Distance Metric Learning under Covariate Shift

Bin Cao[1], **Xiaochuan Ni**[2], **Jian-Tao Sun**[2], **Gang Wang**[2], **Qiang Yang**[1]
[1]HKUST, Clear Water Bay, Hong Kong
[2]Microsoft Research Asia, Beijing
{caobin, qyang}@cse.ust.hk, {xini, jtsun, gawa}@microsoft.com

Abstract

Learning distance metrics is a fundamental problem in machine learning. Previous distance-metric learning research assumes that the training and test data are drawn from the same distribution, which may be violated in practical applications. When the distributions differ, a situation referred to as *covariate shift*, the metric learned from training data may not work well on the test data. In this case the metric is said to be inconsistent. In this paper, we address this problem by proposing a novel metric learning framework known as *consistent distance metric learning* (CDML), which solves the problem under covariate shift situations. We theoretically analyze the conditions when the metrics learned under covariate shift are consistent. Based on the analysis, a convex optimization problem is proposed to deal with the CDML problem. An importance sampling method is proposed for metric learning and two importance weighting strategies are proposed and compared in this work. Experiments are carried out on synthetic and real world datasets to show the effectiveness of the proposed method.

1 Introduction

Distance Metric learning (DML) is an important problem in many machine learning problems, such as classification through nearest neighborhood methods [Sriperumbudur and Lanckriet, 2007], clustering [Davis *et al.*, 2007] and semi-supervised learning [Yeung and Chang, 2007], etc. DML aims at learning a distance metric for an input space from some additional information such as must-link/cannot-link constraints between data instances. In the case of classification, for example, the class label information can be converted to such constraints, where data instances with the same label can be used to construct must-link pairs and those from different classes can be used to form cannot-link pairs. The key intuition in DML is to find a distance metric that can pull must-link instance pairs close to each other while push the cannot-link pairs away from each other.

Although different DML algorithms have been proposed [Xing *et al.*, 2003; Yeung and Chang, 2007; Davis *et al.*, 2007], previous research works generally assumed that the training and test data are drawn from the same distribution. If this assumption holds, the metric learned from the training data will work well on the test data. However, in many real world applications, it may be inappropriate to make this assumption. For example, in some cases, the training data may be collected with a sampling bias [Zadrozny, 2004], and in other cases, the data distribution may change due to the changing environment [Pan *et al.*, 2008]. In these situations, the distance metric learned on training data cannot be directly utilized on the test data. Figure 1 shows a synthetic data example: for the training data, the first dimension (x-axis) has more discriminative information. However, for the test data, the second dimension (y-axis) contains more discriminative information. Thus, the metric that keeps the must-link training instances close to each other does not necessarily keep them to be close after the distribution changes. We refer to such discrepancies as the *inconsistency problem* in metric learning.

The problem of learning when the training and test data have different distributions has been studied from different perspectives, e.g., as covariate shift [Bickel *et al.*, 2007], sample selection bias [Zadrozny, 2004] or domain adaptation [Daume and Marcu, 2006]. Previous related research in transfer learning mainly focused on supervised learning, including classification and regression. To the best of our knowledge, there has been no work that addresses metric learning when the training and test data have different distributions. In fact, the concept of distances is closely related to data distribution. It is well known that the Euclidean distance is linked to Gaussian distribution and the Manhattan distance measure (1-norm distance) is associated with Laplace distribution. Therefore, any change of distribution will cause problems for metric learning, invalidating many previous results.

In this paper, we concentrate on the problem of consistent distance metric learning under covariate shift. The assumption made under covariate shift in classification problems is that although the data distributions can change, the conditional distributions of labels given features keep stable. For metric learning we make a similar assumption that the conditional distributions of the indicator variable of must-link/cannot-link keep stable. We theoretically analyze the consistency property of distance metric learning when training and test data follow different distributions. Based on our

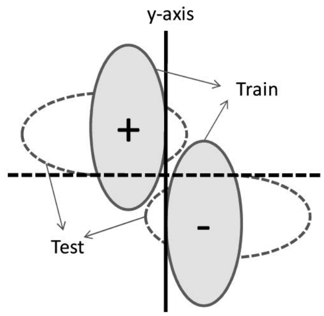

Figure 1: On the training set, the x-axis is crucial to distinguish the two classes. However, on the test set the y-axis is more important instead. The metric learned from the training set will have bias when generalized to test set.

analysis, we propose a convex optimization problem for consistent distance metric learning (CDML). We novelly adapt importance sampling methods used in supervised learning for metric learning. Two importance weighting strategies are investigated. The first strategy estimates the importance weights for data instances before using them to calculate importance weights for instance pairs. The second strategy directly estimates the importance weights for instance pairs. The two strategies are compared and analyzed from both theoretical and practical perspectives. We conduct empirical experimentation with both synthetic and real world datasets to demonstrate the effectiveness of the proposed algorithm.

