Joint Semi-Supervised Similarity Learning for Linear Classification

Maria-Irina Nicolae[1,2], Éric Gaussier[2], Amaury Habrard[1], and Marc Sebban[1]

[1] Université Jean Monnet, Laboratoire Hubert Curien, France
[2] Université Grenoble Alpes, CNRS-LIG/AMA, France
{Irina.Nicolae,Eric.Gaussier}@imag.fr,
{Marc.Sebban,Amaury.Habrard}@univ-st-etienne.fr

Abstract. The importance of metrics in machine learning has attracted a growing interest for distance and similarity learning. We study here this problem in the situation where few labeled data (and potentially few unlabeled data as well) is available, a situation that arises in several practical contexts. We also provide a complete theoretical analysis of the proposed approach. It is indeed worth noting that the metric learning research field lacks theoretical guarantees that can be expected on the generalization capacity of the classifier associated to a learned metric. The theoretical framework of (ϵ, γ, τ)-good similarity functions [1] has been one of the first attempts to draw a link between the properties of a similarity function and those of a linear classifier making use of it. In this paper, we extend this theory to a method where the metric and the separator are jointly learned in a semi-supervised way, setting that has not been explored before, and provide a theoretical analysis of this joint learning via Rademacher complexity. Experiments performed on standard datasets show the benefits of our approach over state-of-the-art methods.

Keywords: similarity learning, (ϵ, γ, τ)-good similarity, Rademacher complexity.

1 Introduction

Many researchers have used the underlying geometry of the data to improve classification algorithms, *e.g.* by learning Mahanalobis distances instead of the standard Euclidean distance, thus paving the way for a new research area termed *metric learning* [5,6]. If most of these studies have based their approaches on distance learning [3,9,10,22,24], similarity learning has also attracted a growing interest [2,12,16,20], the rationale being that the cosine similarity should in some cases be preferred over the Euclidean distance. More recently, [1] have proposed a complete framework to relate similarities with a classification algorithm making use of them. This general framework, that can be applied to any bounded similarity function (potentially derived from a distance), provides generalization guarantees on a linear classifier learned from the similarity. Their algorithm

does not enforce the positive definiteness constraint of the similarity, like most state-of-the-art methods. However, to enjoy such generalization guarantees, the similarity function is assumed to be known beforehand and to satisfy (ϵ, γ, τ)-goodness properties. Unfortunately, [1] do not provide any algorithm for learning such similarities. In order to overcome these limitations, [4] have explored the possibility of independently learning an (ϵ, γ, τ)-good similarity that they plug into the initial algorithm [1] to learn the linear separator. Yet the similarity learning step is done in a completely supervised way, while the setting in [1] opens the door to the use of unlabeled data.

In this paper, we aim at better exploiting the semi-supervised setting underlying the theoretical framework of [1], which is based on similarities between labeled data and unlabeled reasonable points (roughly speaking, the reasonable points play the same role as that of support vectors in SVMs). Furthermore, and unlike [4], we propose here to jointly learn the metric and the classifier, so that both the metric and the separator are learned in a semi-supervised way. To our knowledge, this approach has not been explored before in metric learning. Enforcing (ϵ, γ, τ)-goodness allows us to preserve the theoretical guarantees from [1] on the classifier in relation to the properties of the similarity. We use the Rademacher complexity to derive new generalization bounds for the joint optimization problem. Lastly, we provide an empirical study on seven datasets and compare our method to different families of supervised and semi-supervised learning algorithms.

The remainder of this paper is organized as follows: Section 2 reviews some previous results in metric and similarity learning. Section 3 reminds the theory of (ϵ, γ, τ)-good similarities and introduces our method that jointly learns the metric and the linear classifier, followed in Section 4 by generalization guarantees for our formulation. Finally, Section 5 presents an experimental study on various standard datasets.

2 Related Work

We denote vectors by lower-case bold symbols ($\mathbf{x}$) and matrices by upper-case bold symbols ($\mathbf{A}$). Consider the following learning problem: we are given access to labeled examples $\mathbf{z} = (\mathbf{x}, y)$ drawn from some unknown distribution P over $\mathcal{S} = \mathcal{X} \times \mathcal{Y}$, where $\mathcal{X} \subseteq \mathbb{R}^d$ and $\mathcal{Y} = \{-1, 1\}$ are respectively the instance and the output spaces. A pairwise similarity function over $\mathcal{X}$ is defined as $K : \mathcal{X} \times \mathcal{X} \to [-1, 1]$, and the hinge loss as $[c]_+ = \max(0, 1 - c)$. We denote the L_1 norm by $||\cdot||_1$, the L_2 norm by $||\cdot||_2$ and the Frobenius norm by $||\cdot||_{\mathcal{F}}$.

Metric learning aims at finding the parameters of a distance or similarity function that best account for the underlying geometry of the data. This information is usually expressed as pair-based ($\mathbf{x}$ and $\mathbf{x}'$ should be (dis)similar) or triplet-based constraints ($\mathbf{x}$ should be more similar to $\mathbf{x}'$ than to $\mathbf{x}''$). Typically, the learned metric takes the form of a matrix and is the result of solving an optimization problem. The approaches that have received the most attention in this field involve learning a Mahalanobis distance, defined as $d_{\mathbf{A}}(\mathbf{x}, \mathbf{x}') =$

$\sqrt{(\mathbf{x}-\mathbf{x}')^T\mathbf{A}(\mathbf{x}-\mathbf{x}')}$, in which the distance is parameterized by the symmetric and positive semi-definite (PSD) matrix $\mathbf{A} \in \mathbb{R}^{d\times d}$. One nice feature of this type of approaches is its interpretability: the Mahalanobis distance implicitly corresponds to computing the Euclidean distance after linearly projecting the data to a different (possibly lower) feature space. The PSD constraint on $\mathbf{A}$ ensures $d_{\mathbf{A}}$ is a pseudo metric. Note that the Mahalanobis distance reduces to the Euclidean distant when $\mathbf{A}$ is set to the identity matrix.

Mahalanobis distances were used for the first time in metric learning in [25]. In this study, they aim to learn a PSD matrix $\mathbf{A}$ as to maximize the sum of distances between dissimilar instances, while keeping the sum of distances between similar instances small. Eigenvalue decomposition procedures are used to ensure that the learned matrix is PSD, which makes the computations intractable for high-dimensional spaces. In this context, LMNN [23,24] is one of the most widely-used Mahalanobis distance learning methods. The constraints they use are pair- and triplet-based, derived from each instance's nearest neighbors. The optimization problem they solve is convex and has a special-purpose solver. The algorithm works well in practice, but is sometimes prone to overfitting due to the absence of regularization, especially when dealing with high dimensional data. Another limitation is that enforcing the PSD constraint on $\mathbf{A}$ is computationally expensive. One can partly get around this latter shortcoming by making use of specific solvers or using information-theoretic approaches, such as ITML [9]. This work was the first one to use LogDet divergence for regularization, and thus provides an easy and cheap way for ensuring that $\mathbf{A}$ is a PSD matrix. However, the learned metric $\mathbf{A}$ strongly depends on the initial value $\mathbf{A}_0$, which is an important shortcoming, as $\mathbf{A}_0$ is handpicked.

