A Theory of Multiple-Source Adaptation with Limited Target Labeled Data

Yishay Mansour[1] **Mehryar Mohri**[1,2] **Jae Ro**[1] **Ananda Theertha Suresh**[1] **Ke Wu**[1]
[1]Google Research, [2]Courant Institute of Mathematical Sciences.

Abstract

We present a theoretical and algorithmic study of the multiple-source domain adaptation problem in the common scenario where the learner has access only to a limited amount of labeled target data, but where the learner has at their disposal a large amount of labeled data from multiple source domains. We show that a new family of algorithms based on model selection ideas benefits from very favorable guarantees in this scenario and discuss some theoretical obstacles affecting some alternative techniques. We also report the results of several experiments with our algorithms that demonstrate their practical effectiveness.

1 Introduction

A common assumption in supervised learning is that training and test distributions coincide. In practice, however, this assumption often does not hold. This is because the amount of labeled data available is too modest to train an accurate model. Instead, the learner must resort to using labeled samples from one or several alternative source domains or distributions that are expected to be close to the target domain. How can we leverage the labeled data from these source domains to come up with an accurate predictor for the target domain? This is the challenge of the *domain adaptation problem* that arises in a variety of different applications, such as in natural language processing (Blitzer et al., 2007; Dredze et al., 2007; Jiang and Zhai, 2007), speech processing (Gauvain and Lee, 1994; Jelinek, 1997), and computer vision (Leggetter and Woodland, 1995).

In practice, in addition to a relatively large number of total labeled data from source domains, the learner also has at their disposal a large amount of unlabeled data from the target domain, but only little or no labeled data from the target domain. Various scenarios of adaptation can be distinguished, depending on parameters including the number of source domains, the presence or absence of target labeled data, and access to labeled source data or only to predictors trained on each source domain.

The theoretical analysis of adaptation has been the subject of several publications in the last decade or so. The single-source adaptation problem was first studied by Ben-David et al. (2007), as well as follow-up publications (Blitzer et al., 2008; Ben-David et al., 2010), where the authors presented an analysis in terms of a d_A-distance, including VC-dimension learning bounds for the zero-one loss. Later, Mansour et al. (2009c) and Cortes and Mohri (2011, 2014) presented a general analysis of single-source adaptation for arbitrary loss functions, where they introduced the notion of *discrepancy*, which they argued is the suitable divergence measure in adaptation. The authors further gave Rademacher complexity learning bounds in terms of the discrepancy for arbitrary hypothesis sets and loss functions, as well as pointwise learning bounds for kernel-based hypothesis sets. The notion of discrepancy coincides with the d_A-distance in the special case of the zero-one loss.

Mansour et al. (2009a,b) and Hoffman et al. (2018); Zhang et al. (2020) considered the *multiple-source adaptation* (MSA) scenario where the learner has access to unlabeled samples and a trained predictor for each source domain, with no access to source labeled data. This approach has been further used in many applications such as object recognition (Hoffman et al., 2012; Gong et al., 2013a,b). Zhao et al. (2018) and Wen et al. (2020) considered MSA with only unlabeled target data available and provided generalization bounds for classification and regression.

There has been a very large recent literature dealing with experimental studies of domain adaptation in various tasks. Ganin et al. (2016) proposed to learn features that cannot discriminate between source and target domains. Tzeng et al. (2015) proposed a CNN architecture to exploit unlabeled and sparsely labeled

Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130.

target domain data. Motiian et al. (2017b), Motiian et al. (2017a) and Wang et al. (2019) proposed to train maximally separated features via adversarial learning. Saito et al. (2019) proposed to use a minmax entropy method for domain adaptation. We further discuss related previous work in Appendix A.

This paper presents a theoretical and algorithmic study of MSA with limited target labeled data, a scenario that is similar to the one examined by Konstantinov and Lampert (2019), who considered the problem of learning from multiple untrusted sources and a single target domain. We propose a new family of algorithms based on model selection ideas and show that they benefit from very favorable guarantees in this scenario and discuss some theoretical obstacles affecting some alternative techniques. We also report the results of several experiments with our algorithms that demonstrate their practical effectiveness.

In Section 2, we introduce some definitions and notation and formulate our learning problem. In Section 3, we provide some preliminary results. In Section 4, we present and analyze our algorithmic solutions (*Limited target data MSA* or LMSA algorithms) for the adaptation problem considered, which we prove benefit from near-optimal guarantees. In Section 5, we discuss some theoretical obstacles affecting some alternative techniques. Finally, in Section 6, we report the results of experiments with our LMSA algorithms and compare them with several other techniques and baselines.

2 Preliminaries

In this section, we introduce the definitions and notation used in our analysis and discuss a natural baseline and the formulation of the learning problem we study.

2.1 Definitions and notation

Let $\mathcal{X}$ denote the input space and $\mathcal{Y}$ the output space. We focus on the multi-class classification problem where $\mathcal{Y}$ is a finite set of classes, but much of our results can be extended straightforwardly to regression and other problems. The hypotheses we consider are of the form $h\colon \mathcal{X} \to \Delta_{\mathcal{Y}}$, where $\Delta_{\mathcal{Y}}$ stands for the simplex over $\mathcal{Y}$. Thus, $h(x)$ is a probability distribution over the classes or categories that can be assigned to $x \in \mathcal{X}$. We denote by $\mathcal{H}$ a family of such hypotheses. We denote by ℓ a loss function defined over $\Delta_{\mathcal{Y}} \times \mathcal{Y}$ and taking non-negative values with upper bound M. The loss of $h \in \mathcal{H}$ for a labeled sample $(x, y) \in \mathcal{X} \times \mathcal{Y}$ is given by $\ell(h(x), y)$. We denote by $\mathcal{L}_{\mathcal{D}}(h)$ the expected loss of a hypothesis h with respect to a distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$:

$$\mathcal{L}_{\mathcal{D}}(h) = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[\ell(h(x), y)],$$

and by $h_{\mathcal{D}}$ its minimizer: $h_{\mathcal{D}} = \operatorname{argmin}_{h\in\mathcal{H}} \mathcal{L}_{\mathcal{D}}(h)$.

We denote by $\mathcal{D}_0$ the target domain distribution and by $\mathcal{D}_1, \ldots, \mathcal{D}_p$ the p source domain distributions. During training, we observe m_k independent samples from distribution $\mathcal{D}_k$. We denote by $\widehat{\mathcal{D}}_k$ the corresponding empirical distribution. We also denote by $m = \sum_{k=1}^{p} m_k$ the total number of samples observed. In practice, we expect m to be significantly larger than m_0 ($m \gg m_0$).

It was shown by Mansour et al. (2009c) (see also Cortes and Mohri (2011)) that the *discrepancy* is the appropriate divergence between distributions in adaptation. The discrepancy takes into account the hypothesis set and the loss function, both key components of the structure of the learning problem. Furthermore, it has been shown that it can be estimated from finite samples and upper bounded in terms of other divergences, such as the total variation and the relative entropy. The discrepancy also coincides with the $d_{\mathcal{A}}$-distance proposed by Ben-David et al. (2007) in the special case of the zero-one loss.

A finer notion of discrepancy, which we will refer to as the *label-discrepancy*, was introduced by Mohri and Muñoz Medina (2012) (see also (Kuznetsov and Mohri, 2015)), which is useful in contexts where some target labeled data is available, as in our problem here. For two distributions $\mathcal{D}$ and $\mathcal{D}'$ over $\mathcal{X} \times \mathcal{Y}$ and a hypothesis set $\mathcal{H}$, the label-discrepancy is defined as follows:

$$\operatorname{disc}_{\mathcal{H}}(\mathcal{D}, \mathcal{D}') = \max_{h\in\mathcal{H}} |\mathcal{L}_{\mathcal{D}}(h) - \mathcal{L}_{\mathcal{D}'}(h)|.$$

This notion of discrepancy leads to tighter generalization bounds. When it is small, by definition, the expected loss of any hypothesis in $\mathcal{H}$ with respect to a source $\mathcal{D}$ is close to its expected loss with respect to $\mathcal{D}'$. In the rest of the paper, we use label-discrepancy and will refer to it simply by discrepancy.

2.2 Problem formulation

What is the best that one can achieve without data from any source distribution? Suppose we train on the target domain samples $\widehat{\mathcal{D}}_0$ alone, and obtain a model $h_{\widehat{\mathcal{D}}_0}$. By standard learning-theoretic tools (Mohri et al., 2018), the generalization bound for this model can be stated as follows: for simplicity let the loss be the zero-one loss. With probability at least $1 - \delta$, the minimizer of the empirical risk $\mathcal{L}_{\widehat{\mathcal{D}}_0}(h)$ satisfies,

$$\mathcal{L}_{\mathcal{D}_0}(h_{\widehat{\mathcal{D}}_0}) \leq \min_{h\in\mathcal{H}} \mathcal{L}_{\mathcal{D}_0}(h) + \mathcal{O}\left(\sqrt{\frac{d}{m_0}} + \sqrt{\frac{\log(1/\delta)}{m_0}}\right), \quad (1)$$

where d is the VC-dimension of the hypothesis class $\mathcal{H}$. For simplicity, we provided generalization bounds in terms of VC-dimension. They can be easily extended to bounds based on Rademacher complexity (Mohri et al., 2018) or pseudo-dimension (Pollard, 2012) for general losses. Finally, there exist distributions and hypotheses where (1) is tight (Mohri et al., 2018, Theorem 3.23).

Let Δ_p be the set of probability distributions over $[p]$. In order to provide meaningful bounds and improve upon (1), following (Mansour et al., 2009b; Hoffman et al., 2018), we assume that the target distribution is close to some convex combination of sources in the discrepancy measure, that is, we assume that there is a $\lambda \in \Delta_p$ such that $\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_\lambda)$ is small, where $\mathcal{D}_\lambda = \sum_{k=1}^p \lambda_k \mathcal{D}_k$.

