, the divergence does not perform well at ranking in this case.

These more nuanced results speak to the conservative nature of bounds (and their contained terms), in general. In particular, upperbounds are subject to “false positives” – in which the actual bound is high, but the quantity controlled is low; e.g., 1 is a valid bound on 1 and so are 2, 10, 50, and 1000. While this undesirable property impacts the **Digits** and **PACS+OH** cases, it is also worth mentioning that the divergences perform well in many other cases. Depending on the application, a conservative measure of performance change may actually be desirable.

## References

- Isabela Albuquerque, João Monteiro, Tiago H Falk, and Ioannis Mitliagkas. Adversarial target-invariant representation learning for domain generalization. *arXiv:1911.00804*, 2020.
- Amiran Ambroladze, Emilio Parrado-Hernández, and John Shawe-Taylor. Tighter pac-bayes bounds. *NeurIPS*, 19:9, 2007.
- 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, 2010a.
- Shai Ben-David, Tyler Lu, Teresa Luu, and Dávid Pál. Impossibility theorems for domain adaptation. In *AISTATS*, pages 129–136. JMLR Workshop and Conference Proceedings, 2010b.
- 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.
- Chloé Braud, Maximin Coavoux, and Anders Søgaard. Cross-lingual rst discourse parsing. *arXiv:1701.02946*, 2017.
- Lynn Carlson, Daniel Marcu, and Mary Ellen Okurowski. Building a discourse-tagged corpus in the framework of rhetorical structure theory. In *Current and new directions in discourse and dialogue*, pages 85–112. Springer, 2003.
- Olivier Catoni. PAC-Bayesian supervised classification: the thermodynamics of statistical learning. *arXiv:0712.0248v1*, 2007.
- Koby Crammer, Michael Kearns, and Jennifer Wortman. Learning from multiple sources. In *NeurIPS*, pages 321–328, 2007.
- 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, 2019.
- 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, 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, Alexandre Lacasse, François Laviolette, and Mario Marchand. PAC-Bayesian learning of linear classifiers. In *ICML*, 2009.
- Pascal Germain, Alexandre Lacasse, Francois Laviolette, Mario March, and Jean-Francois Roy. Risk Bounds for the Majority Vote: From a PAC-Bayesian Analysis to a Learning Algorithm. *JMLR*, 16:787–860, 2015.
- Pascal Germain, Amaury Habrard, François Laviolette, and Emilie Morvant. Pac-bayes and domain adaptation. *Neurocomputing*, 379:379–397, 2020.
- Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *JMLR*, 13(1):723–773, 2012.
- Benjamin Guedj. A primer on PAC-Bayesian learning. *arXiv:1901.05353v3*, 2019.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

- J. J. Hull. A database for handwritten text recognition research. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(5):550–554, 1994. doi: 10.1109/34.291440.
- Fredrik D Johansson, David Sontag, and Rajesh Ranganath. Support and invertibility in domain-invariant representations. In *AISTATS*, pages 527–536. PMLR, 2019.
- Yudai Kishimoto, Yugo Murawaki, and Sadao Kurohashi. Adapting BERT to implicit discourse relation classification with a focus on discourse connectives. In *LREC*, pages 1152–1158, Marseille, France, May 2020. European Language Resources Association. ISBN 979-10-95546-34-4. URL <https://aclanthology.org/2020.lrec-1.145>.
- 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 Matthias Seeger. Bounds for averaging classifiers. *CMU Technical Report*, 2001.
- Yann LeCun and Corinna Cortes. MNIST handwritten digit database. *web*, 2010. URL <http://yann.lecun.com/exdb/mnist/>.
- 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 classifier 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.
- 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.
- William C Mann and Sandra A Thompson. *Rhetorical structure theory: A theory of text organization*. University of Southern California, Information Sciences Institute Los Angeles, 1987.
- Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation: Learning bounds and algorithms. *arXiv:0902.3430*, 2009.
- Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. Building a large annotated corpus of english: The penn treebank. *Computational Linguistics*, 1993.
- Andreas Maurer. A note on the pac bayesian theorem. *arXiv cs/0411099*, 2004.
- David McAllester. A PAC-Bayesian tutorial with a dropout bound. *arXiv:1307.2118v1*, 2013.
- David A McAllester. Some PAC-Bayesian theorems. *Machine Learning*, 37:355–363, 1999.
- Daniel McNamara and Maria-Florina Balcan. Risk bounds for transferring representations with and without fine-tuning. In *ICML*, pages 2373–2381. PMLR, 2017.
- Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. *NeurIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*, 2011.
- Emilio Parrado-Hernández, Amiran Ambroladze, John Shawe-Taylor, and Shiliang Sun. PAC-Bayes bounds with data dependent priors. *JMLR*, 13:3507–3531, 2012.
- Maria 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.
- 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.
- Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv:1908.10084*, 2019.
- Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.
- John Shawe-Taylor and Robert C Williamson. A PAC analysis of a Bayesian estimator. In *COLT*, 1997.
- 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.
- 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.
- Amir Zeldes. The gum corpus: Creating multilayer resources in the classroom. *LREC*, 51(3):581–612, 2017.
- Deniz Zeyrek, Amália Mendes, Yulia Grishina, Murathan Kurfalt, 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.
- 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.
- Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Deep domain-adversarial image generation for domain generalisation. *arXiv:2003.06054*, 2020.