2 Distance Metric Learning

Distance Metric learning (DML) aims to learn a distance metric from a given collection of must-link/cannot-link instance pairs while preserving the distance relation among the training data instances. In general, DML can be formulated as the following multi-objective optimization problem:

$$\begin{cases} \min . \mathbb{E}_{x_i,x_j\sim\Pr(x)}\left[d_\theta(x_i,x_j)\right], & (i,j)\in\mathcal{S} \\ \max . \mathbb{E}_{x_i,x_j\sim\Pr(x)}\left[d_\theta(x_i,x_j)\right], & (i,j)\in\mathcal{D} \end{cases} \tag{1}$$

where x_i and x_j are instances drawn from distribution $\Pr(x)$ and $d_\theta(x_i, x_j)$ is the distance function for x_i and x_j with the parameter θ. $\mathcal{S}$ is the set of must-link pairs and $\mathcal{D}$ is the set of cannot-link pairs. In most distance metric learning algorithms, the distance function type is restricted to the Malahanobis distance, which can be defined as

$$d_A(x,y) = \sqrt{(x-y)^{\mathrm{T}}A(x-y)}. \tag{2}$$

where A is a parameter in the distance function that is a positive semi-definite matrix. Since we can always factorize A into $A = L^{\mathrm{T}}L$, the Malahanobis distance in the original feature space can be regarded as the Euclidean distance in a new feature space after a linear transformation is applied to the original feature space, as shown below,

$$d_A(x,y) = \sqrt{(x-y)^{\mathrm{T}}A(x-y)} = \sqrt{(Lx-Ly)^{\mathrm{T}}(Lx-Ly)}. \tag{3}$$

To be precise, this is a pseudo metric rather than a metric from a mathematical perspective. However, in this paper we still follow the terminologies of previous work.

The first objective in Equation 1 is to minimize the expected distance of must-link pairs. The second objective is to maximize the distance of cannot-link pairs. When Malahanobis distance is used, the square of the distance is usually considered instead of the distance function (Equation 2), in order to simplify the formulation.

In order to solve the multi-objective optimization problem, the objective functions can be converted to a single-objective optimization problem. One conversion strategy is to use one objective as the optimization goal and the other as constraints, as done in [Xing *et al.*, 2003]. A second strategy is to regard both objectives as constraints and optimize a new objective function as in [Davis *et al.*, 2007]. The constraints for must-link relations can either be hard or soft. For hard constraints, we have

$$d_A^2(x_i,x_j) < u, \tag{4}$$

and for the soft constraints, we have

$$d_A^2(x_i,x_j) < u + \xi \tag{5}$$

where u is the upper bound for the distance between must-link pairs and ξ is a slack variable. Similarly, for cannot-link pair we can obtain the hard constraints.

$$d_A^2(x_i,x_j) > l \tag{6}$$

and the soft ones

$$d_A^2(x_i,x_j) > l - \xi, \tag{7}$$

where l is a lower bound for the distance of cannot-link pairs.

A more general formulation of DML is to consider a loss function over the distance function. We can show that different strategies mentioned above can be regarded as using different loss functions. In fact, we can define the loss function $\mathcal{L}(s_{ij}, d_A(x_i, x_j))$ similar to hinge loss, where s_{ij} is a label set to 1 for must link pairs and -1 for cannot-link pairs. For must-link pairs,

$$\mathcal{L}(1, d_A(x_i,x_j)) = \begin{cases} 0, & d_A^2(x_i,x_j) < u \quad (8a) \\ d_A^2(x_i,x_j) - u, & d_A^2(x_i,x_j) \geq u \quad (8b) \end{cases}$$

and for cannot-link pairs,

$$\mathcal{L}(-1, d_A(x_i,x_j)) = \begin{cases} 0, & d_A^2(x_i,x_j) > l \quad (9a) \\ l - d_A^2(x_i,x_j), & d_A^2(x_i,x_j) \leq l \quad (9b) \end{cases}$$

