Generalized Discrepancy for Domain Adaptation

Corinna Cortes  
Google Research  
111 8th Avenue  
New York, NY 10011

Mehryar Mohri  
Courant Institute and Google  
251 Mercer Street  
New York, NY 10012

Andrés Muñoz Medina  
Google Research  
111 8th Avenue  
New York, NY 10011

Abstract

We present a new algorithm for domain adaptation improving upon the discrepancy minimization algorithm (DM), which was previously shown to outperform a number of popular algorithms designed for this task. Unlike most previous approaches adopted for domain adaptation, our algorithm does not consist of a fixed reweighting of the losses over the training sample. Instead, it uses a reweighting that depends on the hypothesis considered and is based on the minimization of a new measure of generalized discrepancy. We give a detailed description of our algorithm and show that it can be formulated as a convex optimization problem. We also present a detailed theoretical analysis of its learning guarantees, which helps us select its parameters. Finally, we report the results of experiments demonstrating that it improves upon the DM algorithm in several tasks.

1 Introduction

A standard assumption in much of learning theory and algorithms is that the training and test data are sampled from the same distribution. In practice, however, this assumption often does not hold. The learner then faces the more challenging problem of domain adaptation where the source and target distributions are somewhat different. This problem arises in a variety of applications such as natural language processing and computer vision [Dredze et al., 2007, Blitzer et al., 2007b, Jiang and Zhai, 2007, Leggetter and Woodland, 1995, Martínez, 2002, Hoffman et al., 2014] and many others.

Over the past decade, a large body of literature has been devoted to the theory of domain adaptation. This theory was pioneered by Ben-David et al. [2006] and Blitzer et al. [2007a] who introduced the $d_A$-distance as an appropriate way to measure divergence between distributions in domain adaptation. This work was later extended in [Mansour et al., 2009, Cortes and Mohri, 2011] where the notion of discrepancy was introduced as a generalization of the $d_A$-distance to arbitrary loss functions.

The learning guarantees given in the aforementioned papers motivated a discrepancy minimization (DM) algorithm [Cortes and Mohri, 2013] with good learning guarantees and with state-of-the-art performance on several data sets. At a high level, the DM algorithm reweights the loss at every point on the source sample to make it more similar to the target distribution. The notion of reweighting is natural and has been proposed by different authors as a way of minimizing some notion of divergence between distributions [Huang et al., 2006, Sugiyama et al., 2007]. While this is an intuitive idea, a drawback of this type of algorithms is that the reweighting of the losses is fixed and independent of the labeling function, which allows for little flexibility at training time. For instance, the DM algorithm selects weights to minimize a quantity defined as a maximum over all hypothesis pairs, thereby ignoring the fact that most of these hypothesis are not even close to the optimal hypothesis considered by the learning algorithm, thereby making the choice of weights too conservative.

To address this issue, we propose a hypothesis dependent reweighting. The goal of our algorithm is to ensure that for every hypothesis the loss function on the source data is as close as possible to
the loss function on the target data. By doing so, we show that the minimizer of the reweighted loss function will also be close to the minimizer of the target loss function.

This paper is organized as follows: we first describe the learning scenario of domain adaptation in Section 2. Then, we give a detailed description of our algorithm and prove that it can be formulated as a convex optimization problem (Section 3). Finally, we report the results of experiments demonstrating that our algorithm improves upon the DM algorithm in several tasks in Section 4.

2 Learning scenario

Let \( \mathcal{X} \) denote the input space and \( \mathcal{Y} \subseteq \mathbb{R} \) the output space. We define a domain as a pair formed by a distribution over \( \mathcal{X} \) and a target labeling function mapping from \( \mathcal{X} \) to \( \mathcal{Y} \). Throughout the paper, \((Q,f_Q)\) denotes the source domain and \((P,f_P)\) the target domain with \(Q\) the source and \(P\) the target distribution over \( \mathcal{X} \) while \( f_Q, f_P : \mathcal{X} \rightarrow \mathcal{Y} \) are the source and target labeling functions respectively.