The following metric learning methods use a semi-supervised setting, in order to improve the performance through the use of unlabeled data. LRML [14,15] learns Mahalanobis distances with manifold regularization using a Laplacian matrix. Their approach is applied to image retrieval and image clustering. LRML performs particularly well compared to fully supervised methods when side information is scarce. M-DML [28] uses a similar formulation to that of LRML, with the distinction that the regularization term is a weighted sum using multiple metrics, learned over auxiliary datasets. SERAPH [19] is a semi-supervised information-theoretic approach that also learns a Mahalanobis distance. The metric is optimized to maximize the entropy over labeled similar and dissimilar pairs, and to minimize it over unlabeled data.

However, learning Mahalanobis distances faces two main limitations. The first one is that enforcing the PSD and symmetry constraints on $\mathbf{A}$, beyond the cost it induces, often rules out natural similarity functions for the problem at hand. Secondly, although one can experimentally notice that state-of-the-art Mahalanobis distance learning methods give better accuracy than using the Euclidean distance, no theoretical guarantees are provided to establish a link between the quality of the metric and the behavior of the learned classifier. In this context, [21,20] propose to learn similarities with theoretical guarantees for the kNN based algorithm making use of them, on the basis of perceptron algorithm

presented in [11]. The performance of the classifier obtained is competitive with those of ITML and LMNN. More recently, [1] introduced a theory for learning with so called (ϵ, γ, τ)-good similarity functions based on non PSD matrices. This was the first stone to establish generalization guarantees for a *linear* classifier that would be learned with such similarities. Their formulation is equivalent to a relaxed L_1-norm SVM [29]. The main limitation of this approach is however that the similarity function K is predefined and [1] do not provide any learning algorithm to design (ϵ, γ, τ)-good similarities. This problem has been fixed by [4] who optimize the (ϵ, γ, τ)-goodness of a bilinear similarity function under Frobenius norm regularization. The learned metric is then used to build a good global linear classifier. Moreover, their algorithm comes with a uniform stability proof [8] which allows them to derive a bound on the generalization error of the associated classifier. However, despite good results in practice, one limitation of this framework is that it imposes to deal with strongly convex objective functions.

Recently, [13] extended the theoretical results of [4]. Using the Rademacher complexity (instead of the uniform stability) and Khinchin-type inequalities, they derive generalization bounds for similarity learning formulations that are regularized w.r.t. more general matrix-norms including the L_1 and the mixed $L_{(2,1)}$-norms. Moreover, they show that such bounds for the learned similarities can be used to upper bound the true risk of a linear SVM. The main distinction between this approach and our work is that we propose a method that jointly learns the metric and the linear separator at the same time. This allows us to make use of the semi-supervised setting presented by [1] to learn well with only a small amount of labeled data.

3 Joint Metric and Classifier Learning

In this section, we first briefly recall the (ϵ, γ, τ)-good framework [1] that we are using, prior to presenting our semi-supervised framework for jointly learning a similarity function and a linear separator from data. The (ϵ, γ, τ)-good framework is based on the following definition of a good similarity.

Definition 1. *[1] K is a (ϵ, γ, τ)-good similarity function in hinge loss for a learning problem P if there exists a random indicator function $R(\mathbf{x})$ defining a probabilistic set of "reasonable points" such that the following conditions hold:*

1. *We have*
$$\mathbb{E}_{(\mathbf{x},y)\sim P}\left[[1 - yg(\mathbf{x})/\gamma]_+\right] \leq \epsilon, \tag{1}$$
where $g(\mathbf{x}) = \mathbb{E}_{(\mathbf{x}',y'),R(\mathbf{x}')}\left[y'K(\mathbf{x},\mathbf{x}')|R(\mathbf{x}')\right]$.
2. $\Pr_{\mathbf{x}'}(R(\mathbf{x}')) \geq \tau$.

The first condition can be interpreted as having a $(1-\epsilon)$ proportion of examples $\mathbf{x}$ on average 2γ more similar to random reasonable examples $\mathbf{x}'$ of their own label than to random reasonable examples $\mathbf{x}'$ of the other label. It also expresses the tolerated margin violations in an averaged way: this allows

for more flexibility than pair- or triplet-based constraints. The second condition sets the minimum mass of reasonable points one must consider (greater than τ). Notice that no constraint is imposed on the form of the similarity function. Definition 1 is used to learn a linear separator:

Theorem 1. *[1] Let K be an (ϵ, γ, τ)-good similarity function in hinge loss for a learning problem P. For any $\epsilon_1 > 0$ and $0 < \delta < \gamma\epsilon_1/4$ let $\mathcal{S} = \{\mathbf{x}'_1, \mathbf{x}'_2, \ldots, \mathbf{x}'_{d_u}\}$ be a sample of $d_u = \frac{2}{\tau}\left(log(2/\delta) + 16\frac{log(2/\delta)}{(\epsilon_1\gamma)^2}\right)$ landmarks drawn from P. Consider the mapping $\phi^{\mathcal{S}} : \mathcal{X} \to \mathbb{R}^{d_u}$, $\phi^{\mathcal{S}}_i(\mathbf{x}) = K(\mathbf{x}, \mathbf{x}'_i), i \in \{1, \ldots, d_u\}$. With probability $1 - \delta$ over the random sample $\mathcal{S}$, the induced distribution $\phi^{\mathcal{S}}(P)$ in $\mathbb{R}^{d_u}$, has a separator achieving hinge loss at most $\epsilon + \epsilon_1$ at margin γ.*

In other words, if K is (ϵ, γ, τ)-good according to Definition 1 and enough points are available, there exists a linear separator $\boldsymbol{\alpha}$ with error arbitrarily close to ϵ in the space $\phi^{\mathcal{S}}$. The procedure for finding the separator involves two steps: first using d_u potentially unlabeled examples as landmarks to construct the feature space, then using a new labeled set of size d_l to estimate $\boldsymbol{\alpha} \in \mathbb{R}^{d_u}$. This is done by solving the following optimization problem:

$$\min_{\boldsymbol{\alpha}} \Big\{ \sum_{i=1}^{d_l} \Big[1 - \sum_{j=1}^{d_u} \alpha_j y_i K(\mathbf{x}_i, \mathbf{x}_j)\Big]_+ : \sum_{j=1}^{d_u} |\alpha_j| \leq 1/\gamma \Big\}. \tag{2}$$

This problem can be solved efficiently by linear programming. Furthermore, as it is L_1-constrained, tuning the value of γ will produce a sparse solution. Lastly, the associated classifier takes the following form:

$$y = \operatorname{sgn} \sum_{j=1}^{d_u} \alpha_j K(\mathbf{x}, \mathbf{x}_j). \tag{3}$$