Then we can formulate the DML problem as,

Problem 1.

$$\min \ell(x;A) \equiv \mathbb{E}_{x_i,x_j\sim\Pr(x)}[\mathcal{L}(s_{ij}, d_A(x_i,x_j))], \tag{10}$$

Similar to supervised learning, it is infeasible to optimize the expected loss in Problem 1 in practical problems. We need to solve the following problem where the empirical loss is minimized.

Problem 2.

$$\min \ell_{emp}(x;A) \equiv \frac{1}{N}\sum_{i,j} \mathcal{L}(s_{ij}, d_A(x_i,x_j)), \tag{11}$$

where N is the number of pairs considered.

This would introduce the problem of generalization ability, which will be discussed in the next section.

3 Consistency in Metric Learning

The above formulation of DML has an implicit assumption to guarantee generalization ability: the distributions of training and test data are identical. However, when the training and test data are drawn from different distributions, the metric that keeps the must-link pairs of training instances close to each other may not keep the ones in the test data close due to the changes in data distributions. In this section, we will first define consistency in distance metric learning problem. Then we will investigate how to handle this problem when covariate shift exists.

Definition 1. *A metric learning algorithm is consistent if*

$$\lim_{N\to\infty} \ell_{emp}(x; A) = l(x; A), \;\; x \sim \Pr(x) \tag{12}$$

where N is the number of training data.

A consistent metric-learning method will guarantee the ability of generalizing the metric learned on a finite set of training data to the test data. Different from the consistency problem defined for pattern recognition problems, metric learning is defined on instance pairs rather than instances. In other words, it considers the relations between instances. Therefore, the consistency of metric learning is about the generalization ability of relation between data instances.

In the case of covariate shift, the distributions of training and test data are different. Thus, we have

$$x_{train} \sim \Pr(x), \quad x_{test} \sim \Pr'(x) \tag{13}$$

Accordingly, general DML algorithms will fail to satisfy the consistency property.

In a supervised learning setting, the covariate shift problem can be solved by importance sampling methods [Shimodaira, 2000]. Based on Theorem 1, presented as follows, we can also adapt importance sampling methods to the distance metric learning problem.

Theorem 1. *Suppose that x_i and x_j are drawn independently from $\Pr(x)$. If a metric learning algorithm for Problem 2 is consistent without covariate shift, then minimizing the following function (Equation 14) using this algorithm can produce consistent solutions under covariate shift.*

$$\min \ell'_{emp}(x; A) = \min \sum_{i,j} w_{ij}\mathcal{L}(s_{ij}, d_A(x_i, x_j)) \tag{14}$$

where $w_{ij} = \frac{\Pr'(x_i)\Pr'(x_j)}{\Pr(x_i)\Pr(x_j)}$

Proof. Since the metric learning algorithm is consistent without covariate shift, we have

$$\lim_{N\to\infty} \ell'_{emp}(x; A) = \mathbb{E}_{x_i,x_j\sim\Pr(x)}[w_{ij}\mathcal{L}(s_{ij}, d_A(x_i, x_j))] \tag{15}$$

Let us denote the right hand side of the above equation by $\ell'(x; A)$.

$$\begin{aligned}\ell'(x; A) &= \mathbb{E}_{x_i,x_j\sim\Pr(x)}[w_{ij}\mathcal{L}(s_{ij}, d_A(x_i, x_j))] \\ &= \int w_{ij}\mathcal{L}(s_{ij}, d_A(x_i, x_j))\Pr(x_i)\Pr(x_j)\mathrm{d}x_i\mathrm{d}x_j \\ &= \int \mathcal{L}(s_{ij}, d_A(x_i, x_j))\Pr'(x_i)\Pr'(x_j)\mathrm{d}x_i\mathrm{d}x_j \\ &= \mathbb{E}_{x_i,x_j\sim\Pr'(x)}[\mathcal{L}(s_{ij}, d_A(x_i, x_j))]\end{aligned}$$

