Algorithmic Robustness for Semi-Supervised (ϵ, γ, τ)-Good Metric Learning

Maria-Irina Nicolae[1,2], Marc Sebban[1], Amaury Habrard[1], Eric Gaussier[2], and Massih-Reza Amini[2]

[1] Université Jean Monnet, Laboratoire Hubert Curien, France
[2] Université Grenoble Alpes, CNRS-LIG/AMA, France

Abstract. Using the appropriate metric is crucial for the performance of most of machine learning algorithms. For this reason, a lot of effort has been put into distance and similarity learning. However, it is worth noting that this 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] provides means to relate 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. We furthermore prove the robustness of our algorithm, which allows us to provide a generalization bound for this approach. The behavior of our method is illustrated via some experimental results.

1 Introduction

The importance of the underlying geometry of the data for improving the performance of learning algorithms has determined the expansion of a new research area termed *metric learning* [5]. From the point of view of the metric, most of these approaches focus on distance learning [3,6,7,14,16], but similarity learning has also attracted a growing interest [2,8,11,13], as the cosine similarity is more appropriate for certain problems than the euclidean distance. More recently, [1] have proposed the first framework that formalizes the relation between the quality of a metric and that of a classification algorithm making use of them. This broad framework, that can be used with a large range of similarity functions, provides generalization guarantees on a linear classifier learned from the similarity. However, to enjoy these guarantees, the similarity function is assumed to be known beforehand and to satisfy (ϵ, γ, τ)-goodness properties. The main limitation is that [1] does not provide any algorithm for learning such similarities.

In order to complete this framework, [4] have developped a method that independently learns an (ϵ, γ, τ)-good similarity. It is then plugged into the initial algorithm [1] to learn the linear separator using the metric. However, the similarity learning step is done in a completely supervised way while the (ϵ, γ, τ)-good framework opens the door to the use of unlabeled data.

In this paper, our objective is to jointly learn the metric and the classifier in the theoretical framework of (ϵ, γ, τ)-good similarities. Furthermore, and unlike [4], the whole process is done in a semi-supervised way. To our knowledge, joint learning has not been explored before for semi-supervised metric learning. Enforcing (ϵ, γ, τ)-goodness allows us to preserve the theoretical guarantees from [1]. Lastly, proving the algorithmic robustness [17] of our method leads to consistency bounds for different types of similarity functions.

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

2 Notations and Related Work

In our developments, vectors are denoted by lower-case bold symbols ($\mathbf{x}$) and matrices by upper-case bold symbols ($\mathbf{A}$). A pairwise similarity function over $\mathcal{X} \subseteq \mathcal{R}^d$ is defined as $K : \mathcal{X} \times \mathcal{X} \to [-1, 1]$, and the hinge loss as $[c]_+ = \max(0, 1 - c)$. We note the L_1 norm by $|| \cdot ||_1$ and the L_2 norm by $|| \cdot ||_2$. The purpose of metric learning is to learn the parameters of a distance or similarity function that best fits the underlying geometry of the data. The learning is usually done using side information, 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 $\mathbf{x}''$). The metric is commonly represented by a matrix of values resulting from solving an optimization problem.

Most of state-of-the-art approaches focus on learning a Mahalanobis distance, defined as $d_{\mathbf{A}}(\mathbf{x}, \mathbf{x}') = \sqrt{(\mathbf{x} - \mathbf{x}')^T \mathbf{A} (\mathbf{x} - \mathbf{x}')}$. The distance is parameterized by the symmetric and positive semi-definite (PSD) matrix $\mathbf{A} \in \mathbb{R}^{d \times d}$. This metric implicitly corresponds to computing the Euclidean distance after linearly projecting the data to a different feature space. The PSD constraint on $\mathbf{A}$ ensures $d_{\mathbf{A}}$ is a proper metric. Setting $\mathbf{A}$ to the identity matrix gives the Euclidean distance. In this context, LMNN [16] is one of the most widely-used Mahalanobis distance learning methods. The constraints 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. Workarounds include using a specific solver or opting for information-theoretic approaches. ITML [6] was the first method to use LogDet divergence for regularization, providing an easy way for ensuring that $\mathbf{A}$ is a PSD matrix. However, the learned metric $\mathbf{A}$ is strongly influenced by the initial value $\mathbf{A}_0$, which is an important shortcoming, as $\mathbf{A}_0$ is handpicked. LRML [10] learns Mahalanobis distances with

manifold regularization using a Laplacian matrix in a semi-supervised setting. It performs particularly well compared to fully supervised methods when side information is scarce.