With the above definitions, we can specify how good a mixture weight λ is. For a given λ, a natural algorithm is to combine samples from the empirical distributions $\widehat{\mathcal{D}}_k$ to obtain the mixed empirical distribution $\overline{\mathcal{D}}_\lambda = \sum_{k=1}^p \lambda_k \widehat{\mathcal{D}}_k$, and minimize loss on $\overline{\mathcal{D}}_\lambda$. Let $h_{\overline{\mathcal{D}}_\lambda}$ be the minimizer of this loss. A good λ should lead to $h_{\overline{\mathcal{D}}_\lambda}$ with the performance close to that of the optimal estimator for $\mathcal{D}_0$. In other words, the goal is to find λ that minimizes

$$\mathcal{L}_{\mathcal{D}_0}(h_{\overline{\mathcal{D}}_\lambda}) - \mathcal{L}_{\mathcal{D}_0}(h_{\mathcal{D}_0}).$$

The above term can be bounded by a uniform excess risk bound as follows:

$$\begin{aligned}&\mathcal{L}_{\mathcal{D}_0}(h_{\overline{\mathcal{D}}_\lambda}) - \mathcal{L}_{\mathcal{D}_0}(h_{\mathcal{D}_0}) \\ &\le 2 \max_{h \in \mathcal{H}} |\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h) - \mathcal{L}_{\mathcal{D}_\lambda}(h)| + 2\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_\lambda). \quad (2)\end{aligned}$$

The derivation of (2) is given in Appendix B. Let the uniform bound on the excess risk for a given λ be

$$\mathcal{E}(\lambda) = 2 \max_{h \in \mathcal{H}} |\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h) - \mathcal{L}_{\mathcal{D}_\lambda}(h)| + 2\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_\lambda), \quad (3)$$

and λ^* be the mixture weight that minimizes the above uniform excess bound, i.e.

$$\lambda^* = \operatorname*{argmin}_{\lambda \in \Delta_p} \mathcal{E}(\lambda).$$

Our goal is to come up with a model with error close to $\mathcal{E}(\lambda^*)$, without the knowledge of λ^*. If the target domain $\mathcal{D}_0$ is not exactly a convex combination of the sources, then it is captured by the discrepancy term $\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_{\lambda^*})$ in the definition of $\mathcal{E}(\lambda^*)$. Our results degrade smoothly as $\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_{\lambda^*})$ increases. Furthermore, our results depend on $\mathcal{E}(\lambda^*)$, which is hypothesis-independent and an upper bound on the uniform excess risk bound. Replacing it with a hypothesis-dependent upper bound is an interesting future direction. Before we review the existing algorithms, we provide a bound on $\mathcal{E}(\lambda^*)$.

3 Fixed target mixture

The adaptation problem we are considering can be broken down into two parts: (i) finding the minimizing mixture weight λ^*; (ii) determining the hypothesis that minimizes the loss over corresponding distribution $\widehat{\mathcal{D}}_{\lambda^*}$. In this section, we discuss guarantees for (ii), for a known mixture weight λ^*. This will later serve as a reference for our analysis in the more general case. More generally, we consider here guarantees for a fixed mixture weight λ.

Let $\mathbf{m}$ denote the empirical distribution of samples $(m_1/m, m_2/m, \ldots, m_p/m)$. Skewness between distributions is defined as $\mathfrak{s}(\lambda \| \mathbf{m}) = \sum_{k=1}^p \frac{\lambda_k^2}{\mathbf{m}_k}$. Skewness is a divergence and measures how far λ and the empirical distribution of samples $\mathbf{m}$ are. It naturally arises in the generalization bounds of weighted mixtures. For example, if $\lambda = \mathbf{m}$, then $\frac{\mathfrak{s}(\lambda \| \mathbf{m})}{m} = \frac{1}{m}$ and the generalization bound in Proposition 1 will be the same as the bound for the uniform weighted model. If $\lambda = (1, 0, \ldots, 0)$, then $\frac{\mathfrak{s}(\lambda \| \mathbf{m})}{m} = \frac{1}{m_1}$ and the generalization bound will be the same as the bound for training on a single domain. Thus, skewness smoothly interpolates between the uniform weighted model and the single domain model. For a fixed λ, the following generalization bound of Mohri et al. (2019) holds (see also Blitzer et al. (2008) in the special case of the zero-one loss).

Proposition 1. *Let $\lambda \in \Delta_p$. Then with probability at least $1 - \delta$, $\mathcal{E}(\lambda)$ is bounded by*

$$4M \sqrt{\frac{\mathfrak{s}(\lambda \| \mathbf{m})}{m} \cdot \left(d \log \frac{em}{d} + \log \frac{1}{\delta} \right)} + 2 \mathit{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_\lambda).$$

Since $m \gg m_0$, this guarantee is substantially stronger than the bound given for a model trained on the target data only (1).

4 Unknown target mixture

Here, we analyze the more realistic scenario where no information about the target mixture weight is assumed. Our objective is to come up with a hypothesis whose excess risk guarantee is close to the one shown in the known target mixture setting.

One natural idea to tackle this problem consists of first determining the mixture weight λ for which $\mathcal{D}_\lambda$ is the closest to $\mathcal{D}_0$ for some divergence measure such as a Bregman divergence B:

$$\min_{\lambda \in \Delta_p} \mathsf{B}(\widehat{\mathcal{D}}_0 \| \overline{\mathcal{D}}_\lambda),$$

But, as discussed in Appendix C.1, this approach is subject to several issues resulting in poor theoretical guarantees. An alternative consists of seeking λ to minimize the following objective function:

$$\mathcal{L}_{\mathcal{D}_\lambda}(h) + \mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_\lambda).$$

However, this requires estimating both the expected loss and the discrepancy terms and, as discussed in Appendix C.2, in general, the guarantees for this technique are comparable to those of the straightforward baseline of training on $\widehat{\mathcal{D}}_0$.

Instead, we will describe a family of algorithms based on a natural model selection idea, which we show benefits from strong theoretical guarantees. Unlike the straightforward baseline algorithm or other techniques just discussed, the dominating term of the learning bounds for our algorithms are in $\widetilde{O}(\sqrt{p/m_0})$, that is the square-root of the ratio of the number of sources and the number of target labeled samples and do not depend on the complexity of the hypothesis set. This is in contrast, for example, to the $O(\sqrt{d/m_0})$ bound for the straightforward baseline, where d is the VC-dimension.

We will show that the hypothesis $h_{\mathcal{A}}$ returned by our algorithm verifies the following inequality:

$$\mathcal{L}_{\mathcal{D}_0}(h_{\mathcal{A}}) \leq \min_{h\in\mathcal{H}} \mathcal{L}_{\mathcal{D}_0}(h) + \mathcal{E}(\lambda^*) + \tilde{\mathcal{O}}\left(\sqrt{\frac{p}{m_0}}\right).$$

We further show that the above additional penalty of $\tilde{\mathcal{O}}\left(\sqrt{\frac{p}{m_0}}\right)$ is necessary, by showing an information-theoretic lower bound. We show that for any algorithm $\mathcal{A}$, there exists a hypothesis class $\mathcal{H}$ and domains such that $\mathcal{E}(\lambda^*) = 0$ and

$$\mathcal{L}_{\mathcal{D}_0}(h_{\mathcal{A}}) \geq \min_{h\in\mathcal{H}} \mathcal{L}_{\mathcal{D}_0}(h) + \Omega\left(\sqrt{\frac{p}{m_0}}\right).$$

These results characterize the penalty term for MSA with limited target data up to *log* factors. We now present our algorithms for the *limited target data MSA* problems: LMSA, LMSA-Boost, and LMSA-Minmax, as well as an information-theoretic lower bound.

4.1 LMSA algorithm

Since $\mathcal{D}_0 \approx \sum_k \lambda_k^* \mathcal{D}_k$, one approach inspired by model selection consists of determining the hypothesis with the minimal loss for each value of λ and selecting among them the hypothesis that performs best on $\mathcal{D}_0$. We call this general algorithm (LMSA) (see Figure 1).

The algorithm takes as an input a subset Λ of Δ_p, which can be chosen to be a finite cover of Δ_p. For each element of Λ, it finds the best estimator for $\overline{\mathcal{D}}_\lambda$, denoted by $h_{\overline{\mathcal{D}}_\lambda}$. Let $\mathcal{H}_\Lambda$ be the resulting set of hypotheses. The algorithm then selects the best hypothesis out this set, by using $\widehat{\mathcal{D}}_0$. The algorithm is relatively parameter-free and straightforward to implement.

We now show that LMSA benefits from the following favorable guarantee, when Λ is a finite cover of Δ_p.

Theorem 1. *Let $\epsilon \leq 1$. Let Λ be a minimal cover of Δ_p such that for each $\lambda \in \Delta_p$, there exists a $\lambda_\epsilon \in \Lambda$ such that $\|\lambda - \lambda_\epsilon\|_1 \leq \epsilon$. Then, for any $\delta > 0$, with probability at least $1-\delta$, the hypothesis h_m returned by* LMSA(Λ) *satisfies the following inequality:*

$$\mathcal{L}_{\mathcal{D}_0}(h_m) - \min_{h\in\mathcal{H}} \mathcal{L}_{\mathcal{D}_0}(h) \leq \mathcal{E}(\lambda^*) + 2\epsilon M + \frac{2M\sqrt{p\log\frac{p}{\delta\epsilon}}}{\sqrt{m_0}}.$$

1. For any $\lambda \in \Lambda$, compute $h_{\overline{\mathcal{D}}_\lambda}$ defined by
$$h_{\overline{\mathcal{D}}_\lambda} = \operatorname*{argmin}_{h\in\mathcal{H}} \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h).$$
2. Define $\mathcal{H}_\Lambda = \{h_{\overline{\mathcal{D}}_\lambda} : \forall \lambda \in \Lambda\}$.
3. Return h_m defined by
$$h_m = \operatorname*{argmin}_{h\in\mathcal{H}_\Lambda} \mathcal{L}_{\widehat{\mathcal{D}}_0}(h).$$

Figure 1: Algorithm LMSA(Λ).