This shows the conclusion holds. □

In supervised learning with covariate shift, $\Pr'(x)/\Pr(x)$ is called the importance weight. The weight w_{ij} is the product of importance weights for x_i and x_j. Therefore, this approach needs to estimate the importance weight for x first. Since the loss function is defined over pairs of instances, it is possible to directly estimate the importance weight on instance pairs in some cases. For example, it is possible when the distance is induced by a norm, which indicates $d(x_i, x_j) = f(x_i - x_j)$. The Malahanobis distance also belongs to such a case. From this perspective, we can calculate the importance weight for instance pairs using

$$w_{ij} = \frac{\Pr'(x_i - x_j)}{\Pr(x_i - x_j)} = \frac{\Pr'(\Delta x)}{\Pr(\Delta x)} \tag{16}$$

where $\Delta x = x_i - x_j$. We need to introduce the concept of cross-correlation in this case. If X and Y are two independent random variables with probability distributions g and h, respectively, then the probability distribution of the difference $X - Y$ is given by the cross-correlation $g \star h$,

$$(g \star h)(t) = \int_{-\infty}^{\infty} g^*(\tau)h(t+\tau)\mathrm{d}\tau \tag{17}$$

where g^* denotes the complex conjugate of g. Then, we can have another formulation of CDML solution directly defined using Δx.

Theorem 2. *If the distance is introduced by a norm, then $d_\theta(x_i, x_j) = f(\Delta x)$. Suppose that δ_x is a random variable drawn from $Q(x) = \Pr(x) \star \Pr(x)$, and that a metric learning algorithm for Problem 2 is consistent without covariate shift, minimizing the following problem (equation 18) using this algorithm can produce consistent solutions under covariate shift.*

$$\min \ell'_{emp}(x; A) = \min \sum_{i,j} w_{ij}\mathcal{L}(s_{ij}, f(\delta_x)) \tag{18}$$

where $w_{ij} = \Pr'(\delta_x)/\Pr(\delta_x)$.

The proof is similar with the one in Theorem 1 except the two variable now changed to one variable δ_x and the details are omitted here. From this point of view, the problem is treated as a supervised learning problem where instances are pairs. Although both approaches can be used to solve the CDML problem, they are not equivalent, with each having its own advantages and drawbacks. We will discuss and compare the two weighting approaches later.

In our analysis, we have the prerequisite that the original distance metric learning itself is consistent without covariate shift. This condition, although nontrivial to address, is beyond the scope of this paper. We believe that this condition can be addressed similarly to what was done in pattern recognition [Vapnik, 1995].

4 Metric Learning Under Covariate Shift

As we analyzed in the above section, the corresponding consistent metric learning problem can be formulated as

Problem 3.

$$\min \sum_{i,j} w(x_i, x_j)\mathcal{L}(s_{ij}, d_A(x_i, x_j)), \;\; x_i, x_j \sim \Pr(x) \tag{19}$$

It can also be regarded as a cost-sensitive distance metric learning problem, where violating different pair constraints could introduce different costs. In the following, we formulate it as a convex optimization problem.

4.1 Our Approach

We have so far proposed a general framework for consistent distance metric learning. In this section, we will propose a specific convex optimization problem to solve the CDML.

Given the loss function defined in previous section, the cost sensitive learning problem then becomes,

$$\begin{aligned}\min_{A\succeq 0} \quad & \sum w_{ij}\xi_{ij} \\ s.t. \quad & \mathrm{tr}(A(x_i-x_j)(x_i-x_j)^{\mathrm{T}}) \le u+\xi_{ij} \quad (i,j)\in\mathcal{S}, \\ & \mathrm{tr}(A(x_i-x_j)(x_i-x_j)^{\mathrm{T}}) \ge l-\xi_{ij} \quad (i,j)\in\mathcal{D}, \\ & \xi_{ij}\ge 0\end{aligned} \tag{20}$$

where $\mathrm{tr}(M)$ is the trace of the matrix M.