In the scenario of domain adaptation we consider, the learner receives two samples: a labeled sample of \( m \) points from the source domain \( \mathcal{S} = ((x_1,y_1),\ldots,(x_m,y_m)) \in (\mathcal{X} \times \mathcal{Y})^m \) with \( x_1,\ldots,x_m \) drawn i.i.d. according to \( Q \) and \( y_i = f_Q(x_i) \) for \( i \in [1,m] \); and an unlabeled sample \( \mathcal{T} = (x_1',\ldots,x_m') \in \mathcal{X}^n \) of size \( n \) drawn i.i.d. according to the target distribution \( P \). We denote by \( \hat{Q} \) the empirical distribution corresponding to \( x_1,\ldots,x_m \) and by \( \hat{P} \) the empirical distribution corresponding to \( \mathcal{T} \). We will be in fact more interested in the scenario commonly encountered in practice where, in addition to these two samples, a small amount of labeled data from the target domain \( \mathcal{T}' = ((x_1'',y_1''),\ldots,(x_m'',y_m'')) \in (\mathcal{X} \times \mathcal{Y})^s \) is received by the learner.

We consider a loss function \( L : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}_+ \) jointly convex in its two arguments. The \( L_p \) losses commonly used in regression and defined by \( L_p(y,y') = |y-y'|^p \) for \( p \geq 1 \) are special instances of this definition. For any two functions \( h, h' : \mathcal{X} \rightarrow \mathcal{Y} \) and any distribution \( D \) over \( \mathcal{X} \), we denote by \( L_D(h,h') \) the expected loss of \( h(x) \) and \( h'(x) \): \( L_D(h,h') = \mathbb{E}_{x \sim D}[L(h(x), h'(x))] \). The learning problem consists of selecting a hypothesis \( h \) out of a hypothesis set \( H \) with a small expected loss \( L_P(h,f_P) \) with respect to the target domain. We further extend this notation to arbitrary functions \( q : \mathcal{X} \rightarrow \mathbb{R} \) with a finite support as follows: \( L_q(h,h') = \sum_{x \in \mathcal{X}} q(x)L(h(x), h'(x)) \).

Finally, the notion of discrepancy \( \text{disc}(P,Q) \) originally introduced by Mansour et al. [2009] is given by

\[
\text{disc}(P,Q) = \max_{h,h' \in H} |L_P(h,h') - L_Q(h,h')|.
\]

3 Algorithm

In this section, we present our adaptation algorithm. We first review related previous work, next we introduce the key idea behind the algorithm and derive its general form, and finally formulate it as a convex optimization problem.

3.1 Previous work

One of the best theoretically motivated algorithms for domain adaptation is that of discrepancy minimization (DM) [Cortes and Mohri, 2013]. Given a positive semi-definite (PSD) kernel \( K \), the hypothesis returned by this algorithm is the solution of the following optimization problem

\[
\min_{h \in \mathcal{H}} \lambda \|h\|^2_K + L_{q_{\text{min}}}(h,f_Q),
\]

where \( \| \cdot \|_K \) is the norm on the reproducing Hilbert space \( \mathcal{H} \) induced by the kernel \( K \) and \( q_{\text{min}} \) is a distribution over the support of \( \hat{Q} \) such that \( q_{\text{min}} = \argmin_{q \in \mathcal{Q}} \text{disc}(q,\hat{P}) \), where \( \mathcal{Q} \) is the set of all distributions defined over the support of \( \hat{Q} \).

Observe that, by definition, the objective function optimized by \( q_{\text{min}} \) corresponds to a maximum over all pairs of hypotheses. But, the maximizing pair of hypotheses may not be among the candidates considered by the learning algorithm. Thus, a learning algorithm based on discrepancy minimization tends to be too conservative.
3.2 Main idea

From here on we assume the algorithm selected by the learner is an instance of a regularized risk minimization algorithm over the Hilbert space $\mathbb{H}$ induced by a PSD kernel $K$. With knowledge of the target labels, these algorithms return a hypothesis $h^*$ solution of $\min_{h \in \mathbb{H}} F(h)$ where

$$F(h) = \lambda \|h\|_K^2 + \mathcal{L}_\hat{p}(h, f_P),$$

and $\lambda \geq 0$ is a regularization parameter. Thus, $h^*$ can be viewed as the ideal hypothesis.