We now extend this framework to jointly learn the similarity and the separator in a semi-supervised way. Let $\mathcal{S}$ be a sample set of d_l labeled examples $(\mathbf{x}, y) \in \mathcal{Z} = \mathcal{X} \times \{-1; +1\}$) and d_u unlabeled examples. Furthermore, let $K_{\mathbf{A}}(\mathbf{x}, \mathbf{x}')$ be a generic (ϵ, γ, τ)-good similarity function, parameterized by the matrix $\mathbf{A} \in \mathbb{R}^{d \times d}$. We assume that $K_{\mathbf{A}}(\mathbf{x}, \mathbf{x}') \in [-1; +1]$ and that $||\mathbf{x}||_2 \leq 1$, but all our developments and results can directly be extended to any bounded similarities and datasets. Our goal here is to find the matrix $\mathbf{A}$ and the global separator $\boldsymbol{\alpha} \in \mathbb{R}^{d_u}$ that minimize the empirical loss (in our case, the hinge loss) over a finite sample $\mathcal{S}$, with some guarantees on the generalization error of the associated classifier. To this end, we propose a generalization of Problem (2) based on a joint optimization of the metric and the global separator:

$$\min_{\boldsymbol{\alpha}, \mathbf{A}} \quad \sum_{i=1}^{d_l} \Big[1 - \sum_{j=1}^{d_u} \alpha_j y_i K_{\mathbf{A}}(\mathbf{x}_i, \mathbf{x}_j)\Big]_+ + \lambda ||\mathbf{A} - \mathbf{R}|| \tag{4}$$

$$\text{s.t.} \ \sum_{j=1}^{d_u} |\alpha_j| \leq 1/\gamma \tag{5}$$

$$\mathbf{A} \text{ diagonal}, \quad |A_{kk}| \leq 1, \quad 1 \leq k \leq d, \tag{6}$$

where $\lambda > 0$ is a regularization parameter, and $\mathbf{R} \in \mathbb{R}^{d \times d}$ is a fixed diagonal matrix such that $||\mathbf{R}|| \leq d$. Here, the notation $|| \cdot ||$ refers to a generic matrix norm, for instance L_1 or L_2 norms.

The novelty of this formulation is the *joint optimization* over $\mathbf{A}$ and $\boldsymbol{\alpha}$: by solving Problem (4), we are learning the metric and the separator at the same time. One of its significant advantages is that it extends the semi-supervised setting from the separator learning step to the metric learning, and the two problems are solved using the same data. This method can naturally be used in situations where one has access to few labeled examples and some unlabeled ones: the labeled examples are used in this case to select the unlabeled examples that will serve to classify new points. Another important advantage of our technique, coming from [1], is that the constraints on the pair of points do not need to be satisfied entirely, as the loss is averaged on all the reasonable points. In other words, this formulation is less restrictive than pair or triplet-based settings. Constraint (5) takes into account the desired margin γ and is the same as in [1]. Constraint (6) ensures that the learned similarity is bounded in $[-1; +1]$. Note that the diagonality constraint on $\mathbf{A}$ can be relaxed (in which case the bound constraint becomes $||\mathbf{A}|| \leq 1$ and $\mathbf{R}$ is no longer diagonal); we restrict ourselves to diagonal matrices to simplify the presentation and to limit the number of parameters to be learned.

The matrix $\mathbf{R}$ can be used to encode prior knowledge one has on the problem, in a way similar to what is proposed in [9]. If the non parameterized version of the similarity considered performs well, then a natural choice of $\mathbf{R}$ is the identity matrix I, so that the learned matrix will preserve the good properties of the non parameterized version (and will improve it through learning). Another form of prior knowledge relates to the importance of each feature according to the classes considered. Indeed, one may want to give more weight to features that are more representative of one of the classes $\{-1; +1\}$. One way to capture this importance is to compare the distributions of each feature on the two classes, *e.g.* through Kullback–Leibler (KL) divergence. We assume here that each feature follows a Gaussian distribution in each class, with means μ_1 (class $+1$) and μ_2 (class -1) and standard deviations σ_1 (class $+1$) and σ_2 (class -1). The KL divergence is expressed in that case as:

$$D_{KL}^k = \log\left(\frac{\sigma_1}{\sigma_2}\right) + \frac{1}{2}\left(\frac{\sigma_1^2}{\sigma_2^2} - \frac{\sigma_2^2}{\sigma_1^2} + \frac{(\mu_2 - \mu_1)^2}{\sigma_2^2}\right), \quad 1 \leq k \leq d.$$

and the matrix $\mathbf{R}$ corresponds to $\text{diag}(D_{KL}^1, D_{KL}^2, \cdots, D_{KL}^d)$.

Lastly, once $\mathbf{A}$ and $\boldsymbol{\alpha}$ have been learned, the associated (binary) classifier takes the form given in Eq. (3).

4 Generalization Bound for Joint Similarity Learning

In this section, we establish a generalization bound for our joint similarity learning formulation (4) under constraints (5) and (6). This theoretical analysis is

based on the Rademacher complexity and holds for any regularization parameter $\lambda > 0$. Note that when $\lambda = 0$, we can also prove consistency results based on the algorithmic robustness framework [26,27]. In such a case, the proof is similar to the one in [18]. Before stating the generalization bound, we first introduce some notations.

Definition 2. *A pairwise similarity function $K_{\mathbf{A}} : \mathcal{X} \times \mathcal{X} \to [-1, 1]$, parameterized by a matrix $\mathbf{A} \in \mathbb{R}^{d \times d}$, is said to be (β, c)-admissible if, for any matrix norm $||\cdot||$, there exist $\beta, c \in \mathbb{R}$ such that $\forall \mathbf{x}, \mathbf{x}' \in \mathcal{X}, |K_{\mathbf{A}}(\mathbf{x}, \mathbf{x}')| \leq \beta + c \cdot ||\mathbf{x}'\mathbf{x}^T|| \cdot ||\mathbf{A}||$.*

Examples: Using some classical norm properties and the Frobenius inner product, we can show that:

- The bilinear similarity $K^1_{\mathbf{A}}(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T\mathbf{A}\mathbf{x}'$ is $(0, 1)$-admissible, that is $|K^1_{\mathbf{A}}(\mathbf{x}, \mathbf{x}')| \leq ||\mathbf{x}'\mathbf{x}^T|| \cdot ||\mathbf{A}||$;
- The similarity derived from the Mahalanobis distance $K^2_{\mathbf{A}}(\mathbf{x}, \mathbf{x}') = 1 - (\mathbf{x} - \mathbf{x}')^T\mathbf{A}(\mathbf{x} - \mathbf{x}')$ is $(1, 4)$-admissible, that is $|K^2_{\mathbf{A}}(\mathbf{x}, \mathbf{x}')| \leq 1 + 4 \cdot ||\mathbf{x}'\mathbf{x}^T|| \cdot ||\mathbf{A}||$.

Note that we will make use of these two similarity functions $K^1_{\mathbf{A}}$ and $K^2_{\mathbf{A}}$ in our experiments. For any $\mathbf{B}, \mathbf{A} \in \mathbb{R}^{n \times d}$ and any matrix norm $||\cdot||$, its dual norm $||\cdot||_*$ is defined, for any $\mathbf{B}$, by $||\mathbf{B}||_* = \sup_{||\mathbf{A}|| \leq 1} \mathrm{Tr}(\mathbf{B}^T\mathbf{A})$, where $\mathrm{Tr}(\cdot)$ denotes the trace of a matrix. Denote $X_* = \sup_{\mathbf{x}, \mathbf{x}' \in \mathcal{X}} ||\mathbf{x}'\mathbf{x}^T||_*$.