It is easy to show the above problem is a convex optimization problem. More specifically, the problem can be converted to the following semi-definite problem (SDP), where general SDP solvers can be applied to.

$$\begin{aligned}\min_{A\succeq 0} \; & \mathrm{tr}(C\tilde{X}), \\ s.t. \; & \mathrm{tr}(P_{ij}\tilde{X}) = u \quad (i,j)\in\mathcal{S}, \\ & \mathrm{tr}(P_{ij}\tilde{X}) = l \quad (i,j)\in\mathcal{D}\end{aligned} \tag{21}$$

where

$$C=\begin{pmatrix}\mathbf{0} & \mathbf{0}\\ \mathbf{0} & W\end{pmatrix}, \quad \tilde{X}=\begin{pmatrix}A & \mathbf{0}\\ \mathbf{0} & \Lambda\end{pmatrix}, \quad P_{ij}=\begin{pmatrix}B & \mathbf{0}\\ \mathbf{0} & E_{ij}\end{pmatrix}, \tag{22}$$

and $W=\mathrm{diag}(w_{ij}), \Lambda=\mathrm{diag}(\xi_{ij}), B=(x_i-x_j)(x_i-x_j)^{\mathrm{T}}$. For $(i,j)\in\mathcal{S}$, E_{ij} has only one nonzero element with $E_{ij}(i,j)=-1$; For $(i,j)\in\mathcal{D}$, E_{ij} has only one nonzero element with $E_{ij}(i,j)=1$.

In this paper we use a general SDP solver (SDPT3[1]) to solve our problem. As shown in [Weinberger and Saul, 2008], it is possible to investigate the structure in the pairwise relationship to reduce the complexity. We plan to develop algorithms that are more efficient for this problem in the future.

If we have some prior knowledge over the distribution and metrics, we can include one regularization term into Equation 20. In [Kulis *et al.*, 2006], Kulis has shown the connection between KL-divergence and Bregman Matrix Divergence. The Burg matrix divergence can be introduced as a regularization term

$$\begin{aligned}\min_{A\succeq 0} \quad & \sum w_{ij}\xi_{ij} + \gamma D_{Burg}(A,A_0) \\ s.t. \quad & \mathrm{tr}(A(x_i-x_j)(x_i-x_j)^{\mathrm{T}}) \le u+\xi_{ij} \quad (i,j)\in\mathcal{S} \\ & \mathrm{tr}(A(x_i-x_j)(x_i-x_j)^{\mathrm{T}}) \ge l-\xi_{ij} \quad (i,j)\in\mathcal{D} \\ & \xi_{ij}\ge 0\end{aligned} \tag{23}$$

where $D_{Burg}(A,A_0)=(\mathrm{tr}(AA_0^{-1})-\log\det(AA_0^{-1}))-n$. We let $A_0=I$ if we do not have any specific prior information. The problem is still a convex optimization problem. We adopt the successive approximation method used in [Grant and Boyd., 2009] to solve this convex optimization problem.

Our approach is related to information-theoretic metric learning (ITML) [Davis *et al.*, 2007] if we do not consider the importance weights. However, the loss function used in ITML only minimized the divergence between the learned matrix and its prior. As such the loss function in ITML cannot be directly applied to cost sensitive learning. For this reason, hinge loss is introduced in our method.

4.2 Estimating Importance Weights

Accurately estimating the importance weights is crucial in covariate shift. In this paper, we use a state-of-the-art algorithm proposed by Tsuboi in [Tsuboi *et al.*, 2008] for this estimation. We will briefly introduce the algorithm in this section.

Since estimating the distribution of x is not trivial when the data dimension is high, directly estimating the importance weight $\frac{\Pr'(x)}{\Pr(x)}$ is a preferable approach [Tsuboi *et al.*, 2008]. Let $w(x)=\sum_k \alpha_k\varphi_k(x)$, where α_k are parameters to be learned from data samples and $\{\varphi_k(x)\}$ are the basis functions such that $\varphi_k(x)\ge 0$ for all x. The importance weight is obtained by minimizing $\mathrm{KL}(\Pr_{test}(x)||\Pr'_{test}(x))$, where $\Pr'_{test}(x)=w(x)\Pr_{train}(x)$. It can be further converted to a convex optimization problem.

$$\begin{aligned}\max \sum_j \log(\sum_k \alpha_k\varphi_k(x_j^{test})), \\ s.t. \quad \sum_i\sum_k \alpha_k\varphi_k(x_j^{test}) = n_{train} \text{ and } \alpha_k\ge 0\end{aligned} \tag{24}$$

where n_{train} is the number of training data.