In view of that, we formulate our objective, in the presence of a domain adaptation problem, as that of finding a hypothesis $h$ whose loss $\hat{\mathcal{L}}_\hat{p}(h, f_P)$ with respect to the target domain is as close as possible to $\hat{\mathcal{L}}_\hat{p}(h^*, f_P)$. To do so, we will seek in fact a hypothesis $h$ that is as close as possible to $h^*$, which would imply the closeness of the losses with respect to the target domains. We do not have access to $f_P$ and can only access the labels of the training sample $S$. Thus, we must resort to using in our objective function, instead of $\mathcal{L}_\hat{p}(h, f_P)$, a reweighted empirical loss over the training sample $S$. The main idea behind our algorithm is to define, for any $h \in \mathbb{H}$, a reweighting function $Q_h : \mathcal{S}_X \rightarrow \mathbb{R}$ such that the objective function $G$ defined for all $h \in \mathbb{H}$ by

$$G(h) = \lambda \|h\|_K^2 + \mathcal{L}_{Q_h}(h, f_Q)$$

is uniformly close to $F$, thereby resulting in close minimizers. Since the first term of (2) and (3) coincide, the idea consists equivalently of seeking $Q_h$ such that $\mathcal{L}_{Q_h}(h, f_Q)$ and $\hat{\mathcal{L}}_\hat{p}(h, f_P)$ be as close as possible. Observe that this departs from the standard reweighting methods: instead of reweighting the training sample with some fixed set of weights, we allow the weights to vary as a function of the hypothesis $h$. Note that we have further relaxed the condition commonly adopted by reweighting techniques that the weights must be non-negative and sum to one. Allowing the weights to be in a richer space than the space of probabilities over $\mathcal{S}_X$ could raise over-fitting concerns but, we will later see that this in fact does not affect our learning guarantees and leads to good empirical results.

Of course, searching for $Q_h$ to directly minimize $|\mathcal{L}_{Q_h}(h, f_Q) - \mathcal{L}_\hat{p}(h, f_P)|$ is in general not possible since we do not have access to $f_P$, instead we consider a convex surrogate hypothesis set $H'' \subset H$ that could contain a good approximation of $f_P$. This leads us to the following definition of a hypothesis-dependent reweighting function:

$$Q_h = \arg\min_{q \in \mathcal{F}(\mathcal{S}_X, \mathbb{R})} \max_{h'' \in H''} |\mathcal{L}_q(h, f_Q) - \mathcal{L}_\hat{p}(h, h'')|.$$  

(4)

We now show how to express (3) as a convex function in the case where $H''$ is a convex set.

**Proposition 1.** For any $h \in \mathbb{H}$, let $Q_h$ be defined by (4). Then, the following identity holds for any $h \in \mathbb{H}$:

$$\mathcal{L}_{Q_h}(h, f_Q) = \frac{1}{2} \left( \max_{h'' \in H''} \mathcal{L}_\hat{p}(h, h'') + \min_{h'' \in H''} \mathcal{L}_\hat{p}(h, h'') \right).$$

In view of this proposition, with our choice of $Q_h$ based on (4), the objective function $G$ of our algorithm (3) can be equivalently written for all $h \in \mathbb{H}$ as follows:

$$G(h) = \lambda \|h\|_K^2 + \frac{1}{2} \left( \max_{h'' \in H''} \mathcal{L}_\hat{p}(h, h'') + \min_{h'' \in H''} \mathcal{L}_\hat{p}(h, h'') \right).$$  

(5)

It is not hard to see that since $\mathcal{L}_\hat{p}$ is a jointly convex function and $H''$ is a convex set, the above objective is in fact convex too. Of course, the performance of our algorithm depends on the choice of $H''$. This choice can be in fact motivated by the pointwise learning guarantees given in Appendix B. In particular, we can show the existence of $H''$ for which the learning bound for our algorithm is tighter than that of DM.