Proof. Since h_m is the minimizer of $\mathcal{L}_{\widehat{\mathcal{D}}_0}(h)$,

$$\mathcal{L}_{\mathcal{D}_0}(h_m) - \min_{h\in\mathcal{H}_\Lambda} \mathcal{L}_{\mathcal{D}_0}(h) \leq 2\max_{h\in\mathcal{H}_\Lambda} |\mathcal{L}_{\widehat{\mathcal{D}}_0}(h) - \mathcal{L}_{\mathcal{D}_0}(h)|. \tag{4}$$

We now bound the number of elements in the cover Λ. Consider the cover Λ given as follows. For each coordinate $k < p$, the domain weight λ_k belongs to the set $\{0, \epsilon/p, 2\epsilon/p, \ldots, 1\}$, and $(\lambda_\epsilon)_p$ is determined by the fact that $\sum_k (\lambda_\epsilon)_k = 1$. The cover has at most $(p/\epsilon)^{p-1}$ elements and for every λ, there is a λ_ϵ such that $\|\lambda - \lambda_\epsilon\|_1 \leq \epsilon$. Hence, the size of the minimal cover is at most $(p/\epsilon)^{p-1}$. Thus, by McDiarmid's inequality and the union bound, with probability at least $1-\delta$, the following holds:

$$\max_{h\in\mathcal{H}_\Lambda} |\mathcal{L}_{\widehat{\mathcal{D}}_0}(h) - \mathcal{L}_{\mathcal{D}_0}(h)| \leq \frac{M\sqrt{p\log\frac{p}{\delta\epsilon}}}{\sqrt{m_0}}. \tag{5}$$

Let h_λ denote $h_{\mathcal{D}_\lambda}$ and $h_{\hat{\lambda}}$ denote $h_{\overline{\mathcal{D}}_\lambda}$. For any λ,

$$\begin{aligned}
&\min_{h\in\mathcal{H}_\Lambda} \mathcal{L}_{\mathcal{D}_0}(h) - \min_{h\in\mathcal{H}} \mathcal{L}_{\mathcal{D}_0}(h)\\
&\overset{(a)}{\leq} \min_{h\in\mathcal{H}_\Lambda} \mathcal{L}_{\mathcal{D}_\lambda}(h) - \min_{h\in\mathcal{H}} \mathcal{L}_{\mathcal{D}_\lambda}(h) + 2\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_\lambda)\\
&\leq \mathcal{L}_{\mathcal{D}_\lambda}(h_{\hat{\lambda}_\epsilon}) - \mathcal{L}_{\mathcal{D}_\lambda}(h_\lambda) + 2\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_\lambda)\\
&\leq \mathcal{L}_{\mathcal{D}_\lambda}(h_{\hat{\lambda}_\epsilon}) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_{\hat{\lambda}_\epsilon}) + \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_{\hat{\lambda}_\epsilon}) - \mathcal{L}_{\mathcal{D}_\lambda}(h_\lambda)\\
&\quad + 2\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_\lambda)\\
&\overset{(b)}{\leq} \mathcal{L}_{\mathcal{D}_\lambda}(h_{\hat{\lambda}_\epsilon}) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_{\hat{\lambda}_\epsilon}) + \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_\lambda) - \mathcal{L}_{\mathcal{D}_\lambda}(h_\lambda)\\
&\quad + 2\epsilon M + 2\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_\lambda)\\
&\overset{(c)}{\leq} \mathcal{E}(\lambda) + 2\epsilon M,
\end{aligned} \tag{6}$$

(a) follows from the definition of discrepancy and (c) follows from the definition of $\mathcal{E}(h)$. For (b), observe that by the definition of h_λ and h_{λ_ϵ},

$$\begin{aligned}
\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_{\hat{\lambda}_\epsilon}) &\leq \mathcal{L}_{\overline{\mathcal{D}}_{\lambda_\epsilon}}(h_{\hat{\lambda}_\epsilon}) + \epsilon M \leq \mathcal{L}_{\overline{\mathcal{D}}_{\lambda_\epsilon}}(h_{\hat{\lambda}}) + \epsilon M\\
&\leq \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_{\hat{\lambda}}) + 2\epsilon M \leq \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_\lambda) + 2\epsilon M,
\end{aligned}$$

where the second inequality follows by observing that $h_{\hat{\lambda}_\epsilon}$ is the optimal estimator for $\mathcal{L}_{\overline{\mathcal{D}}_{\lambda_\epsilon}}$. The last inequality follows similarly. Combining equations (4),

(5), and (6) and taking the minimum over λ yields the theorem. □

Note that the guarantee for LMSA is closer to the setting of a known mixture setting λ^*. The algorithm finds a mixture weight λ^* that not only admits a small discrepancy with respect to the distribution $\mathcal{D}_0$, but also has a small skewness and thus generalizes better. In particular, if there are multiple distributions that are very close to $\mathcal{D}_0$, then it chooses the one that generalizes better. Furthermore, if there is a λ^* such that $\mathcal{D}_0 = \sum_{k=1}^p \lambda_k^* \mathcal{D}_k$, then the algorithm chooses either λ^* or another λ that is slightly worse in terms of discrepancy, but generalizes substantially better.

Finally, the last term in Theorem 1, $\frac{2M\sqrt{p\log\frac{p}{\delta\epsilon}}}{\sqrt{m_0}}$, is the penalty for model selection and only depends on the number of samples from $\mathcal{D}_0$ and is independent of λ. Note that for the guarantee of this algorithm to be favorable than that of the local model (1), we need $p < d$. This, however, is a fairly reasonable assumption in practice since the number of domains in applications is in the order of several hundreds, while the typical number of model parameters can be significantly more than several millions. Furthermore, by combining the cover-based bound (5) with VC-dimension bounds, one can reduce the penalty of model selection to the following: $\mathcal{O}\left(\min\left(\frac{M\sqrt{p\log\frac{p}{\delta\epsilon}}}{\sqrt{m_0}}, \sqrt{\frac{d}{m_0}}\right)\right)$. Let T denote the time complexity of finding $h_{\overline{\mathcal{D}}_\lambda}$ for a given λ. Then, the overall time complexity of LMSA is $\left(\frac{p}{\epsilon}\right)^{p-1} T$. Thus, the algorithm is efficient for small values of p.

4.2 LMSA-Boost algorithm

In this section, we seek a more efficient boosting-type solution to the MSA problem that we call LMSA-Boost. This consists of considering the family of base predictors $\{h_\lambda : \lambda \in \Lambda\}$ and searching for an optimal ensemble, where h_λ is the best hypothesis for the λ-weighted mixture of source domains. The problem is therefore the following convex optimization in terms of the mixture weights λ:

$$\min_{\alpha} \mathcal{L}_{\mathcal{D}_0}\Big(\sum_{\lambda\in\Lambda} \alpha_\lambda h_\lambda\Big), \tag{7}$$

subject to $\sum_{\lambda\in\Lambda} \alpha_\lambda = 1$ and $\alpha_\lambda \geq 0$ for all λ.

We first show that the solution of this optimization problem benefits from a generalization guarantee similar to that of LMSA(Λ).

Proposition 2. *Let $\epsilon \leq 1$ and ℓ be L Lipschitz. Let Λ be the ϵ-cover defined in Theorem 1. Then, for any $\delta > 0$, with probability at least $1-\delta$, the solution of (7) h_m satisfies the following inequality:*

$$\mathcal{L}_{\mathcal{D}_0}(h_m) - \min_{h\in\mathcal{H}} \mathcal{L}_{\mathcal{D}_0}(h) \leq \mathcal{E}(\lambda^*) + 2\epsilon M + 2L\sqrt{\frac{2p\log\frac{p}{\epsilon}}{m_0}} + 2M\sqrt{\frac{\log\frac{1}{\delta}}{m_0}}.$$

Proof. Let $\mathrm{conv}(\mathcal{H}_\Lambda)$ denote the convex hull of $\mathcal{H}_\Lambda$. Then,

$$\begin{aligned}
&\mathcal{L}_{\mathcal{D}_0}(h_m) - \min_{h\in\mathcal{H}} \mathcal{L}_{\mathcal{D}_0}(h)\\
&= \mathcal{L}_{\mathcal{D}_0}(h_m) - \min_{h\in\mathrm{conv}(\mathcal{H}_\Lambda)} \mathcal{L}_{\mathcal{D}_0}(h)\\
&+ \min_{h\in\mathrm{conv}(\mathcal{H}_\Lambda)} \mathcal{L}_{\mathcal{D}_0}(h) - \min_{h\in\mathcal{H}} \mathcal{L}_{\mathcal{D}_0}(h).
\end{aligned}$$

By (6), for any λ,

$$\begin{aligned}
&\min_{h\in\mathrm{conv}(\mathcal{H}_\Lambda)} \mathcal{L}_{\mathcal{D}_0}(h) - \min_{h\in\mathcal{H}} \mathcal{L}_{\mathcal{D}_0}(h)\\
&\leq \min_{h\in\mathcal{H}_\Lambda} \mathcal{L}_{\mathcal{D}_0}(h) - \min_{h\in\mathcal{H}} \mathcal{L}_{\mathcal{D}_0}(h) \leq \mathcal{E}(\lambda) + 2\epsilon M.
\end{aligned}$$

We now show that

$$\begin{aligned}
&\mathcal{L}_{\mathcal{D}_0}(h_m) - \min_{h\in\mathrm{conv}(\mathcal{H}_\Lambda)} \mathcal{L}_{\mathcal{D}_0}(h)\\
&\leq 2L\sqrt{\frac{2p\log\frac{p}{\epsilon}}{m_0}} + 2M\sqrt{\frac{\log\frac{1}{\delta}}{m_0}}.
\end{aligned} \tag{8}$$

Combining the above three equations yields the result. To prove (8), observe that since $\mathcal{L}_{\mathcal{D}_0}\Big(\sum_{\lambda\in\Lambda} \alpha_\lambda h_\lambda\Big)$ is convex, h_m is the minimizer of $\mathcal{L}_{\widehat{\mathcal{D}}_0}(h)$. Hence,

$$\begin{aligned}
&\mathcal{L}_{\mathcal{D}_0}(h_m) - \min_{h\in\mathrm{conv}(\mathcal{H}_\Lambda)} \mathcal{L}_{\mathcal{D}_0}(h)\\
&\qquad\leq 2\max_{h\in\mathrm{conv}(\mathcal{H}_\Lambda)} |\mathcal{L}_{\mathcal{D}_0}(h) - \mathcal{L}_{\widehat{\mathcal{D}}_0}(h)|.
\end{aligned}$$

By McDiarmid's inequality, with probability at least $1-\delta$,

$$\begin{aligned}
&\max_{h\in\mathrm{conv}(\mathcal{H}_\Lambda)} |\mathcal{L}_{\mathcal{D}_0}(h) - \mathcal{L}_{\widehat{\mathcal{D}}_0}(h)|\\
&\leq \mathbb{E}\max_{h\in\mathrm{conv}(\mathcal{H}_\Lambda)} |\mathcal{L}_{\mathcal{D}_0}(h) - \mathcal{L}_{\widehat{\mathcal{D}}_0}(h)| + M\sqrt{\frac{\log\frac{1}{\delta}}{m_0}},
\end{aligned}$$

By the definition of the Rademacher complexity,

$$\mathbb{E}\Big[\max_{h\in\mathrm{conv}(\mathcal{H}_\Lambda)} |\mathcal{L}_{\mathcal{D}_0}(h) - \mathcal{L}_{\widehat{\mathcal{D}}_0}(h)|\Big] \leq \mathfrak{R}_{m_0}(\mathrm{conv}(\ell(\mathcal{H}_\Lambda))).$$

Since the Rademacher complexity of a convex hull coincides with that of the class,

$$\begin{aligned}
\mathfrak{R}_{m_0}(\ell(\mathrm{conv}(\mathcal{H}_\Lambda))) &\leq L\mathfrak{R}_{m_0}(\mathrm{conv}(\mathcal{H}_\Lambda))\\
&= L\mathfrak{R}_{m_0}(\mathcal{H}_\Lambda) \leq L\sqrt{\frac{2p\log\frac{p}{\epsilon}}{m_0}}.
\end{aligned}$$

This completes the proof. □

Since the loss function is convex, (7) is convex in α_λ. However, the number of predictors is $(p/\epsilon)^{p-1}$, which can be potentially large. This scenario is very similar to that of boosting where the number of base predictors such as decision trees can be very large and where the goal is to find a convex combination that performs well. To tackle this problem, we can use randomized or block-randomized coordinate decent (RCD) (Nesterov, 2012). The convergence guarantees follow from known results on RCD (Nesterov, 2012).