Let us now rewrite the minimization problem (4) with a more generalized notation of the loss function:

$$\min_{\boldsymbol{\alpha}, \mathbf{A}} \quad \frac{1}{d_l} \sum_{i=1}^{d_l} \ell(\mathbf{A}, \boldsymbol{\alpha}, \mathbf{z}_i = (\mathbf{x}_i, y_i)) + \lambda ||\mathbf{A} - \mathbf{R}||, \tag{7}$$

$$\text{s.t.} \sum_{j=1}^{d_u} |\alpha_j| \leq 1/\gamma \tag{8}$$

$$\mathbf{A} \text{ diagonal}, \quad |A_{kk}| \leq 1, \quad 1 \leq k \leq d, \tag{9}$$

where $\ell(\mathbf{A}, \boldsymbol{\alpha}, \mathbf{z}_i = (\mathbf{x}_i, y_i))) = \left[1 - \sum_{j=1}^{d_u} \alpha_j y_i K_{\mathbf{A}}(\mathbf{x}_i, \mathbf{x}_j)\right]_+$ is the instantaneous loss estimated at point $(\mathbf{x}_i, y_i)$. Note that from constraints (8) and (9), we deduce that $||\mathbf{A}|| < d$. Let $\mathcal{E}_{\mathcal{S}}(\mathbf{A}, \boldsymbol{\alpha}) = \frac{1}{d_l} \sum_{i=1}^{d_l} \ell(\mathbf{A}, \boldsymbol{\alpha}, \mathbf{z}_i)$ be the overall empirical loss over the training set $\mathcal{S}$, and let $\mathcal{E}(\mathbf{A}, \boldsymbol{\alpha}) = \mathbb{E}_{\mathbf{z} \sim \mathcal{Z}} \ell(\mathbf{A}, \boldsymbol{\alpha}, \mathbf{z})$ be the true loss w.r.t. the unknown distribution $\mathcal{Z}$. The target of generalization analysis for joint similarity learning is to bound the difference $\mathcal{E}(\mathbf{A}, \boldsymbol{\alpha}) - \mathcal{E}_{\mathcal{S}}(\mathbf{A}, \boldsymbol{\alpha})$.

Our generalization bound is based on the Rademacher complexity which can be seen as an alternative notion of the complexity of a function class and has the particularity to be (unlike the VC-dimension) data-dependent.

Definition 3. *Let $\mathcal{F}$ be a class of uniformly bounded functions. For every integer n, we call*

$$R_n(\mathcal{F}) := \mathbb{E}_{\mathcal{S}} \mathbb{E}_{\sigma} \left[\sup_{f \in \mathcal{F}} \frac{1}{n} \sum_{i=1}^{n} \sigma_i f(z_i) \right]$$

the Rademacher average over $\mathcal{F}$, where $\mathcal{S} = \{z_i : i \in \{1, \ldots, n\}\}$ are independent random variables distributed according to some probability measure and $\{\sigma_i : i \in \{1, \ldots, n\}\}$ are independent Rademacher random variables, that is, $\Pr(\sigma_i = 1) = \Pr(\sigma_i = -1) = \frac{1}{2}$.

The Rademacher average w.r.t. the dual matrix norm is then defined as:

$$\mathcal{R}_{d_l} := \mathbb{E}_{\mathcal{S},\sigma}\left[\sup_{\tilde{\mathbf{x}}\in\mathcal{X}}\left|\left|\frac{1}{d_l}\sum_{i=1}^{d_l}\sigma_i y_i \mathbf{x}_i \tilde{\mathbf{x}}^T\right|\right|_*\right]$$

We can now state our generalization bound.

Theorem 2. *Let $(\mathbf{A}_{\mathcal{S}}, \boldsymbol{\alpha}_{\mathcal{S}})$ be the solution to the joint problem* (7) *and $K_{\mathbf{A}_{\mathcal{S}}}$ a (β, c)-admissible similarity function. Then, for any $0 < \delta < 1$, with probability at least $1 - \delta$, the following holds:*

$$\mathcal{E}(\mathbf{A}_{\mathcal{S}}, \boldsymbol{\alpha}_{\mathcal{S}}) - \mathcal{E}_{\mathcal{S}}(\mathbf{A}_{\mathcal{S}}, \boldsymbol{\alpha}_{\mathcal{S}}) \leq 4\mathcal{R}_{d_l}\left(\frac{cd}{\gamma}\right) + \left(\frac{\beta + cX_* d}{\gamma}\right)\sqrt{\frac{2\ln\frac{1}{\delta}}{d_l}}.$$

Theorem 2 proves that learning $\mathbf{A}$ and $\boldsymbol{\alpha}$ in a joint manner from a training set minimizes the generalization error, as the latter is bounded by the empirical error of our joint regularized formulation. Its proof makes use of the Rademacher symmetrization theorem and contraction property (Theorem 3 and Lemma 1):

Theorem 3. *[7] Let $R_n(\mathcal{F})$ be the Rademacher average over $\mathcal{F}$ defined as previously. We have:*

$$\mathbb{E}\left[\sup_{f\in\mathcal{F}}\left(\mathbb{E}f(\mathcal{S}) - \frac{1}{n}\sum_{i=1}^{n} f(z_i)\right)\right] \leq 2R_n(\mathcal{F}).$$

Lemma 1. *[17] Let F be a class of uniformly bounded real-valued functions on (Ω, μ) and $m \in \mathbb{N}$. If for each $i \in \{1, \ldots, m\}$, $\phi_i : \mathbb{R} \to \mathbb{R}$ is a function having a Lipschitz constant c_i, then for any $\{x_i\}_{i\in\mathbb{N}_m}$,*

$$\mathbb{E}_\epsilon\left(\sup_{f\in F}\sum_{i\in\mathbb{N}_m}\epsilon_i\phi_i(f(x_i))\right) \leq 2\mathbb{E}_\epsilon\left(\sup_{f\in F}\sum_{i\in\mathbb{N}_m} c_i\epsilon_i f(x_i)\right).$$

Proof (Theorem 2).

Let $\mathbb{E}_{\mathcal{S}}$ denote the expectation with respect to sample $\mathcal{S}$. Observe that $\mathcal{E}_{\mathcal{S}}(\mathbf{A}_{\mathcal{S}}, \boldsymbol{\alpha}_{\mathcal{S}}) - \mathcal{E}(\mathbf{A}_{\mathcal{S}}, \boldsymbol{\alpha}_{\mathcal{S}}) \leq \sup_{\mathbf{A},\boldsymbol{\alpha}}[\mathcal{E}_{\mathcal{S}}(\mathbf{A}, \boldsymbol{\alpha}) - \mathcal{E}(\mathbf{A}, \boldsymbol{\alpha})]$. Also, for any $\mathcal{S} = (\mathbf{z}_1, \ldots, \mathbf{z}_k, \ldots, \mathbf{z}_{d_l})$ and $\tilde{\mathcal{S}} = (\mathbf{z}_1, \ldots, \tilde{\mathbf{z}}_k, \ldots, \mathbf{z}_{d_l})$, $\quad 1 \leq k \leq d_l$:

$$\begin{aligned}&\left|\sup_{\mathbf{A},\boldsymbol{\alpha}}[\mathcal{E}_{\mathcal{S}}(\mathbf{A}, \boldsymbol{\alpha}) - \mathcal{E}(\mathbf{A}, \boldsymbol{\alpha})] - \sup_{\mathbf{A},\boldsymbol{\alpha}}[\mathcal{E}_{\tilde{\mathcal{S}}}(\mathbf{A}, \boldsymbol{\alpha}) - \mathcal{E}(\mathbf{A}, \boldsymbol{\alpha})]\right| \\ \leq &\sup_{\mathbf{A},\boldsymbol{\alpha}}|\mathcal{E}_{\mathcal{S}}(\mathbf{A}, \boldsymbol{\alpha}) - \mathcal{E}_{\tilde{\mathcal{S}}}(\mathbf{A}, \boldsymbol{\alpha})|\end{aligned}$$