4 Experiments

We now present the results of evaluating the performance of our algorithm and comparing it with several others. GDM is compared to DM and to training on the source distribution. The following algorithms were also used as a baseline:
1. The KMM algorithm [Huang et al., 2006] reweights data samples to match empirical target and source means on the feature space induced by Gaussian kernels. The hyper-parameters of this algorithm were set to the recommended values of $B = 1000$ and $\epsilon = \sqrt{\frac{m}{m-1}}$.

2. KLIEP [Sugiyama et al., 2007] minimizes the KL-divergence between the source and target empirical distributions. Distributions are modeled as a mixture of Gaussians. The bandwidth of the kernel for both KLIEP and KMM was selected from the set \{\sigma d: \sigma = 2^{-5}, \ldots, 1\} via validation on the test set, where $d$ is the mean distance between points sampled from the source domain.

3. FE [Daumé III, 2007]. This algorithm maps source and target data to a common high-dimensional feature space where the difference of the distributions is hoped to be smaller.

We refrained from comparing against the two-stage algorithm of Bickel et al. [2007], as it was already shown to perform similarly to KMM and KLIEP [Cortes and Mohri, 2013].

The hypothesis set $H$ was given by linear functions. The learning algorithm used for all tasks was ridge regression and the performance evaluated by the mean squared error. The method for selecting the hyperparameters in our algorithm can be found in [Cortes et al., 2015]. In order to be thorough, we also report the results of training only on the small amount of target data. To make a fair comparison, we allowed the use of the small amount of labeled data to all other baselines. To do so, we simply added this data to the training set and ran the algorithms on the extended source data.

For our experiment, we considered the cross-domain sentiment analysis data set of Blitzer et al. [2007b]. This data set consists of consumer reviews from 4 different domains: books, kitchen, electronics and dvds. We used the top 1,000 unigrams and bigrams as features. For each pair of adaptation tasks we sampled 700 points from the source distribution and 700 unlabeled points from the target. Only 50 labeled points from the target distribution were used to tune $r$. The final evaluation was done on a test set of 1,000 points. Figure 1 shows normalized MSE of all algorithms when adapting from electronics to all others.

We provide a more comprehensive list of experiments in Appendix C. The result of these experiments show that not only does our algorithm consistently outperform all other baselines, but in several tasks its performance is in fact really close to that of training on the target distribution.

5 Conclusion

We presented a new theoretically well-founded domain adaptation algorithm seeking to minimize a less conservative quantity than the DM algorithm. We showed that our algorithm can always guarantee a better performance than the DM algorithm. Furthermore, we derived a convex formulation for our algorithm and empirically showed that it outperforms other common baselines for domain adaptation.
References


A Proof of Proposition 1

Proposition 1. For any $h \in \mathcal{H}$, let $Q_h$ be defined by (4). Then, the following identity holds for any $h \in \mathcal{H}$:

$$\mathcal{L}_{Q_h}(h, f_Q) = \frac{1}{2} \left( \max_{h'' \in \mathcal{H}''} \mathcal{L}_{\hat{P}}(h, h'') + \min_{h'' \in \mathcal{H}''} \mathcal{L}_{\hat{P}}(h, h'') \right).$$