More generally, Mahalanobis distance learning faces two main limitations: firstly, enforcing the PSD and symmetry constraints on $\mathbf{A}$ is costly and often rules out natural similarity functions; secondly, although state-of-the-art Mahalanobis distance learning methods yield better accuracy than using the Euclidean distance, no theoretical guarantees are provided to establish a link between the quality of the metric and that of the classifier that makes use of it. [1] defined the (ϵ, γ, τ)-good similarity functions based on non PSD matrices, which uses similarities between labeled data and unlabeled reasonable points (roughly speaking, the reasonable points play the same role as that of support vectors in SVMs). Their theory was the first stone to establish generalization guarantees for a linear classifier that would be learned by making use of such similarities. Their results are derived based on the definition of a good similarity function for a given problem: considering a set of "reasonable points", a $(1-\epsilon)$ proportion of examples $\mathbf{x}$ are 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. For this, the proportion of reasonable points from the sample must be greater than τ. In their definition, the margin violation is averaged over all reasonable points which leads to a more flexible setting than pair- or triplet-based constraints. If K is (ϵ, γ, τ)-good and enough reasonable points are available, there exists a linear separator $\boldsymbol{\alpha}$ with error arbitrarily close to ϵ in the space $\phi^{\mathcal{S}}$. Finding the separator is done by solving the following optimization problem:

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

The previous problem can be solved efficiently by linear programming. Also, tuning the value of γ (L_1 constraint) will produce a sparse solution. The main limitation of this approach is that the similarity function K is considered known.

This limitation has been partly overcome by SLLC [4] by optimizing the (ϵ, γ, τ)-goodness of a bilinear similarity function under Frobenius norm regularization. The learned metric is then used to build a global linear classifier with guarantees. Moreover, a bound on the generalization error of the associated classifier through uniform stability can be obtained. More recently, [9] derived generalization bounds for similarity learning formulations that are regularized with more general matrix-norms, based on the Rademacher complexity and Khinchin-type inequalities.

There are three main distinctions between these approaches and our work. Firstly, 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. Secondly, our setting uses the algorithmic robustness to establish bounds, which enables us to characterize our algorithm by exploiting the geometry of the data; that is not the case with the Rademacher complexity. Lastly, regularization is integrated

through constraints in our setting, as explained in the following sections, which leads to a formulation with less hyperparameters.

3 Learning Consistent Good Similarity Functions

In this section, we present our semi-supervised framework for jointly learning a similarity function and a linear separator from data. We also provide a generalization bound for our approach based on the recent algorithmic robustness framework [17]. We end this section by presenting some particular similarity functions that can be used in our setting.

3.1 Optimization Problem

Let $\mathcal{S}$ be a sample set of d_l labeled examples $(\mathbf{x}, l(\mathbf{x})) \in \mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ ($\mathcal{X} \subseteq \mathcal{R}^d$) and d_u unlabeled examples. We assume that $\mathcal{X}$ is bounded, which can be expressed, after normalization, by $||\mathbf{x}||_2 \leq 1$. 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 want to optimize the goodness of $K_{\mathbf{A}}$ w.r.t. the empirical loss of a finite sample. To this end, we must find the matrix $\mathbf{A}$ and the global separator $\boldsymbol{\alpha} \in \mathbb{R}^{d_u}$ that minimize the loss function (in ¡our case, the hinge loss) over the training set $\mathcal{S}$. Our learning algorithm takes the form of the following constrained optimization problem.

$$\min_{\boldsymbol{\alpha}, \mathbf{A}} \quad \frac{1}{d_l} \sum_{i=1}^{d_l} \left[1 - \sum_{j=1}^{d_u} \alpha_j l(\mathbf{x}_i) K_{\mathbf{A}}(\mathbf{x}_i, \mathbf{x}_j) \right]_+ \tag{1}$$

$$\text{s.t.} \sum_{j=1}^{d_u} |\alpha_j| \leq 1/\gamma \tag{2}$$

$$\mathbf{A} \text{ diagonal}, \quad |A_{kk}| \leq 1, \quad 1 \leq k \leq d, \tag{3}$$