$$
\begin{aligned}
&= \frac{1}{d_l} \sup_{\mathbf{A},\boldsymbol{\alpha}} \left| \sum_{\mathbf{z}=(\mathbf{x},y)\in\mathcal{S}} \left[1 - \sum_{j=1}^{d_u} \alpha_j y K_{\mathbf{A}}(\mathbf{x},\mathbf{x}_j) \right]_+ - \sum_{\tilde{\mathbf{z}}=(\tilde{\mathbf{x}},\tilde{y})\in\tilde{\mathcal{S}}} \left[1 - \sum_{j=1}^{d_u} \alpha_j \tilde{y} K_{\mathbf{A}}(\tilde{\mathbf{x}},\mathbf{x}_j) \right]_+ \right| \\
&= \frac{1}{d_l} \sup_{\mathbf{A},\boldsymbol{\alpha}} \left| \left[1 - \sum_{j=1}^{d_u} \alpha_j y_k K_{\mathbf{A}}(\mathbf{x}_k,\mathbf{x}_j) \right]_+ - \left[1 - \sum_{j=1}^{d_u} \alpha_j \tilde{y}_k K_{\mathbf{A}}(\tilde{\mathbf{x}}_k,\mathbf{x}_j) \right]_+ \right| \\
&= \frac{1}{d_l} \sup_{\mathbf{A},\boldsymbol{\alpha}} \left| \sum_{j=1}^{d_u} \alpha_j \tilde{y}_k K_{\mathbf{A}}(\tilde{\mathbf{x}}_k,\mathbf{x}_j) - \sum_{j=1}^{d_u} \alpha_j y_k K_{\mathbf{A}}(\mathbf{x}_k,\mathbf{x}_j) \right| \qquad (10) \\
&\leq \frac{2}{d_l} \sup_{\mathbf{A},\boldsymbol{\alpha}} \left| \sum_{j=1}^{d_u} \alpha_j y_k^{max} K_{\mathbf{A}}(\mathbf{x}_k^{max},\mathbf{x}_j) \right| \text{ where } \mathbf{z}_k^{max} = \underset{\mathbf{z}=(\mathbf{x},y)\in\{\mathbf{z}_k,\tilde{\mathbf{z}}_k\}}{\arg\max} y K_{\mathbf{A}}(\mathbf{x},\mathbf{x}_j) \\
&\leq \frac{2}{d_l} \sup_{\mathbf{A},\boldsymbol{\alpha}} \left\{ \sum_{j=1}^{d_u} |\alpha_j| \cdot |y_k^{max}| \cdot |K_{\mathbf{A}}(\mathbf{x}_k^{max},\mathbf{x}_j)| \right\} \\
&\leq \frac{2}{d_l} \left(\frac{\beta + cX_* d}{\gamma} \right) \qquad (11)
\end{aligned}
$$

Inequality (10) comes from the 1-lipschitzness of the hinge loss; Inequality (11) comes from Constraint (8), $||\mathbf{A}|| \leq d$ and the (β, c)-admissibility of $K_{\mathbf{A}}$. Applying McDiarmid's inequality to the term $\sup_{\mathbf{A},\boldsymbol{\alpha}} [\mathcal{E}_{\mathcal{S}}(\mathbf{A},\boldsymbol{\alpha}) - \mathcal{E}(\mathbf{A},\boldsymbol{\alpha})]$, with probability $1-\delta$, we have:

$$
\sup_{\mathbf{A},\boldsymbol{\alpha}} [\mathcal{E}_{\mathcal{S}}(\mathbf{A},\boldsymbol{\alpha}) - \mathcal{E}(\mathbf{A},\boldsymbol{\alpha})] \leq \mathbb{E}_{\mathcal{S}} \sup_{\mathbf{A},\boldsymbol{\alpha}} [\mathcal{E}_{\mathcal{S}}(\mathbf{A},\boldsymbol{\alpha}) - \mathcal{E}(\mathbf{A},\boldsymbol{\alpha})] + \left(\frac{\beta + cX_* d}{\gamma} \right) \sqrt{\frac{2 \ln \frac{1}{\delta}}{d_l}}.
$$

In order to bound the gap between the true loss and the empirical loss, we now need to bound the expectation term of the right hand side of the above equation.

$$
\begin{aligned}
&\mathbb{E}_{\mathcal{S}} \sup_{\mathbf{A},\boldsymbol{\alpha}} [\mathcal{E}_{\mathcal{S}}(\mathbf{A},\boldsymbol{\alpha}) - \mathcal{E}(\mathbf{A},\boldsymbol{\alpha})] \\
&= \mathbb{E}_{\mathcal{S}} \sup_{\mathbf{A},\boldsymbol{\alpha}} \left\{ \frac{1}{d_l} \sum_{i=1}^{d_l} \left[1 - \sum_{j=1}^{d_u} \alpha_j y_i K_{\mathbf{A}}(\mathbf{x}_i,\mathbf{x}_j) \right]_+ - \mathcal{E}(\mathbf{A},\boldsymbol{\alpha}) \right\} \\
&\leq 2\mathbb{E}_{\mathcal{S},\sigma} \sup_{\mathbf{A},\boldsymbol{\alpha}} \left\{ \frac{1}{d_l} \sum_{i=1}^{d_l} \sigma_i \left[1 - \sum_{j=1}^{d_u} \alpha_j y_i K_{\mathbf{A}}(\mathbf{x}_i,\mathbf{x}_j) \right]_+ \right\} \qquad (12) \\
&\leq 4\mathbb{E}_{\mathcal{S},\sigma} \sup_{\mathbf{A},\boldsymbol{\alpha}} \left| \frac{1}{d_l} \sum_{i=1}^{d_l} \sigma_i y_i \sum_{j=1}^{d_u} \alpha_j K_{\mathbf{A}}(\mathbf{x}_i,\mathbf{x}_j) \right| \qquad (13) \\
&\leq 4 \left(\frac{cd}{\gamma} \right) \mathbb{E}_{\mathcal{S},\sigma} \sup_{\tilde{\mathbf{x}}} \left\| \frac{1}{d_l} \sum_{i=1}^{d_l} \sigma_i y_i \mathbf{x}_i \tilde{\mathbf{x}}^T \right\|_* = 4\mathcal{R}_{d_l} \left(\frac{cd}{\gamma} \right). \qquad (14)
\end{aligned}
$$

We obtain Inequality (12) by applying Theorem 3, while Inequality (13) comes from the use of Lemma 1. The Inequality on line (14) makes use of the (β, c)-admissibility of the similarity function $K_{\mathbf{A}}$ (Definition 2). Combining Inequalities (11) and (14) completes the proof of the theorem. □

After proving the generalization bound of our joint similarity approach, we now move to the experimental validation of the approach proposed.