Proof. For any $h \in \mathcal{H}$, the equation $\mathcal{L}_q(h, f_Q) = l$ with $l \in \mathbb{R}$ admits a solution $q \in \mathcal{F}(\mathcal{S}_X, \mathbb{R})$. Thus, for any $h \in \mathcal{H}$, we can write

$$\mathcal{L}_{Q_h}(h, f_Q) \arg \min_{l \in \mathcal{L}_q(h, f_Q) : q \in \mathcal{F}(\mathcal{S}_X, \mathbb{R})} \max_{h'' \in \mathcal{H}''} |l - \mathcal{L}_{\hat{P}}(h, h'')|$$

$$= \arg \min_{l \in \mathbb{R}} \max_{h'' \in \mathcal{H}''} |l - \mathcal{L}_{\hat{P}}(h, h'')|$$

$$= \arg \min_{l \in \mathbb{R}} \max_{h'' \in \mathcal{H}''} \left\{ \mathcal{L}_{\hat{P}}(h, h'') - l, l - \mathcal{L}_{\hat{P}}(h, h'') \right\}$$

$$= \arg \min_{l \in \mathbb{R}} \left\{ \max_{h'' \in \mathcal{H}''} \mathcal{L}_{\hat{P}}(h, h'') - l, l - \min_{h'' \in \mathcal{H}''} \mathcal{L}_{\hat{P}}(h, h'') \right\}$$

$$= \frac{1}{2} \left( \max_{h'' \in \mathcal{H}''} \mathcal{L}_{\hat{P}}(h, h'') + \min_{h'' \in \mathcal{H}''} \mathcal{L}_{\hat{P}}(h, h'') \right),$$

since the minimizing $l$ is obtained for

$$\max_{h'' \in \mathcal{H}''} \mathcal{L}_{\hat{P}}(h, h'') - l = l - \min_{h'' \in \mathcal{H}''} \mathcal{L}_{\hat{P}}(h, h'').$$

B Learning Guarantees

Here, we provide learning guarantees for our algorithm, which in turn will help us select the set $\mathcal{H}''$ defining our algorithm. As in previous work, we assume that the loss function $L$ is $\mu$-admissible: there exists $\mu > 0$ such that

$$|L(h(x), y) - L(h'(x), y)| \leq \mu |h(x) - h'(x)|$$

holds for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$ and $h, h' \in \mathcal{H}$, a condition that is somewhat weaker than $\mu$-Lipschitzness with respect to the first argument. The $L_\mu$ losses commonly used in regression, $p \geq 1$, verify this condition [Cortes and Mohri, 2013].

B.1 Generalization bounds

The existing pointwise guarantees for the DM algorithm are directly derived from a bound on the norm of the difference of the ideal function $h^*$ and the hypothesis obtained after reweighting the sample losses using a distribution $q$. The bound is expressed in terms of the discrepancy and a term $\eta_H(f_P, f_Q)$ measuring the difference of the source and target labeling functions defined by

$$\eta_H(f_P, f_Q) = \min_{h \in \mathcal{H}} \left( \max_{x \in \text{supp}(\hat{P})} |f_P(x) - h_0(x)| + \max_{x \in \text{supp}(\hat{Q})} |f_Q(x) - h_0(x)| \right),$$

and is given by the following theorem.

Theorem 1 ([Cortes and Mohri, 2013]). Let $q$ be an arbitrary distribution over $\mathcal{S}_X$ and let $h^*$ and $h_q$ be the hypotheses minimizing $\lambda \|h\|_K^2 + \mathcal{L}_{\hat{P}}(h, f_P)$ and $\lambda \|h\|_K^2 + \mathcal{L}_q(h, f_Q)$ respectively. Then, the following inequality holds:

$$\lambda \|h^* - h_q\|_K^2 \leq \mu \eta_H(f_P, f_Q) + \text{disc}(\hat{P}, q).$$

The DM algorithm is defined by selecting the distribution $q$ minimizing the right-hand side of the bound (7), that is $\text{disc}(\hat{P}, q)$.