Motivated by this, the algorithm proceeds as follows. Let λ^t be the coordinate chosen at time t and α_{λ^t}, h_{λ^t} be the corresponding mixture weight and the hypothesis at time t. We propose to find $\alpha_{\lambda^{t+1}}$ and λ^{t+1} as follows. The algorithm randomly selects s values of λ, denoted by S^{t+1} and chooses the one that minimizes

$$(\alpha_{\lambda^{t+1}}, \lambda^{t+1}) = \operatorname*{argmin}_{\alpha, \lambda \in S^{t+1}} \mathcal{L}_{\widehat{\mathcal{D}}_0}\left(\sum_{i=1}^{t} \alpha_{\lambda^i} h_{\lambda^i} + \alpha h_\lambda\right).$$

Furthermore, at each step, we renormalize α so that it sums to one. We refer to this algorithm as LMSA-Boost. It is known that the above algorithm converges to the global optimum (Nesterov, 2012).

In practice, for efficiency purposes, we can use different sampling schemes. Suppose, for example, that we have a hierarchical clustering of Λ. At each round, instead of randomly sampling a set S^t with s values of λ, we could sample s values of λ, one from each cluster and find the λ with the maximum decrease in loss. We can then sample s values of λ, one from each sub-cluster of the chosen cluster. This process is repeated till the reduction in loss is small, at which point we can choose the corresponding λ as λ^{t+1}. This algorithm is similar to heuristics used for boosting with decision trees.

4.3 LMSA-Min-max algorithm

Theorem 1 shows algorithm LMSA(Λ) benefits from favorable guarantees for finite covers. Here, we seek a gradient-descent type solution that mimics LMSA and is computationally efficient. To that end, we extend this result to the entire simplex Δ_p. To extend the result to the entire simplex, we need a continuity argument, which states that if λ and λ' are close, then the optimal hypothesis for convex combination λ and the optimal hypothesis for combination λ' are close. This does hold, in general, for non-convex loss functions. Hence, we make an additional assumption that the loss function ℓ is strongly convex in the parameters of optimization. The generalization bound uses the following lemma proven in Appendix D.

Lemma 1. *Let $h_\lambda = \operatorname{argmin} \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h)$, and ℓ be a μ-strongly convex function whose gradient norms are bounded, $\|\nabla \ell(h(x), y))\| \leq G$ for all x, y. Then for any distribution $\mathcal{D}_0$,*

$$\mathcal{L}_{\mathcal{D}_0}(h_\lambda) - \mathcal{L}_{\mathcal{D}_0}(h_{\lambda'}) \leq \frac{G\sqrt{M}}{\sqrt{\mu}} \cdot \|\lambda - \lambda'\|_1^{1/2}.$$

The following lemma provides a generalization guarantee for LMSA(Δ_p).

Lemma 2. *Under the assumptions of Lemma 1, for any $\delta > 0$, with probability at least $1 - \delta$, the hypothesis h_m returned by* LMSA(Δ_p) *satisfies the following inequality:*

$$\mathcal{L}_{\mathcal{D}_0}(h_m) - \min_{h \in \mathcal{H}} \mathcal{L}_{\mathcal{D}_0}(h) \leq \mathcal{E}(\lambda^*) + \min_{\epsilon \geq 0} \frac{2M\sqrt{p \log \frac{pG^2M}{\epsilon^2 \mu \delta}}}{\sqrt{m_0}} + 2\epsilon M.$$

Proof. The proof is similar to that of Theorem 1, thus we only provide a sketch. Let Λ be the minimal cover of Δ_p in the ℓ_1 distance such that any two elements of the cover have distance at most $\frac{\mu\epsilon^2}{G^2 M}$. Such a cover will have at most $\left(\frac{pG^2M}{\mu\epsilon^2}\right)^p$ elements. Hence, with probability at least $1 - \delta$,

$$\begin{aligned}
&\mathcal{L}_{\mathcal{D}_0}(h_m) - \min_{h \in \mathcal{H}_{\Delta_p}} \mathcal{L}_{\mathcal{D}_0}(h) \\
&\overset{(a)}{\leq} 2 \max_{h \in \mathcal{H}_{\Delta_p}} |\mathcal{L}_{\mathcal{D}_0}(h) - \mathcal{L}_{\widehat{\mathcal{D}}_0}(h)| \\
&\overset{(b)}{\leq} 2 \max_{h \in \mathcal{H}_\Lambda} |\mathcal{L}_{\mathcal{D}_0}(h) - \mathcal{L}_{\widehat{\mathcal{D}}_0}(h)| + 2\epsilon M \\
&\overset{(c)}{\leq} \frac{2M\sqrt{p \log \frac{pG^2M}{2\epsilon^2 \mu \delta}}}{\sqrt{m_0}} + 2\epsilon M, \qquad (9)
\end{aligned}$$

where (a) follows from (4), (b) follows from Lemma 1 and the properties of the cover, and (c) follows by the McDiarmid's inequality together with a union bound over the above cover. By (6),

$$\min_{h \in \mathcal{H}_{\Delta_p}} \mathcal{L}_{\mathcal{D}_0}(h) - \min_{h \in \mathcal{H}} \mathcal{L}_{\mathcal{D}_0}(h) \leq \mathcal{E}(\lambda). \qquad (10)$$

Combining (9) and (10) and taking the minimum over λ completes the proof. □

In view of these results, we propose a gradient descent based algorithm LMSA-Min-max for solving the LMSA objective. The following is the corresponding optimization problem:

$$\min_{h \in \mathcal{H}, \lambda \in \Delta_p} \max_{\gamma \geq 0, h' \in \mathcal{H}} \mathcal{L}_{\widehat{\mathcal{D}}_0}(h) + \gamma\left(\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h').\right) \qquad (11)$$

The above algorithm can be viewed as a two-player game, where the first player controls the hypothesis h and the weights λ and the second player controls the

Lagrange multiplier γ and the alternate hypothesis h'. Here, the goal of the first player is to find the best hypothesis that minimizes the best fitting model, while the second player acts as a *certifier* who determines if the model selected by the first player belongs to $\mathcal{H}_{\Delta_p}$. We show that (11) returns the same solution as $\mathrm{LMSA}(\Delta_p)$ for strictly convex functions.

Theorem 2. *Assume that ℓ is strictly convex. Then, the minimizer of (11) coincides with the output of $\mathrm{LMSA}(\Delta_p)$.*

Proof. If the function ℓ is strictly convex in h,

$$\begin{aligned}
&\min_{h\in\mathcal{H}_{\Delta_p}} \mathcal{L}_{\widehat{\mathcal{D}}_0}(h)\\
&= \min_{h\in\mathcal{H}} \mathcal{L}_{\widehat{\mathcal{D}}_0}(h) + \max_{\gamma\geq 0} \gamma 1_{h\notin\mathcal{H}_{\Delta_p}}\\
&\overset{(a)}{=} \min_{h\in\mathcal{H}} \mathcal{L}_{\widehat{\mathcal{D}}_0}(h) + \max_{\gamma\geq 0} \gamma \min_{\lambda\in\Delta_p} \left(\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_\lambda)\right)\\
&\overset{(b)}{=} \min_{h\in\mathcal{H}} \mathcal{L}_{\widehat{\mathcal{D}}_0}(h) + \min_{\lambda\in\Delta_p} \max_{\gamma\geq 0} \gamma \left(\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_\lambda)\right)\\
&= \min_{h\in\mathcal{H}} \min_{\lambda\in\Delta_p} \max_{\gamma\geq 0} \mathcal{L}_{\widehat{\mathcal{D}}_0}(h) + \gamma \left(\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_\lambda)\right)\\
&= \min_{h\in\mathcal{H}} \min_{\lambda\in\Delta_p} \max_{\gamma\geq 0} \max_{h'\in\mathcal{H}} \mathcal{L}_{\widehat{\mathcal{D}}_0}(h) + \gamma \left(\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h')\right),
\end{aligned}$$

where (a) follows from the fact that ℓ is strongly convex. For (b) we break analysis into two cases. If $h \in \mathcal{H}_{\Delta_p}$, then both $\max_{\gamma\geq 0} \gamma \min_{\lambda\in\Delta_p} \left(\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_\lambda)\right)$ and $\min_{\lambda\in\Delta_p} \max_{\gamma\geq 0} \gamma \left(\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_\lambda)\right)$ are zero. Similarly, if $h \notin \mathcal{H}_{\Delta_p}$, then both of these quantities are infinite and can be achieved by $\gamma \to \infty$. This completes the proof. □

While the objective in (11) is linear in λ and convex in $\mathcal{H}$, it is not jointly convex in both λ and $\mathcal{H}$. Hence, the convergence guarantees of the min-max mirror descent algorithm (Nemirovski and Yudin, 1983) do not hold directly. However, one can use the min-max mirror descent algorithm or stochastic minmax mirror descent algorithms (Juditsky et al., 2011; Namkoong and Duchi, 2016; Cotter et al., 2019; Mohri et al., 2019) to obtain heuristic solutions.