In this paper, our focus is to deal with the consistent metric learning problem. Readers are referred to [Tsuboi *et al.*, 2008] for details on estimating importance weights. There are also other candidate methods for estimating importance weights [Huang *et al.*, 2007]. An advantage of learning the weighting function is that it allows us to generalize importance weights to out-of-sample data. Another point from [Tsuboi *et al.*, 2008] is that the error of importance weight is proportional to $O(1/\sqrt{n})$, where n is the number of instances. For the method of estimating weights on pairs, we consider Δx as variable themselves in the above equations and we can construct more data instances to estimate the weights. Theoretically, we will have more accurate importance weights. However, from the experiments we will see that these additions will not always give better performance in practice.

5 Experiments

In this section, we evaluate the performance of our proposed methods on both synthetic and real world datasets. Since we have two weighting methods, we refer to the first one, which looks at instances and is described in Theorem 1 as CDML1 and the other one, which looks at instance pairs and is in Theorem 2 as CDML2.

5.1 Experiment Setup

We construct Gaussian mixture models for generating the training and test data. Data in the positive and negative

[1] http://www.math.nus.edu.sg/∼mattohkc/sdpt3.html

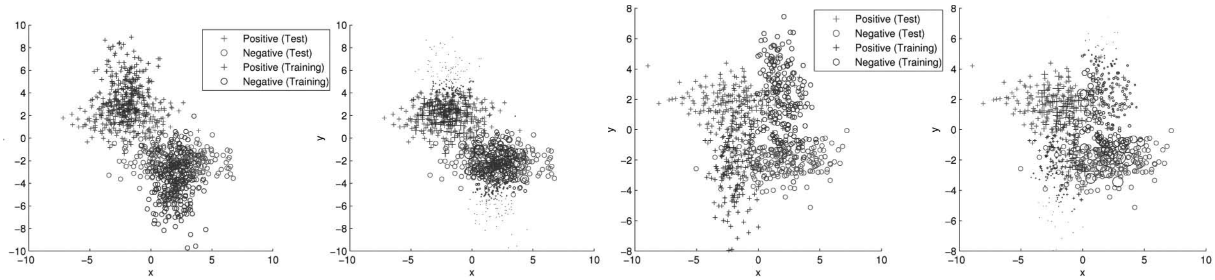

Figure 2: GMM1 and GMM2: Two Gaussian Mixture Datasets. The left figure shows the data display in the 2-dimensional space. The right one shows importance sampling results where the larger mark indicates more importance weight.

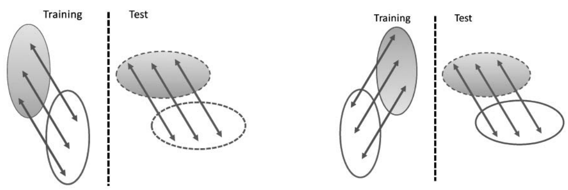

Figure 3: Case one: There exists significant covariate shift in $\Pr(x)$ but not in $Q(\delta_x)$. Case two: There exists significant covariate shift in $\Pr(x)$ as well as in $Q(\delta_x)$.

classes are generated from two 2-dimensional Gaussian mixture models respectively. The mixture weights for training and test data are different. Therefore covariate shift exists in the datasets. Figure 2 displays the two datasets. We also test our algorithm on real-world benchmark datasets. They include two datasets from UCI[2] and another two from IDA[3]. For the UCI datasets, we follow previous research work on sample selection bias: the covariate shift is simulated by artificially introducing bias [Huang *et al.*, 2006]. For the IDA datasets, they are already split into training and test sets and covariate shift already exists in the original split [Sugiyama *et al.*, 2008]. Therefore, we directly use the IDA datasets in the experiments.

We evaluated our method both on classification and clustering problems. For the classification task, the results are obtained by a K-Nearest-Neighbor (KNN) classifier based on the metric learned. The value is the average accuracy of KNN for K ranging from 1 to 5. We also perform clustering based on the metric learned. Normalized Mutual Information (NMI) is used to evaluate the clustering result, which is defined in [Xu *et al.*, 2003].

In all experiments, we randomly sample 50 pairs of must-link/cannot-link from the data and repeat 10 times to calculate the variance of results. The parameter u and l are chosen by the 5% and 95% percentiles of the distribution of Δx.