We will now show a result of the same nature for our hypothesis-dependent reweighting $Q_h$ by showing that its choice also coincides with that of minimizing an upper bound on $\lambda \|h^* - h''\|_K^2$. Let $\mathcal{A}(H)$ be the set of all functions $U : h \mapsto U_h$ mapping $H$ to $\mathcal{F}(\mathcal{S}_X, \mathbb{R})$ such that for all $h \in H$, $h \mapsto \mathcal{L}_{Q_h}(h, f_Q)$ is a convex function. $\mathcal{A}(H)$ contains all constant functions $U$ such that $U_h = q$ for all $h \in H$, where $q$ is a distribution over $\mathcal{S}_X$. By Proposition 1, $\mathcal{A}(H)$ also includes the function $Q : h \mapsto Q_h$ used by our algorithm.
**Definition 1** (generalized discrepancy). For any \( U \in \mathcal{A}(H) \), we define the generalized discrepancy between \( \hat{P} \) and \( U \) as the quantity \( \text{DISC}(\hat{P}, U) \) given by

\[
\text{DISC}(\hat{P}, U) = \max_{h \in H, h' \in H'} |\mathcal{L}_{\hat{P}}(h, h') - \mathcal{L}_{U}(h, f_{Q})|.
\]  

(8)

We also denote by \( d_{\infty}^{\hat{P}}(f_P, H'') \) the following distance of \( f_P \) to \( H'' \) over the support of \( \hat{P} \):

\[
d_{\infty}^{\hat{P}}(f_P, H'') = \min_{h \in H''} \max_{x \in \text{supp}(\hat{P})} |h_0(x) - f_P(x)|.
\]  

(9)

The following theorem gives an upper bound on the norm of the difference of the minimizing hypotheses in terms of the generalized discrepancy and \( d_{\infty}^{\hat{P}}(f_P, H'') \).

**Theorem 2.** Let \( U \) be an arbitrary element of \( \mathcal{A}(H) \) and let \( h^{*} \) and \( h_{U} \) be the hypotheses minimizing \( \lambda||h||_{K}^{2} + \mathcal{L}_{\hat{P}}(h, f_{P}) \) and \( \lambda||h||_{K}^{2} + \mathcal{L}_{U}(h, f_{Q}) \) respectively. Then, the following inequality holds for any convex set \( H'' \subseteq H' \):

\[
\lambda||h^{*} - h_{U}||_{K}^{2} \leq \mu d_{\infty}^{\hat{P}}(f_P, H'') + \text{DISC}(\hat{P}, U).
\]  

(10)

A simple calculation shows that our choice of \( Q_{r} \) in fact minimizes the right hand side of (10). We now show that for a particular choice of \( H'' \) the bound given in (10) is tighter than the bound given in (7), which would imply better guarantees for our algorithm. Let

\[ \mathcal{H} = \{ B(r) : r \geq 0 \}, \]

where \( B(r) = \{ h'' \in H | \mathcal{L}_{Q}(h'', f_{Q}) \leq r^{p} \} \).

**Theorem 3.** Let \( L \) be an \( L_{p} \) loss. There exists \( H'' \in \mathcal{H} \) such that the following holds:

\[
\mu d_{\infty}^{\hat{P}}(f_P, H'') + \max_{h \in H, h' \in H''} |\mathcal{L}_{\hat{P}}(h, h') - \mathcal{L}_{Q}(h, f_{Q})| \leq \mu \eta_H(f_P, f_{Q}) + \text{disc}(\hat{P}, q).
\]

Proof. Let \( h_{0}^{*} \) be the minimizer in the definition of \( \eta_H(f_P, f_{Q}) \):

\[
h_{0}^{*} = \arg\min_{h_{0} \in H} \left( \max_{x \in \text{supp}(\hat{P})} |f_P(x) - h_{0}(x)| + \max_{x \in \text{supp}(Q)} |f_Q(x) - h_{0}(x)| \right),
\]

and let \( r = \max_{x \in \text{supp}(Q)} |f_Q(x) - h_{0}^{*}(x)| \). Let \( q \) be a distribution over \( \mathcal{S}_X \) and choose \( H'' \in \mathcal{H} \) as

\[
H'' = \{ h'' \in H | \mathcal{L}_{Q}(h'', f_{Q}) \leq r^{p} \}. \tag{9}
\]