To evaluate its usefulness, we first conducted experiments on a synthetic regression example, where the ground truth is known. Let $\mathcal{X} = \mathbf{R}^d$, $\mathcal{Y} = \mathbf{R}$, $p = 4$, and $d = 100$. For each domain k, $\mathcal{D}_k(x)$ is distributed $N(0, \mathbf{I}_d/d)$ and $y = w_k^t x + N(0, \mathbf{I}_d\sigma^2)$, where w_k is distributed according to $N(0, \mathbf{I}_d/d)$ independently. We set $\sigma^2 = 0.01$ and $\lambda^* = [0.7, 0.1, 0.1, 0.1]$. For each source domain k, we use $m_k = 10000$ examples and evaluate the results of the algorithm as we vary m_0, the number of samples in the target domain. The results are presented in Table 1. Observe that the model trained only on the target dataset is significantly worse compared to the loss when λ^* is known. However, LMSA-Min-max performs nearly as well as the known mixture algorithm with as few as 100 samples.

Table 1: Test loss of various algorithms as a function of m_0. All losses are scaled by 1000. The loss when w_k and λ^* are known is 4.47.

m_0	$\mathcal{L}_{\widehat{\mathcal{D}}_0}$	LMSA-Min-max
50	13.16	5.15
100	33.33	4.85
200	9.13	4.80
300	6.73	4.66
400	6.06	4.74

4.4 Lower bound

The bounds of Theorem 1 and Lemma 2 contain a model selection penalty of $\mathcal{O}(\sqrt{p/m_0 \log(1/\epsilon)})$. Using an information-theoretic bound, we show that any algorithm incurs a penalty of $\Omega(\sqrt{p/m_0})$ for some problem settings.

Theorem 3. *For any algorithm $\mathcal{A}$, there exists a set of hypotheses $\mathcal{H}$, a loss function ℓ, and distributions $\mathcal{D}_0, \mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_p$, such that $\mathcal{E}(\lambda^*) = 0$ and the following holds. Given infinitely many samples from $\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_p$ and m_0 samples from $\mathcal{D}_0$, the output of the algorithm $h_{\mathcal{A}}$ satisfies,*

$$\mathbb{E}[\mathcal{L}_{\mathcal{D}_0}(h_{\mathcal{A}})] \geq \min_{h\in\mathcal{H}} \mathcal{L}_{\mathcal{D}_0}(h) + c \cdot \sqrt{\frac{p}{m_0}},$$

where c is a constant and the expectation is over the randomization in the algorithm and the samples.

We relegate the proof to Appendix E. In the proof, we construct an example such that $\mathcal{D}_0$ is a convex combination of source domains, but any algorithm will incur an additional loss of at least $c \cdot \sqrt{p/m_0}$.

5 Alternative techniques

Here, we briefly discuss some existing algorithms, in particular the competitive algorithm of Konstantinov and Lampert (2019), which we will compare with our LMSA algorithms in experiments.

One natural approach to tackle the MSA problem we are studying consists of using discrepancy to find λ, by assigning a higher weight λ_k to a source domain k that is closer to the target distribution $\mathcal{D}_0$ (Wen et al., 2020; Konstantinov and Lampert, 2019). This approach therefore relies on the estimation of the pairwise discrepancies $\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_k, \mathcal{D}_0)$ between each source domain k and the target domain. Specifically, the algorithm of Konstantinov and Lampert (2019) consists of selecting λ by minimizing the following objective:

$$\sum_{k=1}^{p} \lambda_k \mathrm{disc}_{\mathcal{H}}(\mathcal{D}_k, \mathcal{D}_0) + \gamma\sqrt{m\mathfrak{s}(\lambda\|\mathbf{m})},$$

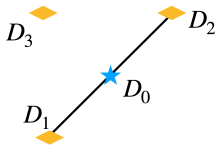

Figure 2: Illustration of the pairwise discrepancy approach.

for some regularization parameter γ.

We argue that this approach can be sub-optimal in various scenarios and that the estimation of the discrepancies in general can lead to weaker guarantees.

To illustrate this, consider the case where the sample size is the same for all source domains and where $\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_1) = \mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_2) = \mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_3) > 0$. Then, for any value γ, the weights assigned by the algorithm coincide: $\lambda_1 = \lambda_2 = \lambda_3$, which is sub-optimal for scenarios such as that of the following example.

Example 1. *Let $p = 3$ and $\mathcal{D}_0 = \frac{\mathcal{D}_1 + \mathcal{D}_2}{2}$, with $\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_1) = \mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_2) = \mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_3) > 0$. Furthermore let the number of samples from each source domain be very large. In this case, observe that $\lambda^* = (0.5, 0.5, 0)$. If we just use the pairwise discrepancies between $\mathcal{D}_0$ and $\mathcal{D}_k$ to set λ, then λ would satisfy $\lambda_1 = \lambda_2 = \lambda_3 = 1/3$, which is far from optimal. The example is illustrated in Figure 2.*

Since the convergence guarantees of this proposed algorithm are based on pairwise discrepancies, loosely speaking, the guarantees are tight in our formulation when $\min_\lambda \mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_\lambda)$ is close to $\min_\lambda \sum_k \lambda_k \mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_k)$. However, for examples similar to above, such an algorithm would be sub-optimal.

Instead of computing pairwise discrepancies, one can compute the discrepancy between $\mathcal{D}_0$ and $\mathcal{D}_\lambda$, that is $\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_\lambda)$, and choose λ to minimize this discrepancy. However, this further requires estimating the discrepancy between the source and target domains and the generalization bound varies as $\tilde{\mathcal{O}}\left(\sqrt{\frac{d}{m_0}}\right)$, which can again be weak or uninformative for small values of m_0. We further discuss this question in more detail in Appendix C.2.

6 Experiments

We evaluated our algorithms and compared them to several baselines on the digits recognition dataset and the visual adaptation Office dataset (Saenko et al., 2010). We first evaluated our algorithm on a standard digits MSA dataset composed of four domains: MNIST (LeCun and Cortes, 2010), MNIST-M (Ganin and Lempitsky, 2015), SVHN (Netzer et al., 2011), and SynthDigits (Ganin and Lempitsky, 2015), by treating one of MNIST, MNIST-M, SVHN, or SynthDigits as the target domain, and the rest as source. We used the same preprocessing and data split as (Zhao et al., 2018), i.e., 20,000 labeled training samples for each domain when used as a source. When a domain is used as the target, we used the first 1280 examples from the 20,000. We also used the same convolution neural network as the digit classification model in (Zhao et al., 2018), with the exception that we used a regular ReLU instead of leaky ReLU. Unlike (Zhao et al., 2018), we trained the models using stochastic gradient descent with a fixed learning rate without weight decay.

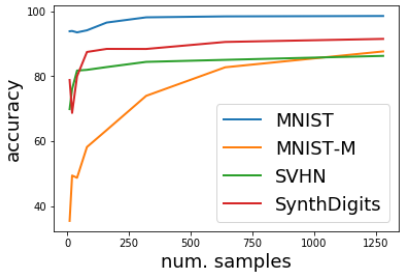

Figure 3: Performance of LMSA as a function of the target sample size m_0.

We used several baselines for comparison:

(i) *best-single-source*: best model trained only on one of the sources;

(ii) *combined-sources*: model trained on dataset obtained by concatenating all the sources;

(iii) *target-only*: model trained only on the limited target data;

(iv) *sources+target*: models trained by combining source and targets;

(v) *sources + target (equal weight)*: models trained by combining source and targets where all of them get the same weight;

(vi) *pairwise discrepancy*: the pairwise discrepancy approach of Konstantinov and Lampert (2019).

Baselines (ii), (iv), and (v) involve data concatenation. For baseline (vi) and the proposed algorithms LMSA, LMSA-Boost, LMSA-Min-max, we report the better results of the following two approaches: one where all 1280 target samples are treated as $\widehat{\mathcal{D}}_0$ and one where 1024 random samples are treated as a separate new source and 256 samples are used as samples from $\widehat{\mathcal{D}}_0$.

The results are presented in Table 2. Our LMSA algorithms perform well compared to the baselines. We note that LMSA-Min-max performed better using all 1280 target samples as $\widehat{\mathcal{D}}_0$, whereas Konstantinov and Lampert (2019), LMSA, and LMSA-Boost performed better using 1024 target samples as a separate

Table 2: Test accuracy of algorithms for different target domains. The instances where the proposed algorithm performs better than all the baselines are highlighted. The standard deviations are calculated over ten runs.

algorithm	MNIST	MNIST-M	SVHN	SynthDigits
best-single-source	98.0(0.1)	56.0(0.7)	83.1(0.4)	86.1(0.4)
combined-sources	98.4(0.1)	67.2(0.4)	81.1(0.6)	87.2(0.1)
target-only	96.4(0.1)	86.3(0.5)	77.7(0.5)	88.5(0.2)
sources+target	**98.6**(0.1)	74.8(0.5)	85.4(0.3)	90.6(0.3)
sources+target (equal weight)	97.4(0.2)	77.8(0.6)	85.5(0.4)	89.8(0.3)
(Konstantinov and Lampert, 2019)	98.4(0.1)	84.6(0.5)	86.3(0.4)	90.5(0.4)
LMSA	**98.5**(0.1)	**87.6**(0.6)	86.2(0.4)	**91.5**(0.2)
LMSA-Boost	98.4(0.2)	**88.1**(0.4)	86.1(0.4)	**91.4**(0.3)
LMSA-Min-max	98.0(0.3)	**89.5**(0.4)	**86.7**(0.4)	**91.7**(0.3)

Table 3: Test accuracy of algorithms for different target domains for the Office dataset. The instances where the proposed algorithm performs better than all the baselines are highlighted.

algorithm	amazon	dslr	webcam
best-single-source	58.8(1.0)	98.7(0.6)	94.0(1.3)
combined-sources	62.0(0.7)	97.0(0.9)	91.9(1.3)
target-only	77.8(0.8)	96.4(0.8)	91.5(0.9)
sources+target	77.7(0.6)	98.8(0.6)	96.6(0.7)
sources+target (equal weight)	76.7(0.5)	99.4(0.5)	96.8(0.6)
LMSA	**78.1**(0.5)	**99.5**(0.4)	**97.5**(0.8)
LMSA-Boost	**78.6**(0.4)	**99.5**(0.5)	**97.6**(0.3)
LMSA-Min-max	77.7(0.7)	98.9(0.3)	**97.0**(0.5)

new source domain. As expected, the performance of proposed algorithms is better than that of the unsupervised domain adaptation algorithms of (Zhao et al., 2018) (see Table 2 in their paper), due to the availability of labeled target samples.

Figure 3 shows the performance of the LMSA as a function of the number of target samples. Of the four target domains, MNIST is the easiest domain and requires very few target samples to achieve good accuracy, and MNIST-M is the hardest and requires many target samples to achieve good accuracy. We omit the curves for LMSA-Boost and LMSA-Min-max because they are similar.