The novelty of this algorithm is the *joint optimization* over $\mathbf{A}$ and $\boldsymbol{\alpha}$: by solving problem (1), we are learning the metric and the separator at the same time. A significant advantage of this formulation 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 many 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 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 (2) takes into account the desired margin γ and is the same as in [1]. The new Constraint (3) serves two purposes: first, it restricts the similarity

$K_{\mathbf{A}}$, thus preserving its (ϵ, γ, τ)-goodness; second, as it bounds the values in the matrix $\mathbf{A}$, it limits the risk of overfitting, and thus plays the role of regularization without imposing sparsity. Regularizing metrics through standard L_1 or $L_{(1,2)}$ norms would slowly push the values in the matrix towards zero, which is not necessarily desirable. Indeed, let $f(\mathbf{x}) = \sum_{j=1}^{d_u} \alpha_j K_{\mathbf{A}}(\mathbf{x}, \mathbf{x}_j)$ be the output of the linear separator w.r.t. $\mathbf{x}$. For some linear similarities $K_{\mathbf{A}}(x, x')$, such as the bilinear form $K_{\mathbf{A}}(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T \mathbf{A} \mathbf{x}'$, computing $f(\mathbf{x})$ boils down to calculating the similarity between $\mathbf{x}$ and the barycenter of the (weighted) unlabeled points, making sparsity superfluous.

3.2 Consistency Guarantees

We now present a theoretical analysis of our approach. For the purpose of discussing the algorithmic robustness of the method, let us rewrite the minimization problem (1) with a more generalized notation of the loss function:

$$\min \frac{1}{d_l} \sum_{i=1}^{d_l} \ell(\mathbf{A}, \boldsymbol{\alpha}, \mathbf{z}_i = (\mathbf{x}_i, l(\mathbf{x}_i))),$$

where $\ell(\mathbf{A}, \boldsymbol{\alpha}, \mathbf{z}_i = (\mathbf{x}_i, l(\mathbf{x}_i))) = \left[1 - \sum_{j=1}^{d_u} \alpha_j l(\mathbf{x}_i) K_{\mathbf{A}}(\mathbf{x}_i, \mathbf{x}_j)\right]_+$ is the instantaneous loss estimated at point $(\mathbf{x}_i, l(\mathbf{x}_i))$. Therefore, the optimization problem (1) under constraints (2) and (3) reduces to minimizing the empirical loss $\hat{\mathcal{R}}^{\ell} = \frac{1}{d_l} \sum_{i=1}^{d_l} \ell(\mathbf{A}, \boldsymbol{\alpha}, \mathbf{z}_i)$ over the training set $\mathcal{S}$. To begin with, let us recall the notion of robustness of an algorithm $\mathcal{A}$.

Definition 1 (Algorithmic Robustness [17]). *Algorithm $\mathcal{A}$ is $(M, \epsilon(\cdot))$-robust, for $M \in \mathbb{N}$ and $\epsilon(\cdot) : \mathcal{Z}^{d_l} \to \mathbb{R}$, if $\mathcal{Z}$ can be partitioned into M disjoint sets, denoted by $\{C_i\}_{i=1}^{M}$, such that the following holds for all $\mathcal{S} \in \mathcal{Z}^{d_l}$:*

$$\forall \mathbf{z} = (\mathbf{x}, l(\mathbf{x})) \in \mathcal{S}, \forall \mathbf{z}' = (\mathbf{x}', l(\mathbf{x}')) \in \mathcal{Z}, \forall i \in [M] :$$
$$\text{if } \mathbf{z}, \mathbf{z}' \in C_i, \text{ then } |\ell(\mathbf{A}, \boldsymbol{\alpha}, \mathbf{z}) - \ell(\mathbf{A}, \boldsymbol{\alpha}, \mathbf{z}')| \leq \epsilon(\mathcal{S}).$$

Roughly speaking, an algorithm is robust if for any example $\mathbf{z}'$ falling in the same subset as a training example $\mathbf{z}$, the gap between the losses associated with $\mathbf{z}$ and $\mathbf{z}'$ is bounded. Subsets are constructed using a partitioning of $\mathcal{Z}$ based on covering numbers [12]. Two examples are close if they belong to the same region, implying that the norm between them is lesser than a fixed quantity ρ (see [17] for details about building the covering). Now we can state the first theoretical contribution of this paper.