5 Experiments

The state of the art in metric learning is dominated by algorithms designed to work in a purely supervised setting. Furthermore, most of them optimize a metric adapted to kNN classification (*e.g.* LMNN, ITML), while our work is designed for finding a global linear separator. For these reasons, it is difficult to propose a totally fair comparative study. In this section, we first evaluate the effectiveness of problem (4) with constraints (5) and (6) (JSL, for Joint Similarity Learning) with different settings. Secondly, we extensively compare it with state-of-the-art algorithms from different categories (supervised, kNN-oriented). Lastly, we study the impact of the quantity of available labeled data on our method. We conduct the experimental study on 7 classic datasets taken from the UCI Machine Learning Repository[3], both binary and multi-class. Their characteristics are presented in Table 1. These datasets are widely used for metric learning evaluation.

Table 1: Properties of the datasets used in the experimental study.

	Balance	Ionosphere	Iris	Liver	Pima	Sonar	Wine
# Instances	625	351	150	345	768	208	178
# Dimensions	4	34	4	6	8	60	13
# Classes	3	2	3	2	2	2	3

5.1 Experimental setting

In order to provide a comparison as complete as possible, we compare two main families of approaches[4]:

1. *Linear classifiers*: in this family, we consider the following methods:
   - BBS, corresponding to Problem (2) and discussed above;
   - SLLC [4], an extension of BBS in which a similarity is learned prior to be used in the BBS framework;

[3] http://archive.ics.uci.edu/ml/.
[4] For all the methods, we used the code provided by the authors.

- JSL, the joint learning framework proposed in this study;
- Linear SVM with L_2 regularization, which is the standard approach for linear classification;

2. *Nearest neighbor approaches*: in this family, we consider the methods:
   - Standard 3-nearest neighbor classifier (3NN) based on the Euclidean distance;
   - ITML [9], which learns a Mahalanobis distance that is used here in 3NN classification;
   - LMNN with a full matrix and LMNN with a diagonal matrix (LMNN-diag) [23,24], also learning a Mahalanobis distance used here in 3NN classification;
   - LRML [14,15]; LRML also learns a Mahalanobis distance used in 3NN classifier, but in a semi-supervised setting. This method is thus the "most" comparable to JSL (even though one is learning a linear separator and the other only a distance).

All classifiers are used in their binary version, in a one-vs-all setting when the number of classes is greater than two. BBS, SLLC and JSL rely on the same classifier from Eq. (3), even though learned in different ways. We solve BBS and JSL using projected gradient descent. In JSL, we rely on an alternating optimization that consists in fixing $\mathbf{A}$ (resp. $\boldsymbol{\alpha}$) and optimizing for $\boldsymbol{\alpha}$ (resp. $\mathbf{A}$), then changing the variable, until convergence.

Data processing and parameter settings All features are centered around zero and scaled to ensure $||\mathbf{x}||_2 \leq 1$, as this constraint is necessary for some of the algorithms. We randomly choose 15% of the data for validation purposes, and another 15% as a test set. The training set and the unlabeled data are chosen from the remaining 70% of examples not employed in the previous sets. In order to illustrate the classification using a restricted quantity of labeled data, the number of labeled points is limited to 5, 10 or 20 examples per class, as this is usually a reasonable minimum amount of annotation to rely on. The number of landmarks is either set to 15 points or to all the points in the training set (in which case their label is not taken into account). These two settings correspond to two practical scenarios: one in which a relatively small amount of unlabeled data is available, and one in which a large amount of unlabeled data is available. When only 15 unlabeled points are considered, they are chosen from the training set as the nearest neighbor of the 15 centroids obtained by applying k-means++ clustering with $k = 15$. All of the experimental results are averaged over 10 runs, for which we compute a 95% confidence interval. We tune the following parameters by cross-validation: $\gamma, \lambda \in \{10^{-4}, \ldots, 10^{-1}\}$ for BBS and JSL (λ only for the second), $\lambda_{ITML} \in \{10^{-4}, \ldots, 10^{4}\}$, choosing the value yielding the best accuracy. For SLLC, we tune $\gamma, \beta \in \{10^{-7}, \ldots, 10^{-2}\}$, $\lambda \in \{10^{-3}, \ldots, 10^{2}\}$, as done by the authors, while for LRML we consider $\gamma_s, \gamma_d, \gamma_i \in \{10^{-2}, \ldots, 10^{2}\}$. For LMNN, we set $\mu = 0.5$, as done in [24].

5.2 Experimental results

Analysis of JSL We first study here the behavior of the proposed joint learning framework w.r.t. different families of similarities and regularization functions (choice of $\mathbf{R}$ and $||\cdot||$). In particular, we consider two types of similarity measures: bilinear (cosine-like) similarities of the form $K^1_{\mathbf{A}}(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T\mathbf{A}\mathbf{x}'$ and similarities derived from the Mahalanobis distance $K^2_{\mathbf{A}}(\mathbf{x}, \mathbf{x}') = 1 - (\mathbf{x} - \mathbf{x}')^T\mathbf{A}(\mathbf{x} - \mathbf{x}')$. For the regularization term, $\mathbf{R}$ is either set to the identity matrix (JSL-I), or to the approximation of the Kullback–Leibler divergence (JSL-KL) discussed in Section 3. As mentioned above, these two settings correspond to different prior knowledge one can have on the problem. In both cases, we consider L_1 and L_2 regularization norms. We thus obtain 8 settings, that we compare in the situation where few labeled points are available (5 points per class), using a small amount (15 instances) of unlabeled data or a large amount (the whole training set) of unlabeled data. The results of the comparisons are reported in Tables 2 and 3.

As one can note from Table 2, when only 15 points are used as landmarks, $K^2_{\mathbf{A}}$ obtains better results in almost all of the cases, the difference being more important on Iris, Pima and Sonar. The noticeable exception to this better behavior of $K^2_{\mathbf{A}}$ is Wine, for which cosine-like similarities outperform Mahalanobis-based similarities by more than 10 points. A similar result was also presented in [21]. The difference between the use of the L_1 or L_2 norms is not as marked, and there is no strong preference for one or the other, even though the L_2 norm leads to slightly better results in average than the L_1 norm. Regarding the regularization matrix $\mathbf{R}$, again, the difference is not strongly marked, except maybe on Sonar. In average, regularizing through the Kullback-Leibler divergence leads to slightly better results than regularizing through the identity matrix.

When all points are used as landmarks (Table 3), similar conclusions can be drawn regarding the similarity functions and the norms used. However, in that case, the regularization based on the identity matrix yields better results than the one based on the KL divergence. It is important to note also that the overall results are in general lower than the ones obtained when only 15 points are used as landmarks. We attribute this effect to the fact that one needs to learn more parameters (via $\boldsymbol{\alpha}$), whereas the amount of available labeled data is the same.

From the above analysis, we focus now on two JSL based methods: JSL-15 with $K^2_{\mathbf{A}}$, L_2 norm and $\mathbf{R}$ =KL when 15 points are used as landmarks, and JSL-all with $K^2_{\mathbf{A}}$, L_2 norm and $\mathbf{R} = I$ when all the points are used as landmarks.

Comparison of the different methods We now study the performance of our method, compared to state-of-the-art algorithms. For this, we consider JSL-15 and JSL-all with 5, 10, respectively 20 labeled examples per class. As our methods are tested using the similarity based on the Mahalanobis distance, we use the euclidean distance for BBS to ensure fairness.