In addition to the digit recognition task, we considered the standard visual adaptation Office dataset (Saenko et al., 2010), which has 3 domains: amazon, dslr, and webcam. This dataset consists of 31 categories of objects commonly found in an office environment. The amazon domain consists of 2817 images, dslr 498, and webcam 795, for a total of 4110 images. For source domains, we used all available samples, and for target domains, we used 20 samples per category for amazon and 8 for both dslr and webcam. However, rather than AlexNet, we used the ResNet50 (He et al., 2015) architecture pre-trained on ImageNet.

Similar to the digits experiment, for baseline (*vi*) and the proposed algorithms LMSA, LMSA-Boost, LMSA-Min-max, we report the better results of the following two approaches: one where all target samples are treated as $\widehat{\mathcal{D}}_0$ and one where some percentage of random samples are treated as a separate new source and the remaining samples are treated as samples from $\widehat{\mathcal{D}}_0$. For the latter approach, due to the limited size of the Office dataset, we used cross validation with 5 different splits to determine what percentage of samples to treat as a separate new source. As discussed in Appendix C.2, empirical estimates of the discrepancy based on small samples are unreliable for small datasets and large model classes. Our experiments corroborated this theory. Since the Office dataset is small and the ResNet50 architecture has many parameters and our loss (log-loss) is unbounded, the empirical pairwise discrepancy estimate was infinite. Hence, we omit the results for the pairwise discrepancy approach of (Konstantinov and Lampert, 2019). The results are presented in Table 3. Our LMSA algorithms perform well compared to the baselines.

7 Conclusion

We presented a theoretical and algorithmic study of multiple-source domain adaptation with limited target labeled data. The algorithms we presented benefit from very favorable learning guarantees and further perform well in our experiments, typically surpassing other baselines. We hope that our analysis will serve as a tool for further theoretical studies of this problem and other related adaptation problems and algorithms.

References

S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira. Analysis of representations for domain adaptation. In *Advances in neural information processing systems*, pages 137–144, 2007.

S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. W. Vaughan. A theory of learning from different domains. *Machine learning*, 79(1-2): 151–175, 2010.

G. Blanchard, G. Lee, and C. Scott. Generalizing from several related classification tasks to a new unlabeled sample. In *NIPS*, pages 2178–2186, 2011.

J. Blitzer, M. Dredze, and F. Pereira. Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification. In *Proceedings of ACL 2007*, Prague, Czech Republic, 2007.

J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. Wortman. Learning bounds for domain adaptation. In *Advances in neural information processing systems*, pages 129–136, 2008.

C. Cortes and M. Mohri. Domain adaptation in regression. In *Proceedings of ALT*, pages 308–323, 2011.

C. Cortes and M. Mohri. Domain adaptation and sample bias correction theory and algorithm for regression. *Theor. Comput. Sci.*, 519:103–126, 2014.

A. Cotter, M. Gupta, H. Jiang, N. Srebro, K. Sridharan, S. Wang, B. Woodworth, and S. You. Training well-generalizing classifiers for fairness metrics and other data-dependent constraints. In *International Conference on Machine Learning*, pages 1397–1405. PMLR, 2019.

T. M. Cover and J. A. Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

K. Crammer, M. J. Kearns, and J. Wortman. Learning from multiple sources. *Journal of Machine Learning Research*, 9(Aug):1757–1774, 2008.

M. Dredze, J. Blitzer, P. Talukdar, K. Ganchev, J. Graca, and F. Pereira. Frustratingly hard domain adaptation for dependency parsing. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 1051–1055, 2007.

L. Duan, I. W. Tsang, D. Xu, and T. Chua. Domain adaptation from multiple sources via auxiliary classifiers. In *ICML*, volume 382, pages 289–296, 2009.

L. Duan, D. Xu, and I. W. Tsang. Domain adaptation from multiple sources: A domain-dependent regularization approach. *IEEE Transactions on Neural Networks and Learning Systems*, 23(3):504–518, 2012.

B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars. Unsupervised visual domain adaptation using subspace alignment. In *Proceedings of the IEEE international conference on computer vision*, pages 2960–2967, 2013.

C. Gan, T. Yang, and B. Gong. Learning attributes equals multi-source domain generalization. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 87–97, 2016.

Y. Ganin and V. S. Lempitsky. Unsupervised domain adaptation by backpropagation. In *ICML*, volume 37, pages 1180–1189, 2015.

Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

J.-L. Gauvain and C.-H. Lee. Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains. *IEEE transactions on speech and audio processing*, 2(2):291–298, 1994.

M. Ghifary, W. Bastiaan Kleijn, M. Zhang, and D. Balduzzi. Domain generalization for object recognition with multi-task autoencoders. In *Proceedings of the IEEE international conference on computer vision*, pages 2551–2559, 2015.

B. Gong, Y. Shi, F. Sha, and K. Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *CVPR*, pages 2066–2073, 2012.

B. Gong, K. Grauman, and F. Sha. Connecting the dots with landmarks: Discriminatively learning domain-invariant features for unsupervised domain adaptation. In *ICML*, volume 28, pages 222–230, 2013a.

B. Gong, K. Grauman, and F. Sha. Reshaping visual datasets for domain adaptation. In *NIPS*, pages 1286–1294, 2013b.

K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition, 2015.

J. Hoffman, B. Kulis, T. Darrell, and K. Saenko. Discovering latent domains for multisource domain adaptation. In *ECCV*, volume 7573, pages 702–715, 2012.

J. Hoffman, M. Mohri, and N. Zhang. Algorithms and theory for multiple-source adaptation. In *Proceedings of NeurIPS*, pages 8256–8266, 2018.

F. Jelinek. *Statistical methods for speech recognition*. MIT press, 1997.

I.-H. Jhuo, D. Liu, D. Lee, and S.-F. Chang. Robust visual domain adaptation with low-rank reconstruction. In *2012 IEEE conference on computer vision and pattern recognition*, pages 2168–2175. IEEE, 2012.

J. Jiang and C. Zhai. Instance weighting for domain adaptation in nlp. In *Proceedings of the 45th annual meeting of the association of computational linguistics*, pages 264–271, 2007.

A. Juditsky, A. Nemirovski, and C. Tauvel. Solving variational inequalities with stochastic mirror-prox algorithm. *Stochastic Systems*, 1(1):17–58, 2011.

A. Khosla, T. Zhou, T. Malisiewicz, A. A. Efros, and A. Torralba. Undoing the damage of dataset bias. In *ECCV*, volume 7572, pages 158–171, 2012.

D. Kifer, S. Ben-David, and J. Gehrke. Detecting change in data streams. *Proceedings of the 30th International Conference on Very Large Data Bases*, 2004.

N. Konstantinov and C. Lampert. Robust learning from untrusted sources. In *International Conference on Machine Learning*, pages 3488–3498, 2019.

V. Kuznetsov and M. Mohri. Learning theory and algorithms for forecasting non-stationary time series. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 541–549, 2015.

Y. LeCun and C. Cortes. MNIST handwritten digit database. http://yann.lecun.com/exdb/mnist/, 2010. URL http://yann.lecun.com/exdb/mnist/.

C. J. Leggetter and P. C. Woodland. Maximum likelihood linear regression for speaker adaptation of continuous density hidden markov models. *Computer speech & language*, 9(2):171–185, 1995.

H. Liu, M. Shao, and Y. Fu. Structure-preserved multi-source domain adaptation. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*, pages 1059–1064. IEEE, 2016.

Y. Mansour, M. Mohri, and A. Rostamizadeh. Multiple source adaptation and the Rényi divergence. In *UAI*, pages 367–374, 2009a.

Y. Mansour, M. Mohri, and A. Rostamizadeh. Domain adaptation with multiple sources. In *NIPS*, pages 1041–1048, 2009b.

Y. Mansour, M. Mohri, and A. Rostamizadeh. Domain adaptation: Learning bounds and algorithms. In *COLT*, 2009c.

M. Mohri and A. Muñoz Medina. New analysis and algorithm for learning with drifting distributions. In *International Conference on Algorithmic Learning Theory*, pages 124–138. Springer, 2012.

M. Mohri, A. Rostamizadeh, and A. Talwalkar. *Foundations of Machine Learning*. MIT Press, second edition, 2018.

M. Mohri, G. Sivek, and A. T. Suresh. Agnostic federated learning. In *International Conference on Machine Learning*, pages 4615–4625. PMLR, 2019.

S. Motiian, Q. Jones, S. Iranmanesh, and G. Doretto. Few-shot adversarial domain adaptation. In *Advances in Neural Information Processing Systems*, pages 6670–6680, 2017a.

S. Motiian, M. Piccirilli, D. A. Adjeroh, and G. Doretto. Unified deep supervised domain adaptation and generalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5715–5725, 2017b.

K. Muandet, D. Balduzzi, and B. Schölkopf. Domain generalization via invariant feature representation. In *ICML*, volume 28, pages 10–18, 2013.

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

A. S. Nemirovski and D. B. Yudin. *Problem complexity and Method Efficiency in Optimization*. Wiley, 1983.

Y. Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012.

Y. Netzer, T. Wang, A. Coates, A. Bissacco, B. Wu, and A. Ng. Reading digits in natural images with unsupervised feature learning. *NIPS*, 01 2011.

Z. Pei, Z. Cao, M. Long, and J. Wang. Multi-adversarial domain adaptation. In *AAAI*, pages 3934–3941, 2018.

X. Peng, Q. Bai, X. Xia, Z. Huang, K. Saenko, and B. Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1406–1415, 2019.

D. Pollard. *Convergence of stochastic processes*. Springer Science & Business Media, 2012.

K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *ECCV*, volume 6314, pages 213–226, 2010.

K. Saito, D. Kim, S. Sclaroff, T. Darrell, and K. Saenko. Semi-supervised domain adaptation via minimax entropy. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 8050–8058, 2019.

Q. Sun, R. Chattopadhyay, S. Panchanathan, and J. Ye. A two-stage weighting framework for multi-source domain adaptation. In *Advances in neural information processing systems*, pages 505–513, 2011.

E. Tzeng, J. Hoffman, T. Darrell, and K. Saenko. Simultaneous deep transfer across domains and tasks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4068–4076, 2015.

T. Wang, X. Zhang, L. Yuan, and J. Feng. Few-shot adaptive faster r-cnn. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7173–7182, 2019.

J. Wen, R. Greiner, and D. Schuurmans. Domain aggregation networks for multi-source domain adaptation. In *International Conference on Machine Learning*, pages 10214–10224. PMLR, 2020.