Theorem 1. *Given a partition of $\mathcal{Z}$ into M subsets $\{C_i\}$ such that $\mathbf{z} = (\mathbf{x}, l(\mathbf{x}))$ and $\mathbf{z}' = (\mathbf{x}', l(\mathbf{x}')) \in C_i$ and $l(\mathbf{x}) = l(\mathbf{x}')$, and provided that $K_{\mathbf{A}}(\mathbf{x}, \mathbf{x}')$ is l-lipschitz w.r.t. its first argument, the optimization problem* (1) *with constraints* (2) *and* (3) *is $(M, \epsilon(\mathcal{S}))$-robust with $\epsilon(\mathcal{S}) = \frac{1}{\gamma} l \rho$, where $\rho = \sup_{\mathbf{x}, \mathbf{x}' \in C_i} ||\mathbf{x} - \mathbf{x}'||$.*

Proof.

$$\big|\ell(\mathbf{A},\boldsymbol{\alpha},\mathbf{z})-\ell(\mathbf{A},\boldsymbol{\alpha},\mathbf{z}')\big| \leq \big|\sum_{j=1}^{d_u}\alpha_j l(\mathbf{x}')K_{\mathbf{A}}(\mathbf{x}',\mathbf{x}_j)-\sum_{j=1}^{d_u}\alpha_j l(\mathbf{x})K_{\mathbf{A}}(\mathbf{x},\mathbf{x}_j)\big| \quad (4)$$

$$=\big|\sum_{j=1}^{d_u}\alpha_j(K_{\mathbf{A}}(\mathbf{x}',\mathbf{x}_j)-K_{\mathbf{A}}(\mathbf{x},\mathbf{x}_j))\big| \leq \sum_{j=1}^{d_u}|\alpha_j|\cdot\big|K_{\mathbf{A}}(\mathbf{x}',\mathbf{x}_j)-K_{\mathbf{A}}(\mathbf{x},\mathbf{x}_j)\big| \quad (5)$$

$$\leq \sum_{j=1}^{d_u}|\alpha_j|\cdot l||\mathbf{x}-\mathbf{x}'|| \leq \frac{1}{\gamma}l\rho \quad (6)$$

Setting $\rho = \sup_{\mathbf{x},\mathbf{x}'\in C_i}||\mathbf{x}-\mathbf{x}'||_1$, we get the Theorem. We get Inequality (4) from the 1-lipschitzness of the hinge loss; Inequality (5) comes from triangle inequality; the first inequality on line (6) is due to the l-lipschitzness of $K_{\mathbf{A}}(\mathbf{x},\mathbf{x}_j)$ w.r.t. its first argument, and the result follows from Condition (2). □

We now give a PAC generalization bound on the true loss making use of the previous robustness result. Let $\mathcal{R}^\ell = \mathbb{E}_{\mathbf{z}\sim\mathcal{Z}}\ell(\mathbf{A},\boldsymbol{\alpha},\mathbf{z})$ be the true loss w.r.t. the unknown distribution $\mathcal{Z}$ and $\hat{\mathcal{R}}^\ell = \frac{1}{d_l}\sum_{i=1}^{d_l}\ell(\mathbf{A},\boldsymbol{\alpha},\mathbf{z}_i)$ be the empirical loss over the training set $\mathcal{S}$. We have the following concentration inequality that allows one to capture statistical information coming from the different regions of the partition of $\mathcal{Z}$.

Proposition 1. *[15] Let $(|N_1|,\ldots,|N_M|)$ be an i.i.d. multinomial random variable with parameters $d_l = \sum_{i=1}^M|N_i|$ and $(p(C_1),\ldots,p(C_M))$. By the Bretagnolle-Huber-Carol inequality we have:* $\Pr\left\{\sum_{i=1}^M\big|\frac{|N_i|}{d_l}-p(C_i)\big|\geq\lambda\right\}\leq 2^M\exp\big(\frac{-d_l\lambda^2}{2}\big)$, *hence with probability at least $1-\delta$, $\sum_{i=1}^M\big|\frac{N_i}{d_l}-p(C_i)\big|\leq\sqrt{\frac{2M\ln 2+2\ln(1/\delta)}{d_l}}$.*

We are now able to present our generalization bound in the following theorem.