5.2 Experimental Results

In this section, we first use the synthetic datasets to illustrate the idea of CDML comprehensively. Then we compare our method with two baseline algorithms. The first one is using Euclidean distance directly. The second is information-theoretic metric learning (ITML), which is one state-of-the-art metric learning algorithm proposed by Davis et al. in [Davis *et al.*, 2007]. We conduct evaluation with both classification and clustering tasks.

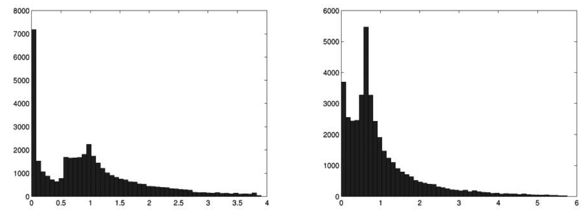

Figure 4: The figures shows the histogram of weight on pairs estimated by CDML2.

Table 1: Classification experiments results

	GMM1	GMM2	iris	wine
Euclidean	88.7(0.0)	69.8(0.0)	93.4 (6.1)	66.2 (1.9)
ITML	88.8(0.2)	73.7(0.5)	93.9(2.1)	70.8 (6.0)
CDML1	**92.7**(1.3)	**74.2**(2.2)	**94.5** (1.6)	**85.7** (8.7)
CDML2	88.3(0.7)	**74.1**(0.8)	93.3 (3.0)	**88.6** (9.7)

As shown in Figure 2, both GMM datasets have covariate shifts if we consider the distribution $\Pr(x)$. The two datasets are different when we consider the relation between their instances. Take the cannot-link pairs as an example, GMM1 does not have significant covariate shift when considering the distribution of $Q(\delta_x)$ as illustrated in Figure 3. However, in GMM2, not only $\Pr(x)$ but also $Q(\delta_x)$ has covariate shift, as shown in Figure 4. This may cause the difference between the results of CDML1 and CDML2. We can observe from Table 1 that the classification accuracy with KNN classifier based on CDML1 has significant improvement while CDML2 does not. For GMM2, both CDML1 and CDML2

Table 2: Classification results on IDA datasets.

Data	Euclidean	ITML	CDML1
splice	71.1(1.5)	74.4(1.4)	**74.5**(1.6)
ringnorm	65.2(1.4)	79.1(1.0)	**80.3**(0.5)

[2]http://www.ics.uci.edu/∼mlearn/MLRepository.html
[3]http://ida.first.fraunhofer.de/projects/bench/

Table 3: Clustering results

Data	Euclidean	ITML	CDML1
wine	0.46(0.03)	0.47(0.03)	**0.83**(0.04)
iris	0.72(0.09)	0.88(0.12)	**0.91**(0.07)

have significant improvement. Since δ_x is invariant with respect to translation, it cannot detect the shift in translation. This experiment shows that, although CDML2 has its advantage over CDML1 theoretically, in practice it may not outperform CDML1. It is interesting to see that CDML1 generally performs better. Table 2 shows the performance comparison using classification accuracy on IDA datasets. From this table, we can find that CDML outperforms Euclidean and is comparable to ITML. Table 3 shows the clustering results on two UCI datasets. We can see the improvement is even more significant than classification.

6 Related Work

The problem of covariate shift was introduced to machine learning community by [Zadrozny, 2004]. The problem of estimating importance weight is addressed by [Huang *et al.*, 2006; Sugiyama *et al.*, 2008]. Huang et al. proposed to use a non-parametric kernel methods which calculates the importance weights by minimizing the difference of means of training and test data in a universal kernel space [Huang *et al.*, 2006]. Sugiyama et al. proposed another similar approach which minimizes the KL divergence between the distributions. In [Sugiyama *et al.*, 2008], Bickel et al. proposed a method to unify the importance weight estimation step and supervised learning step together [Bickel *et al.*, 2007]. However, these works only focus on supervised learning problems. To our best knowledge, there is no previous work on considering covariate shift in metric learning problems.

7 Conclusion and Future Work

In this paper, we address the problem of consistent metric learning under covariate shift. A cost sensitive metric learning algorithm was proposed. Two importance weighting methods were proposed and analyzed. Experiments were carried out on both synthetic and real world datasets on both classification and clustering tasks. Currently we are using the general SDP solvers for our proposed problem. In the future, we plan to develop faster and more scalable algorithm for this problem.