J. Yang, R. Yan, and A. G. Hauptmann. Cross-domain video concept detection using adaptive svms. In *ACM Multimedia*, pages 188–197, 2007.

N. Zhang, M. Mohri, and J. Hoffman. Multiple-source adaptation theory and algorithms. *Annals of Mathematics and Artificial Intelligence*, pages 1–34, 2020.

H. Zhao, S. Zhang, G. Wu, J. M. Moura, J. P. Costeira, and G. J. Gordon. Adversarial multiple source domain adaptation. In *Advances in neural information processing systems*, pages 8559–8570, 2018.

Supplementary material: A Theory of Multiple-Source Adaptation with Limited Target Labeled Data

A Related on domain adaptation

As stated in the introduction, various scenarios of adaptation can be distinguished depending on parameters such as the number of source domains available, the presence or absence of target labeled data, and access to labeled source data or only to predictors trained on each source domain. Single source domain adaptation has been studied in several papers including (Kifer et al., 2004; Ben-David et al., 2010; Mansour et al., 2009c).

Several algorithms have been proposed for multiple-source adaptation. Khosla et al. (2012); Blanchard et al. (2011) proposed to combine all the source data and train a single model. Duan et al. (2009, 2012) used unlabeled target data to obtain a regularizer. Domain adaptation via adversarial learning was studied by Pei et al. (2018); Zhao et al. (2018). Crammer et al. (2008) considered learning models for each source domain, using close-by data of other domains. Gong et al. (2012) ranked multiple source domains by how well they can adapt to a target domain. Other solutions to multiple-source domain adaptation include, clustering (Liu et al., 2016), learning domain-invariant features (Gong et al., 2013a), learning intermediate representations (Jhuo et al., 2012), subspace alignment techniques (Fernando et al., 2013), attributes detection (Gan et al., 2016), using a linear combination of pre-trained classifiers (Yang et al., 2007), using multitask auto-encoders (Ghifary et al., 2015), causal approaches (Sun et al., 2011), two-state weighting approaches (Sun et al., 2011), moments alignment techniques (Peng et al., 2019) and domain-invariant component analysis (Muandet et al., 2013).

B Proof of equation (2)

By the definition of discrepancy,

$$\mathcal{L}_{\mathcal{D}_0}(h_{\overline{\mathcal{D}}_\lambda}) \leq \mathcal{L}_{\mathcal{D}_\lambda}(h_{\overline{\mathcal{D}}_\lambda}) + \mathrm{disc}_{\mathcal{H}}(\mathcal{D}_\lambda, \mathcal{D}_0).$$

Similarly,

$$\begin{aligned}\mathcal{L}_{\mathcal{D}_0}(h_{\mathcal{D}_0}) &= \mathcal{L}_{\mathcal{D}_0}(h_{\mathcal{D}_0}) - \mathcal{L}_{\mathcal{D}_\lambda}(h_{\mathcal{D}_0}) + \mathcal{L}_{\mathcal{D}_\lambda}(h_{\mathcal{D}_0}) \\ &\geq \mathcal{L}_{\mathcal{D}_\lambda}(h_{\mathcal{D}_0}) - \mathrm{disc}_{\mathcal{H}}(\mathcal{D}_\lambda, \mathcal{D}_0) \\ &\geq \mathcal{L}_{\mathcal{D}_\lambda}(h_{\mathcal{D}_\lambda}) - \mathrm{disc}_{\mathcal{H}}(\mathcal{D}_\lambda, \mathcal{D}_0)\end{aligned}$$

Combining the above two equations yields

$$\mathcal{L}_{\mathcal{D}_0}(h_{\overline{\mathcal{D}}_\lambda}) - \mathcal{L}_{\mathcal{D}_0}(h_{\mathcal{D}_0}) \leq \mathcal{L}_{\mathcal{D}_\lambda}(h_{\overline{\mathcal{D}}_\lambda}) - \mathcal{L}_{\mathcal{D}_\lambda}(h_{\mathcal{D}_\lambda}) + 2\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_\lambda, \mathcal{D}_0).$$

Next observe that, by rearranging terms,

$$\begin{aligned}\mathcal{L}_{\mathcal{D}_\lambda}(h_{\overline{\mathcal{D}}_\lambda}) - \mathcal{L}_{\mathcal{D}_\lambda}(h_{\mathcal{D}_\lambda}) &= \mathcal{L}_{\mathcal{D}_\lambda}(h_{\overline{\mathcal{D}}_\lambda}) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_{\overline{\mathcal{D}}_\lambda}) + \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_{\mathcal{D}_\lambda}) - \mathcal{L}_{\mathcal{D}_\lambda}(h_{\mathcal{D}_\lambda}) \\ &\quad + \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_{\overline{\mathcal{D}}_\lambda}) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_{\mathcal{D}_\lambda})\end{aligned}$$

However, by the definition of $h_{\overline{\mathcal{D}}_\lambda}$,

$$\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_{\overline{\mathcal{D}}_\lambda}) \leq \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_{\mathcal{D}_\lambda}).$$

Hence,

$$\begin{aligned}\mathcal{L}_{\mathcal{D}_\lambda}(h_{\overline{\mathcal{D}}_\lambda}) - \mathcal{L}_{\mathcal{D}_\lambda}(h_{\mathcal{D}_\lambda}) &\leq \mathcal{L}_{\mathcal{D}_\lambda}(h_{\overline{\mathcal{D}}_\lambda}) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_{\overline{\mathcal{D}}_\lambda}) + \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_{\mathcal{D}_\lambda}) - \mathcal{L}_{\mathcal{D}_\lambda}(h_{\mathcal{D}_\lambda}) \\ &\leq 2\sup_h |\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h) - \mathcal{L}_{\mathcal{D}_\lambda}(h)|,\end{aligned}$$

where the last inequality follows by taking the supremum. Combining the above equations yields the proof.

C Previous work

C.1 Bregman divergence based non-negative matrix factorization

A natural algorithm is a two step process, where we first identify the optimal λ by minimizing

$$\min_{\lambda \in \Delta_p} \mathsf{B}(\widehat{\mathcal{D}}_0 || \overline{\mathcal{D}}_\lambda),$$

where B is a suitable Bregman divergence. We can then use λ to minimize the weighted loss. However, this approach has both practical and theoretical issues. On the practical side, if $\mathcal{X}$ is a continuous space, then the empirical distribution $\overline{\mathcal{D}}_\lambda$ would be a point mass distribution over observed points and would never converge to the true distribution $\mathcal{D}_\lambda$. To overcome this, we need to first use $\overline{\mathcal{D}}_\lambda$ to estimate the distribution $\mathcal{D}_\lambda$ via kernel density estimation or other methods and then use the estimate instead of $\overline{\mathcal{D}}_\lambda$. Even if we use these methods and find λ, it is likely that we would overfit as the generalization of the algorithm depends on the covering number of $\{\mathcal{D}_\lambda : \lambda \in \Delta_p\}$, which in general can be much larger than that of the class of hypotheses $\mathcal{H}$. Hence such an algorithm would not incur generalization loss of $\mathcal{E}(\lambda^*)$. One can try to reduce the generalization error by using a discrepancy based approach, which we discuss in the next section.

C.2 A convex combination discrepancy-based algorithm

Since pairwise discrepancies would result in identifying a sub-optimal λ, instead of just considering the pairwise discrepancies, one can consider the discrepancy between $\mathcal{D}_0$ and any $\mathcal{D}_\lambda$. Since

$$\mathcal{L}_{\mathcal{D}_0}(h) \le \min_{\lambda \in \Delta_p} \mathcal{L}_{\mathcal{D}_\lambda}(h) + \mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_\lambda),$$

and the learner has more data from $\mathcal{D}_\lambda$ than from $\mathcal{D}_0$, a natural algorithm is to minimize $\mathcal{L}_{\mathcal{D}_\lambda}(h) + \mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_\lambda)$. However, note that this requires estimating both the discrepancy $\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_\lambda)$ and the expected loss over $\mathcal{L}_{\mathcal{D}_\lambda}(h)$. In order to account for both terms, we propose to minimize the upper bound on $\min_{\lambda \in \Delta_p} \mathcal{L}_{\mathcal{D}_\lambda}(h) + \mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_\lambda)$,

$$\min_{\lambda} \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h) + C_\epsilon(\lambda), \tag{12}$$

where $C_\epsilon(\lambda)$ is given by,

$$\mathrm{disc}_{\mathcal{H}}(\widehat{\mathcal{D}}_0, \overline{\mathcal{D}}_\lambda) + \frac{c\sqrt{d + \log \frac{1}{\delta}}}{\sqrt{m_0}} + \epsilon M + \frac{cM\sqrt{\mathfrak{s}(\lambda||\mathbf{m})}}{\sqrt{m}} \cdot \left(\sqrt{d \log \frac{em}{d} + p \log \frac{p}{\epsilon\delta}} \right),$$

for some constant c. We first show that $\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h) + C_\epsilon(\lambda)$ is an upper bound on $\mathcal{L}_{\mathcal{D}_0}(h)$.

Lemma 3. *With probability at least $1 - 2\delta$, for all $h \in \mathcal{H}$ and $\lambda \in \Delta_p$,*

$$|\mathcal{L}_{\mathcal{D}_0}(h) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h)| \le C_\epsilon(\lambda).$$

Proof. By (2) and Proposition 1, with probability at least $1 - \delta$,

$$|\mathcal{L}_{\mathcal{D}_0}(h) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h)| \le 2\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_\lambda) + \frac{4M\sqrt{s(\lambda||\mathbf{m})}}{\sqrt{m}} \cdot \left(\sqrt{d \log \frac{em}{d} + \log \frac{1}{\delta}} \right).$$

Hence, by the union bound over an ϵ-ℓ_1 cover of Δ_p yields, with probability $\ge 1 - \delta$, for all $\lambda \in \Delta_p$,

$$|\mathcal{L}_{\mathcal{D}_0}(h) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h)| \le 2\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_\lambda) + \epsilon M + \frac{4M\sqrt{\mathfrak{s}(\lambda||\mathbf{m})}}{\sqrt{m}} \cdot \left(\sqrt{d \log \frac{em}{d} + p \log \frac{p}{\epsilon\delta}} \right).$$

With probability at least $1 - \delta$, discrepancy can be estimated as

$$|\mathrm{disc}_{\mathcal{H}}(\mathcal{D}_0, \mathcal{D}_\lambda) - \mathrm{disc}_{\mathcal{H}}(\widehat{\mathcal{D}}_0, \overline{\mathcal{D}}_\lambda)| \le \epsilon M + \frac{c\sqrt{d + \log \frac{1}{\delta}}}{\sqrt{m_0}} + \frac{cM\sqrt{\mathfrak{s}(\lambda||\mathbf{m})}}{\sqrt{m}} \cdot \left(\sqrt{d \log \frac{em}{d} + p \log \frac{p}{\epsilon\delta}} \right),$$

for some constant $c > 0$. Combining the above equations yields, with probability at least $1 - 2\delta$,

$$\max_{\lambda} |\mathcal{L}_{\mathcal{D}_0}(h) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h)| \le C_\epsilon(\lambda).$$

□

Let h_R be the solution to (12), we now give a generalization bound for the above algorithm.