Theorem 2. *Considering that problem (1) is $(M,\epsilon(\mathcal{S}))$-robust, and that $K_{\mathbf{A}}$ is l-lipschitz w.r.t. to its first argument, for any $\delta>0$ with probability at least $1-\delta$, we have:*

$$|\mathcal{R}^\ell-\hat{\mathcal{R}}^\ell|\leq\tfrac{1}{\gamma}l\rho+B\sqrt{\tfrac{2M\ln 2+2\ln(1/\delta)}{d_l}},$$

where $B=1+\frac{1}{\gamma}$ is an upper bound of the loss ℓ.

The proof of Theorem 2 follows the one described in [17]. Note that the cover radius ρ can be arbitrarily small at the expense of larger values of M. As M appears in the second term, decreasing to 0 when d_l tends to infinity, this bound provides a standard $O(1/\sqrt{d_l})$ asymptotic convergence.

As one can note, our main theorems strongly depend on the l-lipschitzness of the similarity function. We provide below three standard similarity functions together with their lipschitz property. $K^1_{\mathbf{A}}$ and $K^2_{\mathbf{A}}$ are linear w.r.t. their arguments, and have the advantage of keeping Problem (1) convex. $K^3_{\mathbf{A}}$ is gaussian-like kernel based on the Mahalanobis distance, and is non linear.

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

Ex. 1 Let $K^1_{\mathbf{A}}$ be the bilinear form $K^1_{\mathbf{A}}(\mathbf{x}, \mathbf{x}') = \mathbf{x}^T\mathbf{A}\mathbf{x}'$. $K^1_{\mathbf{A}}(\mathbf{x}, \mathbf{x}')$ is 1-lipschitz w.r.t. its first argument.

Ex. 2 We define $K^2_{\mathbf{A}}(\mathbf{x}, \mathbf{x}') = 1 - (\mathbf{x} - \mathbf{x}')^T\mathbf{A}(\mathbf{x} - \mathbf{x}')$, a similarity derived from the Mahalanobis distance. $K^2_{\mathbf{A}}(\mathbf{x}, \mathbf{x}')$ is 4-lipschitz w.r.t. its first argument.

Ex. 3 Let $K^3_{\mathbf{A}}(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{(\mathbf{x}-\mathbf{x}')^T\mathbf{A}(\mathbf{x}-\mathbf{x}')}{2\sigma^2}\right)$. $K^3_{\mathbf{A}}(\mathbf{x}, \mathbf{x}')$ is l-lipschitz w.r.t. its first argument with $l = \frac{2}{\sigma^2}\left(\exp\left(\frac{1}{2\sigma^2}\right) - \exp\left(\frac{-1}{2\sigma^2}\right)\right)$.

Plugging $l = 1$ (resp. $l = 4$ and $l = \frac{2}{\sigma^2}\left(\exp\left(\frac{1}{2\sigma^2}\right) - \exp\left(\frac{-1}{2\sigma^2}\right)\right)$) in Theorem 2, we obtain consistency results for Problem (1) using $K^1_{\mathbf{A}}(\mathbf{x}, \mathbf{x}')$ (resp. $K^2_{\mathbf{A}}(\mathbf{x}, \mathbf{x}')$ and $K^3_{\mathbf{A}}(\mathbf{x}, \mathbf{x}')$). As the gap between empirical and true loss presented in Theorem 2 is proportional with the l-lipschitzness of each similarity function, we would like to keep this parameter as small as possible. We notice that the generalization bound is tighter for $K^1_{\mathbf{A}}$ than for $K^2_{\mathbf{A}}$. The bound for $K^3_{\mathbf{A}}$ depends on the additional parameter σ, that adjusts the influence of the similarty value w.r.t. the distance to the landmarks.

4 Experiments

Metric learning state-of-the-art algorithms are mostly designed for a supervised setting, and usually optimize a metric for kNN classification. It is thus difficult to propose a totally fair comparative study. We compare our method (JSL – Joint Similarity Learning) with algorithms from different categories (supervised, kNN-oriented). The experimental study is conducted on 7 classic datasets taken from the UCI Machine Learning Repository (Table 1).

4.1 Experimental Setup

For a complete comparison, we analyse two main families of approaches: first, linear classifiers, for which we consider BBS [1], SLLC [4], linear SVM with L_2 regularization and our method, JSL; second, nearest neighbor approaches: ITML [6], LMNN [16] and LRML [10], for which we report accuracies for 3NN classification. All attributes are centered around zero and scaled to ensure $||\mathbf{x}||_2 \leq 1$. 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. We illustrate the classification using a restricted quantity of labeled data by limiting the number of labeled points 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 equal to the size of the training set, either set to 15 points (corresponding to

Table 2: Average accuracy (%) with conf. interval at 95%, 5 labeled points/class.