Figure 1 presents the average accuracy per dataset obtained with 5 labeled points per class. In this setting, JSL outperforms the other algorithms on 5 out of 7 datasets and has similar performances on one other. The exception is the Wine dataset, where none of the JSL settings yields competitive results. As stated before, this is easily explained by the fact cosine-similarities are more

Table 2: Average accuracy (%) with confidence interval at 95%, 5 labeled points per class, 15 unlabeled landmarks.

Sim.	Reg.	Balance	Ionosphere	Iris	Liver	Pima	Sonar	Wine
$K^1_{\mathbf{A}}$	I-L_1	85.2±3.0	85.6±2.4	76.8±3.2	63.3±6.2	71.0±4.1	72.9±3.6	**91.9**±4.2
	I-L_2	85.1±2.9	85.6±2.6	76.8±3.2	63.1±6.3	71.0±4.0	73.2±3.8	91.2±4.5
	KL-L_1	84.9±2.9	85.0±2.6	77.3±2.7	63.9±5.5	71.0±4.0	72.9±3.6	90.8±4.7
	KL-L_2	85.2±3.0	85.8±3.3	76.8±3.2	62.9±6.4	71.3±4.3	74.2±3.8	90.0±5.4
$K^2_{\mathbf{A}}$	I-L_1	**87.2**±2.9	**87.7**±2.6	78.6±4.6	64.7±5.6	75.1±3.5	73.9±5.7	80.8±9.5
	I-L_2	86.8±3.0	**87.7**±2.8	75.9±5.7	64.3±5.4	**75.6**±3.6	74.8±5.8	80.8±8.6
	KL-L_1	**87.2**±2.9	87.3±2.4	78.6±4.6	62.9±5.6	75.0±3.7	75.5±6.2	79.6±11.8
	KL-L_2	**87.1**±2.7	85.8±3.3	**79.1**±5.4	**64.9**±5.9	**75.6**±3.4	**77.1**±5.2	79.6±9.7

Table 3: Average accuracy (%) with confidence interval at 95%, all points used as landmarks.

Sim.	Reg.	Balance	Ionosphere	Iris	Liver	Pima	Sonar	Wine
$K^1_{\mathbf{A}}$	I-L_1	85.8±2.9	88.8±2.5	74.5±3.1	65.5±4.5	71.4±3.8	70.3±6.6	85.8±5.0
	I-L_2	85.8±2.9	87.7±2.7	74.5±3.5	64.7±5.5	71.7±4.1	68.7±6.7	84.6±5.5
	KL-L_1	85.6±3.1	87.9±3.4	75.0±3.5	65.3±4.9	71.6±4.2	70.3±6.8	85.4±5.3
	KL-L_2	85.1±3.1	88.5±3.7	**75.9**±3.4	65.1±4.8	72.1±4.2	71.9±6.7	**86.5**±6.0
$K^2_{\mathbf{A}}$	I-L_1	85.9±2.3	90.4±2.2	71.8±6.1	67.3±3.5	73.1±3.5	72.9±4.2	81.5±8.4
	I-L_2	**86.2**±2.5	**90.6**±2.2	73.2±6.6	**68.6**±3.3	73.3±3.2	**73.2**±4.2	82.7±9.0
	KL-L_1	85.8±2.6	89.4±2.0	72.7±5.5	67.5±3.8	**73.8**±3.5	71.0±4.1	80.0±7.4
	KL-L_2	85.9±2.4	89.6±2.2	74.5±6.2	68.4±3.6	73.1±3.8	72.3±4.8	80.0±11.5

adapted for this dataset. Even though JSL-15 and JSL-all perform the same when averaged over all datasets, the difference between them is marked on some datasets: JSL-15 is considerably better on Iris and Sonar, while JSL-all significantly outperforms JSL-15 on Ionosphere and Liver. Averaged over all datasets (Table 4), JSL obtains the best performance in all configurations with a limited amount of labeled data, which is particularly the setting that our method is designed for. The values in bold are significantly better than the rest of their respective columns, confirmed by a one-sided Student t-test for paired samples with a significance level of 5%.

Impact of the amount of labeled data As an illustration of the methods' behavior when the level of supervision varies, Figure 2 presents the accuracies on two representative datasets, Ionosphere and Pima, with an increasing number of labeled examples. In both cases, the best results are obtained by JSL (and more precisely JSL-15) when less than 50% of the training set is used. This is in agreement with the results reported in Table 4. The results of JSL are furthermore comparable only to BBS for the Pima dataset. Lastly, the accuracy of JSL improves slightly when adding more labeled data, and the results on the whole training set are competitive w.r.t. the other algorithms.

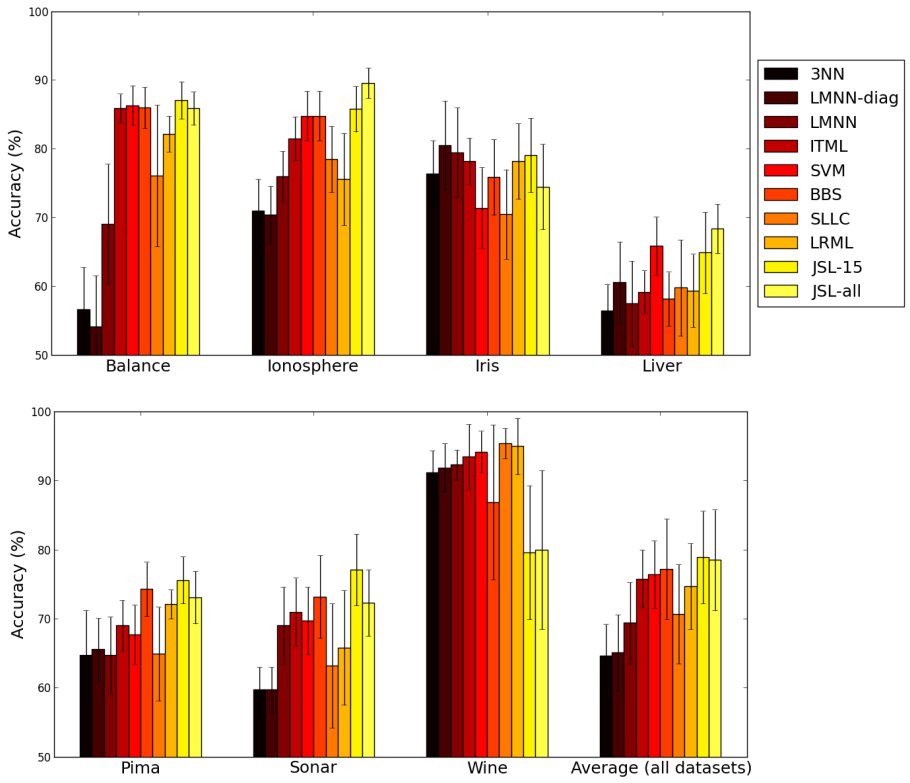


Fig. 1: Average accuracy (%) with confidence interval at 95%, 5 labeled points per class, 15 unlabeled landmarks.