Lemma 4. *With probability at least* $1-2\delta$*, the solution* h_R *for* (12) *satisfies*

$$\mathcal{L}_{\mathcal{D}_0}(h_R) \le \min_{h\in\mathcal{H}} \mathcal{L}_{\mathcal{D}_0}(h) + 2\min_{\lambda} C_\epsilon(\lambda).$$

Proof. By Lemma 3, with probability at least $1-2\delta$,

$$\min_{\lambda\in\Delta_p} |\mathcal{L}_{\mathcal{D}_0}(h) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h)| \le \min_{\lambda\in\Delta_p} C_\epsilon(\lambda).$$

Let h_R be the output of the algorithm and $h_{\mathcal{D}_0}$ be the minimizer of $\mathcal{L}_{\mathcal{D}_0}(h)$.

$$\begin{aligned}
L_{\mathcal{D}_0}(h_R) - \mathcal{L}_{\mathcal{D}_0}(h_{\mathcal{D}_0}) &\le \min_\lambda\big(\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_R) + C_\epsilon(\lambda)\big) - \max_\lambda\big(\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_{\mathcal{D}_0}) - C_\epsilon(\lambda)\big)\\
&\le \min_\lambda\big(\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_R) + C_\epsilon(\lambda)\big) + \min_\lambda\big(-\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_{\mathcal{D}_0}) + C_\epsilon(\lambda)\big)\\
&\le \min_\lambda\big(\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_R) + C_\epsilon(\lambda) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_{\mathcal{D}_0}) + C_\epsilon(\lambda)\big)\\
&\le 2\min_\lambda C_\epsilon(\lambda),
\end{aligned}$$

where the last inequality follows from the fact that h_R is the minimizer of (12).

□

The above bound is comparable to the model trained on only target data as $C_\epsilon(\lambda)$ contains $\mathcal{O}\left(\sqrt{\frac{d}{m_0}}\right)$, which can be large for a small values of m_0. This bound can be improved on certain favorable cases when $\mathcal{D}_0 = \mathcal{D}_k$ for some known k. In this case if we use the same set of samples for $\widehat{\mathcal{D}}_0$ and $\widehat{\mathcal{D}}_k$, then the bound can be improved to $\mathcal{O}\left(\frac{\sqrt{d(1-\lambda_k)}}{\sqrt{m_0}}\right)$, which in favorable cases such that λ_k is large, yields a better bound than the target-only model.

D Proof of Lemma 1

By the strong convexity of ℓ,

$$\begin{aligned}
\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_{\lambda'}) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_\lambda) &\ge \nabla\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_\lambda)\cdot(h_{\lambda'} - h_\lambda) + \frac{\mu}{2}\|h_{\lambda'} - h_\lambda\|^2\\
&= \frac{\mu}{2}\|h_{\lambda'} - h_\lambda\|^2,
\end{aligned}$$

where the equality follows from the definition of h_λ. Similarly, since the function ℓ is bounded by M

$$\begin{aligned}
\mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_{\lambda'}) - \mathcal{L}_{\overline{\mathcal{D}}_\lambda}(h_\lambda) &\le \mathcal{L}_{\overline{\mathcal{D}}_{\lambda'}}(h_{\lambda'}) - \mathcal{L}_{\overline{\mathcal{D}}_{\lambda'}}(h_\lambda) + \|\lambda-\lambda'\|_1 M\\
&\le -\nabla\mathcal{L}_{\overline{\mathcal{D}}_{\lambda'}}(h_{\lambda'})\cdot(h_{\lambda'} - h_\lambda) - \frac{\mu}{2}\|h_{\lambda'} - h_\lambda\|^2 + \|\lambda-\lambda'\|_1 M\\
&= -\frac{\mu}{2}\|h_{\lambda'} - h_\lambda\|^2 + \|\lambda-\lambda'\|_1 M.
\end{aligned}$$

Combining the above equations,

$$\mu\|h_{\lambda'} - h_\lambda\|^2 \le M\|\lambda-\lambda'\|_1.$$

Hence for any distribution $\mathcal{D}_0$,

$$\begin{aligned}
|\mathcal{L}_{\mathcal{D}_0}(h_{\lambda'}) - \mathcal{L}_{\mathcal{D}_0}(h_\lambda)| &\le |\nabla\mathcal{L}_{\mathcal{D}_0}(h_\lambda)\cdot(h_{\lambda'} - h_\lambda)|\\
&\le |\nabla\mathcal{L}_{\mathcal{D}_0}(h_\mathcal{D})\cdot(h_{\lambda'} - h_\lambda)|\\
&= G\|h_{\lambda'} - h_\lambda\|\\
&= \frac{G\sqrt{M}}{\sqrt{\mu}}\cdot\|\lambda-\lambda'\|_1^{1/2}.
\end{aligned}$$

E Proof of Theorem 3

Let p be a multiple of four. Let $\mathcal{X} = \{1, 2, \ldots, p/2\}$ and $\mathcal{Y} = \{0, 1\}$. For all $k \le p/2$, and $x \in \mathcal{X}$, let $\mathcal{D}_k(x) = \frac{2}{p}$. For every even k, let $\mathcal{D}_k(1|\lceil k/2 \rceil) = 1$ and for every odd k, $\mathcal{D}_k(1|\lceil k/2 \rceil) = 0$. For remaining x and k, let $\mathcal{D}_k(1|x) = \frac{1}{2}$.

Let $\mathcal{H}$ be the set of all mappings from $\mathcal{X} \to \mathcal{Y}$ and the loss function be zero-one loss. Let $\mathcal{D}_0 = \mathcal{D}_\lambda$ for some λ. Hence, the optimal estimator h^* is

$$h^*_\lambda(1|x) = 1_{\lambda_{2x} > \lambda_{2x-1}}.$$

Given infinitely number of samples from each $\mathcal{D}_k$, the learner knows the distributions $\mathcal{D}_k$. Hence, roughly speaking the algorithm has to find if $\lambda_{2x} > \lambda_{2x-1}$ for each x.

Let $\epsilon = \frac{1}{100} \cdot \sqrt{\frac{p}{m_0}}$. We restrict $\lambda \in \Lambda$, where Λ is defined as follows. Let Λ be the set of all distributions such that for each $\lambda \in \Lambda$ and x,

$$\lambda_{2x} + \lambda_{2x-1} = \frac{2}{p},$$

and $\lambda_{2x} \in \{\frac{1+\epsilon}{p}, \frac{1-\epsilon}{p}\}$. Note that $|\Lambda| = 2^{p/4}$. For $x \le p/4$, let $s_x = \{2x, 2x-1\}$. Let m_{s_x} be the number of occurrences of elements from s_x. Given m_{s_x}, m_{2x} and m_{2x-1} are random variables from Binomial distribution with parameters m_{s_x} and $\frac{\lambda_{2x}}{\lambda_{2x}+\lambda_{2x-1}}$. This reduces the problem of learning the best classifier into testing $p/2$ Bernoulli distributions and we can use standard tools from information theory such as Fano's inequality (Cover and Thomas, 2012) to provide a lower bound. We provide a proof sketch.

Since $\sum_{x=1}^{p/4} m_{s_x} = m_0$, there are at least $p/8$ values of s_x for which $m_{s_x} \le 8m_0/p$. Consider one such s_x, where $m_{s_x} \le 8m_0/p$. For that x, given m_{s_x} samples from s_x, by Fano's inequality, with probability at least $1/4$, any algorithm cannot differentiate between $\lambda_{2x} > \lambda_{2x-1}$ and $\lambda_{2x} < \lambda_{2x-1}$. Thus, with probability at least $1/4$, any algorithm incorrectly finds the wrong hypothesis for h, and hence,

$$\mathbb{E}[\mathcal{L}(h)|x \in s_x] \ge \mathbb{E}[\mathcal{L}(h^*_\lambda)|x \in s_x] + c\frac{|\lambda_{2x} - \lambda_{2x-1}|}{\lambda_{2x} + \lambda_{2x-1}} \ge \mathbb{E}[\mathcal{L}(h^*_\lambda)|x \in s_x] + c\epsilon,$$

for some constant c. Averaging over all symbols x, yields

$$\begin{aligned}
\mathbb{E}[\mathcal{L}(h)] &= \sum_{s_x} \mathcal{D}_\lambda(s_x)\, \mathbb{E}[\mathcal{L}(h)|x \in s_x] \\
&= \sum_{s_x : m_{s_x} \le 8m_0/p} \frac{2}{p}\, \mathbb{E}[\mathcal{L}(h)|x \in s_x] + \sum_{s_x : m_{s_x} > 8m_0/p} \frac{2}{p}\, \mathbb{E}[\mathcal{L}(h)|x \in s_x] \\
&\ge \sum_{s_x : m_{s_x} \le 8m_0/p} \frac{2}{p}\, (\mathbb{E}[\mathcal{L}(h^*_\lambda)|x \in s_x] + c\epsilon) + \sum_{s_x : m_{s_x} > 8m_0/p} \frac{2}{p}\, \mathbb{E}[\mathcal{L}(h^*_\lambda)|x \in s_x] \\
&= \mathbb{E}[\mathcal{L}(h^*_\lambda)] + \sum_{s_x : m_{s_x} \le 8m_0/p} \frac{2}{p} c\epsilon \\
&\ge \mathbb{E}[\mathcal{L}(h^*_\lambda)] + \frac{p}{8} \cdot \frac{2}{p} c\epsilon \\
&= \mathbb{E}[\mathcal{L}(h^*_\lambda)] + \frac{c\epsilon}{4} = \mathbb{E}[\mathcal{L}(h^*_\lambda)] + \frac{c\epsilon}{400}\sqrt{\frac{p}{m_0}}.
\end{aligned}$$