Lmks.	Sim.	Balance	Ionosphere	Iris	Liver	Pima	Sonar	Wine
all pts.	$K^1_{\mathbf{A}}$	85.7±3.5	88.5±2.6	**74.5**±4.4	63.9±5.3	71.1±3.8	**72.3**±4.1	**87.7**±5.0
	$K^2_{\mathbf{A}}$	**87.1**±2.5	**91.0**±2.0	71.4±5.9	**69.2**±3.2	**72.9**±3.9	71.9±4.2	84.2±6.9
	$K^3_{\mathbf{A}}$	81.1±8.5	86.2±2.8	68.2±8.5	58.6±6.3	71.1±4.3	63.9±10.0	83.5±6.2
15 pts.	$K^1_{\mathbf{A}}$	84.9±2.6	**86.7**±1.6	**75.5**±2.3	63.1±5.9	71.1±4.1	72.9±4.6	87.3±5.5
	$K^2_{\mathbf{A}}$	**87.5**±2.7	85.0±3.8	74.1±6.3	**67.3**±4.3	**74.3**±4.1	**77.4**±6.3	76.9±10.5
	$K^3_{\mathbf{A}}$	79.6±10.0	76.3±7.4	72.7±6.3	59.6±6.0	69.0±8.6	68.7±10.0	**88.5**±5.0

k-means++ cluster centroids). When all the available data is used as landmarks, the L_1 constraint on $\boldsymbol{\alpha}$ forces the algorithm to choose the most valuable of them by adapting their respective weights. 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: $\gamma \in \{10^{-4}, \dots, 10^{-1}\}$ for BBS and JSL, $\lambda_{ITML} \in \{10^{-4}, \dots, 10^{4}\}$, $\gamma_{SLLC}, \beta_{SLLC} \in \{10^{-7}, \dots, 10^{-2}\}$, $\lambda_{SLLC} \in \{10^{-3}, \dots, 10^{2}\}$, while for LRML we consider $\gamma_s, \gamma_d, \gamma_i \in \{10^{-2}, \dots, 10^{2}\}$. For LMNN, we set $\mu = 0.5$, as done in [16]. We solve BBS and JSL using projected gradient descent. In JSL, we alternate the optimization between $\boldsymbol{\alpha}$ and $\mathbf{A}$.

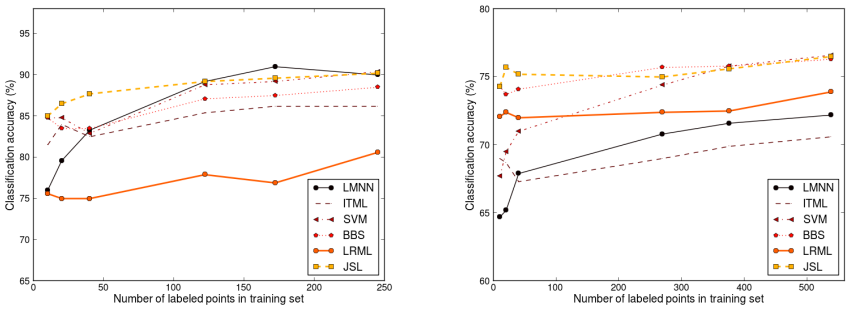
4.2 Results

Choice of similarity We first study the influence of the similarity function on the proposed framework. We plug into JSL the three similarities studied previously (see Section 3.2) and present the results for classification in Table 2. For both unlabeled configurations, 15 points or the whole training set, $K^2_{\mathbf{A}}$ yields the best results on 4 out of 7 datasets, while $K^3_{\mathbf{A}}$ performs best in only one case. We explain this by the topology of the involved datasets, which make the Mahalanobis distance a better dicriminant for classification than the other similarities. In the case of $K^3_{\mathbf{A}}$, there is a trade-off between the tightness of the bound in Theorem 2 and the stability of the results. Large values of σ will lead to tighter bounds (as l is smaller), but the resulting similarity function becomes linear and less discriminative. As a consequence, the results vary more for this similarity function, leading to larger confidence intervals, as can be seen on almost all the collections. When comparing the two unlabeled settings, we notice that there are only a few cases when the best accuracy is attained with less unlabeled points for the same similarity, but that when this happens the improvement is significant. This is due to the fact that the 15 unlabeled points are not chosen randomly, but contain relevant information w.r.t. data topology.