Then, \( h_{0}^{*} \) is in \( H'' \) since

\[
\mathcal{L}_{Q}(h_{0}^{*}, f_{Q}) = \mathbb{E}_{x \sim q} \left[ |h_{0}^{*}(x) - f_{Q}(x)|^{p} \right] \leq \max_{x \in \text{supp}(Q)} |h_{0}^{*}(x) - f_{Q}(x)|^{p} = r^{p}.
\]

For the \( L_{p} \) loss, it is not hard to show [Cortes et al., 2014][Lemma 14] that for all \( h, h'' \in H \),

\[
|\mathcal{L}_{Q}(h, h'') - \mathcal{L}_{Q}(h, f_{Q})| \leq \mu |\mathcal{L}_{Q}(h', f_{Q})|^{\frac{1}{p}}. \tag{7}
\]

In view of this inequality, we can write:

\[
\max_{h \in H, h'' \in H''} |\mathcal{L}_{\hat{P}}(h, h'') - \mathcal{L}_{Q}(h, f_{Q})| \leq \max_{h \in H, h'' \in H''} |\mathcal{L}_{\hat{P}}(h, h'') - \mathcal{L}_{\hat{P}}(h, h'')| + \max_{h \in H, h'' \in H''} |\mathcal{L}_{\hat{P}}(h, h'') - \mathcal{L}_{\hat{P}}(h, f_{Q})| + \mu |\mathcal{L}_{Q}(h', f_{Q})|^{\frac{1}{p}}
\]

\[
\leq \text{disc}(\hat{P}, q) + \max_{h'' \in H''} |\mathcal{L}_{\hat{P}}(h''', f_{Q})|^{\frac{1}{p}}
\]

\[
\leq \text{disc}(\hat{P}, q) + \mu r
\]

\[
= \text{disc}(\hat{P}, q) + \mu \max_{x \in \text{supp}(Q)} |f_Q(x) - h_{0}^{*}(x)|.
\]

Using this inequality and the fact that \( h_{0}^{*} \in H'', \) we can write

\[
\mu d_{\infty}^{\hat{P}}(f_P, H'') + \max_{h \in H, h'' \in H''} |\mathcal{L}_{\hat{P}}(h, h'') - \mathcal{L}_{Q}(h, f_{Q})|
\]

\[
\leq \mu \min_{h'' \in H''} \max_{x \in \text{supp}(\hat{P})} |f_P(x) - h_{0}(x)| + \text{disc}(\hat{P}, q) + \mu \max_{x \in \text{supp}(Q)} |f_Q(x) - h_{0}^{*}(x)|
\]

\[
\leq \mu \left( \max_{x \in \text{supp}(\hat{P})} |f_P(x) - h_{0}(x)| + \max_{x \in \text{supp}(Q)} |f_Q(x) - h_{0}^{*}(x)| \right) + \text{disc}(\hat{P}, q)
\]

\[
= \mu \min_{h'' \in H''} \left( \max_{x \in \text{supp}(\hat{P})} |f_P(x) - h_{0}(x)| + \max_{x \in \text{supp}(Q)} |f_Q(x) - h_{0}(x)| \right) + \text{disc}(\hat{P}, q)
\]

\[
= \mu \eta_H(f_P, f_{Q}) + \text{disc}(\hat{P}, q),
\]

which concludes the proof. \( \square \)
Figure 2: (a) Hypotheses obtained by training on source (green circles), target (red triangles) and using DM (dashed blue) and GDM algorithms (solid blue). (b) Objective functions for source and target distribution as well as GDM and DM algorithms. Sets $H$ and surrogate hypothesis set $H'' \subseteq H$ are shown at the bottom. The vertical lines represent the minimizing hypothesis for each loss.

C Experiments

In this section, provide further empirical evaluation of our algorithm against other baselines with favorable results for GDM.