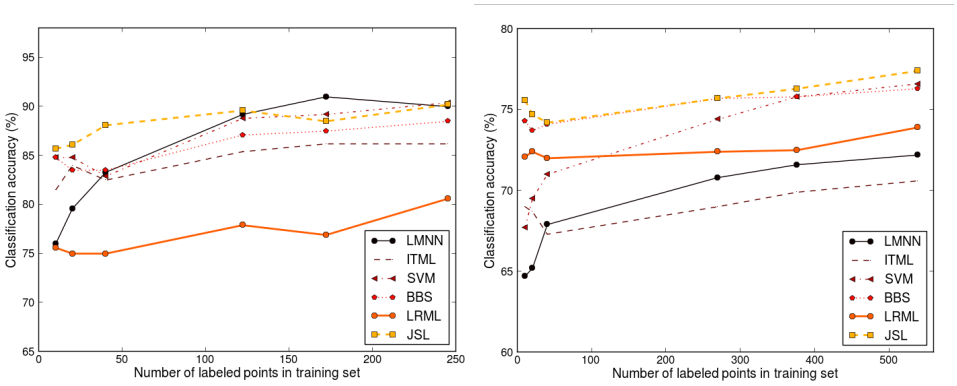
6 Conclusion

In this paper, we have studied the problem of learning similarities in the situation where few labeled (and potentially few unlabeled) data is available. To do so, we have developed a semi-supervised framework, extending the (ϵ, γ, τ)-good of [1], in which the similarity function and the classifier are learned at the same time. To our knowledge, this is the first time that such a framework is provided. The joint learning of the similarity and the classifier enables one to benefit from unlabeled data for both the similarity and the classifier. We have also showed that the proposed method was theoretically well-founded as we derived a Rademacher-based bound on the generalization error of the learned parameters. Lastly, the experiments we have conducted on standard metric learning datasets show that our approach is indeed well suited for learning with few labeled data, and outperforms state-of-the-art metric learning approaches in that situation.

Acknowledgements: Funding for this project was provided by a grant from Région Rhône-Alpes.

Table 4: Average accuracy (%) over all datasets with confidence interval at 95%.

Method	5 pts./cl.	10 pts./cl.	20 pts./cl.
3NN	64.6±4.6	68.5±5.4	70.4±5.0
LMNN-diag	65.1±5.5	68.2±5.6	71.5±5.2
LMNN	69.4±5.9	70.9±5.3	73.2±5.2
ITML	75.8±4.2	76.5±4.5	76.3±4.8
SVM	76.4±4.9	76.2±7.0	77.7±6.4
BBS	77.2±7.3	77.0±6.2	77.3±6.3
SLLC	70.5±7.2	75.9±4.5	75.8±4.8
LRML	74.7±6.2	75.3±5.9	75.8±5.2
JSL-15	**78.9**±6.7	**77.6**±5.5	77.7±6.4
JSL-all	78.2±7.3	76.6±5.8	**78.4**±6.7



(a) Ionosphere (b) Pima

Fig. 2: Average accuracy w.r.t. the number of labeled points with 15 landmarks.

References

1. M.-F. Balcan, A. Blum, and N. Srebro. Improved guarantees for learning via similarity functions. In *COLT*, pages 287–298. Omnipress, 2008.
2. J.-P. Bao, J.-Y. Shen, X.-D. Liu, and H.-Y. Liu. Quick asymmetric text similarity measures. In *ICMLC*, volume 1, pages 374–379, Nov 2003.
3. L. Baoli, L. Qin, and Y. Shiwen. An adaptive k-nearest neighbor text categorization strategy. *ACM TALIP*, 2004.
4. A. Bellet, A. Habrard, and M. Sebban. Similarity learning for provably accurate sparse linear classification. In *ICML*, pages 1871–1878, 2012.
5. A. Bellet, A. Habrard, and M. Sebban. A survey on metric learning for feature vectors and structured data. *arXiv preprint arXiv:1306.6709*, 2013.
6. A. Bellet, A. Habrard, and M. Sebban. *Metric Learning*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2015.
7. S. Boucheron, O. Bousquet, and G. Lugosi. Theory of classification : a survey of some recent advances. *ESAIM: Probability and Statistics*, 9:323–375, 2005.
8. O. Bousquet and A. Elisseeff. Stability and generalization. *JMLR*, 2:499–526, Mar. 2002.

9. J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning. In *ICML*, pages 209–216, New York, NY, USA, 2007. ACM.
10. M. Diligenti, M. Maggini, and L. Rigutini. Learning similarities for text documents using neural networks. In *ANNPR*, 2003.
11. Y. Freund and R. E. Schapire. Large margin classification using the perceptron algorithm. *Machine Learning*, 37(3):277–296, Dec. 1999.
12. M. Grabowski and A. Szałas. A technique for learning similarities on complex structures with applications to extracting ontologies. In *AWIC*, LNAI. Springer Verlag, 2005.
13. Z.-C. Guo and Y. Ying. Guaranteed classification via regularized similarity learning. *CoRR*, abs/1306.3108, 2013.
14. S. C. H. Hoi, W. Liu, and S.-F. Chang. Semi-supervised distance metric learning for collaborative image retrieval. In *CVPR*, 2008.
15. S. C. H. Hoi, W. Liu, and S.-F. Chang. Semi-supervised distance metric learning for collaborative image retrieval and clustering. *TOMCCAP*, 6(3), 2010.
16. A. Hust. Learning Similarities for Collaborative Information Retrieval. In *Machine Learning and Interaction for Text-Based Information Retrieval Workshop, TIR-04*, pages 43–54, 2004.
17. M. Ledoux and M. Talagrand. *Probability in Banach Spaces: Isoperimetry and Processes*. Springer, New York, May 1991.
18. M.-I. Nicolae, M. Sebban, A. Habrard, É. Gaussier, and M. Amini. Algorithmic robustness for learning via (ϵ, γ, τ)-good similarity functions. *CoRR*, abs/1412.6452, 2014.
19. G. Niu, B. Dai, M. Yamada, and M. Sugiyama. Information-theoretic semi-supervised metric learning via entropy regularization. In *ICML*. Omnipress, 2012.
20. A. M. Qamar and É. Gaussier. Online and batch learning of generalized cosine similarities. In *ICDM*, pages 926–931, 2009.
21. A. M. Qamar, É. Gaussier, J. Chevallet, and J. Lim. Similarity learning for nearest neighbor classification. In *ICDM*, pages 983–988, 2008.
22. S. Shalev-Shwartz, Y. Singer, and A. Y. Ng. Online and batch learning of pseudo-metrics. In *ICML*, New York, NY, USA, 2004. ACM.
23. K. Weinberger and L. Saul. Fast solvers and efficient implementations for distance metric learning. In *ICML*, pages 1160–1167. ACM, 2008.
24. K. Weinberger and L. Saul. Distance metric learning for large margin nearest neighbor classification. *JMLR*, 10:207–244, 2009.
25. E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell. Distance metric learning, with application to clustering with side-information. In *NIPS*, volume 15, pages 505–512, 2002.
26. H. Xu and S. Mannor. Robustness and generalization. In *COLT*, pages 503–515, 2010.
27. H. Xu and S. Mannor. Robustness and generalization. *Machine Learning*, 86(3):391–423, 2012.
28. Z.-J. Zha, T. Mei, M. Wang, Z. Wang, and X.-S. Hua. Robust distance metric learning with auxiliary knowledge. In *IJCAI*, pages 1327–1332, 2009.
29. J. Zhu, S. Rosset, T. Hastie, and R. Tibshirani. 1-norm support vector machines. In *NIPS*, page 16. MIT Press, 2003.