Acknowledgments

Bin Cao and Qiang Yang would like to thank the support of Hong Kong RGC/NSFC Grant N_HKUST624/09 and RGC Grant 621010.

References

[Bickel *et al.*, 2007] Steffen Bickel, Michael Brckner, and Tobias Scheffer. Discriminative learning for differing training and test distributions. In *Proceedings of the 24th international conference on Machine learning*, Corvalis, Oregon, 2007. ACM.

[Daume and Marcu, 2006] III Daume and Daniel Marcu. Domain adaptation for statistical classifiers. *Journal of Artficial Intelligence Research*, 26:126, 101, 2006.

[Davis *et al.*, 2007] Jason V. Davis, Brian Kulis, Prateek Jain, Suvrit Sra, and Inderjit S. Dhillon. Information-theoretic metric learning. In *Proceedings of the 24th international conference on Machine learning*, Corvalis, Oregon, 2007. ACM.

[Grant and Boyd., 2009] M. Grant and S. Boyd. *CVX: Matlab software for disciplined convex programming (web page and software)*. http://stanford.edu/ boyd/cvx, May 2009.

[Huang *et al.*, 2006] Jiayuan Huang, Alexander Smola, Arthur Gretton, Karsten Borgwardt, and Bernhard Scholkopf. Correcting sample selection bias by unlabeled data. In *NIPS*, 2006.

[Huang *et al.*, 2007] J. Huang, A.J. Smola, Arthur Gretton, K.M. Borgwardt, and B. Scholkopf. Correcting sample selection bias by unlabeled data. *Advances in neural information processing systems*, 19:601, 2007.

[Kulis *et al.*, 2006] Brian Kulis, Mtys Sustik, and Inderjit Dhillon. Learning low-rank kernel matrices. In *Proceedings of the 23rd international conference on Machine learning*, pages 505–512, Pittsburgh, Pennsylvania, 2006. ACM.

[Pan *et al.*, 2008] Sinno Jialin Pan, Dou Shen, Qiang Yang, and James T. Kwok. Transferring localization models across space. In *AAAI*, pages 1383–1388, 2008.

[Shimodaira, 2000] H Shimodaira. Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):244, 227, October 2000.

[Sriperumbudur and Lanckriet, 2007] Bharath Sriperumbudur and Gert Lanckriet. Metric embedding for nearest neighbor classification. http://arxiv.org/abs/0706.3499, 2007.

[Sugiyama *et al.*, 2008] Masashi Sugiyama, Shinichi Nakajima, Hisashi Kashima, Paul Von Buenau, and Motoaki Kawanabe. Direct importance estimation with model selection and its application to covariate shift adaptation. In *Advances in Neural Information Processing Systems 20*. 2008.

[Tsuboi *et al.*, 2008] Yuta Tsuboi, Hisashi Kashima, Shohei Hido, Steffen Bickel, and Masashi Sugiyama. Direct density ratio estimation for large-scale covariate shift adaptation. In *SDM*, pages 443–454, 2008.

[Vapnik, 1995] Vladimir N. Vapnik. *The nature of statistical learning theory*. Springer-Verlag New York, Inc., New York, NY, USA, 1995.

[Weinberger and Saul, 2008] Kilian Q. Weinberger and Lawrence K. Saul. Fast solvers and efficient implementations for distance metric learning. In *ICML '08: Proceedings of the 25th international conference on Machine learning*, pages 1160–1167, New York, NY, USA, 2008. ACM.

[Xing *et al.*, 2003] Eric Xing, Andrew Ng, Michael Jordan, and Stuart Russell. Distance metric learning, with application to clustering with side-information. In *Advances in Neural Information Processing Systems 15*, volume 15, pages 512, 505, 2003.

[Xu *et al.*, 2003] Wei Xu, Xin Liu, and Yihong Gong. Document clustering based on non-negative matrix factorization. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 267–273, New York, NY, USA, 2003. ACM.

[Yeung and Chang, 2007] DY Yeung and H Chang. A kernel approach for semisupervised metric learning. *IEEE transactions on neural networks / a publication of the IEEE Neural Networks Council*, 18(1):149, 141, 2007.

[Zadrozny, 2004] Bianca Zadrozny. Learning and evaluating classifiers under sample selection bias. In *Proceedings of the twenty-first international conference on Machine learning*, page 114, Banff, Alberta, Canada, 2004. ACM.