C.1 Synthetic data set

To give an empirical comparison of the GDM and DM algorithms, we adopted the following setup inspired by Huang et al. [2006]: we sampled source distribution examples from the uniform distribution over the interval $[2, 1]$ and target samples from the uniform distribution over $[0, 25]$. The labels were given by the map $x \mapsto -x + x^3 + \xi$ where $\xi$ is a Gaussian random variable with mean 0 and standard deviation 0.1 and our hypothesis set was chosen to be that of linear functions.

Figure 2(a) shows the regression hypotheses obtained by training the DM and GDM algorithms as well as those obtained by training on the source and the target distributions. Notice how the GDM solution approaches the ideal solution better than DM. These results can be better explained by Figure 2(b) which plots the objective function minimized by each algorithm as a function of the slope $w$ of the linear function, the only variable of the hypothesis. Vertical lines show the value of the minimizing hypothesis for each loss. Keeping in mind that the regularization parameter $\lambda$ used in ridge regression corresponds to a Lagrange multiplier for the constraint $w^2 \leq \Lambda^2$ for some $\Lambda$ [Cortes and Mohri, 2013][Lemma 1], the hypothesis set $H = \{w: |w| \leq \Lambda\}$ is shown at the bottom of this plot. The shaded region represents the set $H'' = H \cap \{h'' | L_{\text{min}}(h'') \leq r\}$. It is clear from this plot that DM helps approximate the target loss function. Nevertheless, only GDM seems to uniformly approach it. This should come as no surprise since our algorithm was designed precisely for this purpose.

C.2 Adaptation Data Sets

Our first task is given by the 4 kin-8xy Delve data sets [Rasmussen et al., 1996]. These data sets are variations of the same model: a realistic simulation of the forward dynamics of an 8-link all-revolute robot arm. The task in all data sets consists of predicting the distance of the end-effector from a target. Data sets differ by the degree of non-linearity (fairly linear, $x = f$, or non-linear, $x = n$) and the amount of noise in the output (moderate, $y = m$, or high, $y = h$). A sample of 200 points from each domain was used for training and 10 labeled points from the target distribution were used to select $H''$. The experiment was carried out 10 times. The results of testing on a sample of 400 points from the target domain are reported in Figure 3(a). The bars represent the median performance of each algorithm and error bars show the inter-quartile range. All results were normalized in such a way that training on the source had error constantly equal to 1. Notice that the performance of all algorithms is comparable when adapting to kin8-fm since both labeling functions are fairly linear, yet only GDM is able to significantly approach the performance on training on target for all three tasks. In order to better understand the advantages of GDM over DM we plot the relative error of
Figure 3: (a) Normalized MSE performance for different adaptation algorithms when adapting from $k_{8fh}$ to the three other $k_{8xy}$ domains. Small denotes training on small labeled target sample. (b) Relative error of DM over GDM as a function of the ratio $r/\Lambda$.

Figure 4: Normalized MSE of different algorithms adapting from the caltech256 dataset to all other datasets.

DM against GDM as a function of the ratio $r/\Lambda$ in Figure 3(b). Notice that when the ratio $r/\Lambda$ is small, both algorithms behave similarly which typically for the adaptation task $fh$ to $fm$. On the other hand, a better performance of GDM can be obtained when the ratio is larger. This can be interpreted as follows: a small ratio means that the size of $H''$ is small and the hypothesis returned by GDM will be close to that of DM, while for $H''$ large, GDM can find a better hypothesis.

Finally, we considered a key domain adaptation task in the computer vision community [Tommasi et al., 2014] where the domains correspond to 4 well known collections of images: bing, caltech256, sun and imagenet. These data sets have been standardized so that they all share the same feature representation and labeling function [Tommasi et al., 2014]. We used the data from the first 5 shared classes and sampled 800 labeled points from the source distribution and 800 unlabeled points from the target distribution, as well as 50 labeled target points used as validation to determine $r$. The results of testing on 1,000 points from the target domain are shown in Figure 4 where we trained on caltech256. The results of this section confirm that GDM consistently outperforms all other baselines in some cases by a rather large margin.