Comparison of different methods Following the previous analysis, we now propose to study the classification performance of our method. For this purpose, we focus on JSL with $K^2_{\mathbf{A}}$ using 15 unlabeled landmarks. We compare our approach to state-of-the-art methods when a limited amount of labeled data is used and present the results in Table 3. In order to ensure fairness, we fix the similarity function in BBS to the Euclidean distance. On average over all datasets, JSL obtains the best performance in all the settings. The only other

Table 3: Average accuracy (%) over all datasets with confidence interval at 95%.

Data	LMNN-dg	LMNN	ITML	SVM	BBS	SLLC	LRML	JSL
5 pts./cl.	65.1±5.5	69.4±5.9	75.8±4.2	76.4±4.9	77.2±7.3	70.5±7.2	74.7±6.2	**78.4**±2.3
10 pts./cl.	68.2±5.6	70.9±5.3	76.5±4.5	76.2±7.0	77.0±6.2	75.9±4.5	75.3±5.9	**78.7**±1.9
20 pts./cl.	71.5±5.2	73.2±5.2	76.3±4.8	77.7±6.4	77.3±6.3	75.8±4.8	75.8±5.2	**78.3**±1.6



(a) Ionosphere (b) Pima

Fig. 1: Average accuracy w.r.t. the number of labeled points with 15 landmarks.

methods with comparable results are BBS and SVM. We mention that JSL using also $K^2_{\mathbf{A}}$ and all the training set as unlabeled landmarks performs similarly to the setting presented in Table 3. This result proves that we can learn well with a small amount of both labeled and unlabeled data, when the unlabeled points are informative (*e.g.*, correspond to cluster centroids, as it is the case here).

Quantity of labeled data We now study the method's behavior when the level of supervision varies. For this we keep on using JSL with $K^2_{\mathbf{A}}$ and set the number of unlabeled points to 15. Figure 1 presents the accuracies on two representative datasets, Ionosphere and Pima, with an increasing number of labeled examples. JSL obtains the best performance in both cases when less than 50% of the labeled data is used, which is coherent with the results presented in Table 3. For greater amounts of data, JSL performs similarly to the best state-of-the-art methods: SVM and LMNN for Ionosphere, and SVM and BBS for Pima; these results also correspond to those presented in the previous subsection.

5 Conclusion

In this paper, we extend the (ϵ, γ, τ)-good similarity 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. We show that our joint approach is theoretically founded using results from [1] and new results based on algorithmic robustness. The approach we propose is particularly adapted to learning with small amounts of both labeled and unlabeled data, when the unlabeled points are informative. This is revealed in the experiments conducted which illustrate the good behavior of our method in the above setting on various UCI datasets,

in comparison with different standard approaches (LMNN, ITML, SVM, BBS, SLLC, LRML).

References

1. Balcan, M.F., Blum, A., Srebro, N.: Improved guarantees for learning via similarity functions. In: COLT. pp. 287–298. Omnipress (2008)
2. Bao, J.P., Shen, J.Y., Liu, X.D., Liu, H.Y.: Quick asymmetric text similarity measures. ICMLC (2003)
3. Baoli, L., Qin, L., Shiwen, Y.: An adaptive k-nearest neighbor text categorization strategy. ACM TALIP (2004)
4. Bellet, A., Habrard, A., Sebban, M.: Similarity learning for provably accurate sparse linear classification. In: ICML 2012. pp. 1871–1878 (2012)
5. Bellet, A., Habrard, A., Sebban, M.: A survey on metric learning for feature vectors and structured data. arXiv preprint arXiv:1306.6709 (2013)
6. Davis, J.V., Kulis, B., Jain, P., Sra, S., Dhillon, I.S.: Information-theoretic metric learning. In: ICML. pp. 209–216. ACM, New York, NY, USA (2007)
7. Diligenti, M., Maggini, M., Rigutini, L.: Learning similarities for text documents using neural networks. In: ANNPR (2003)
8. Grabowski, M., Szałas, A.: A technique for learning similarities on complex structures with applications to extracting ontologies. In: AWIC. LNAI, Springer (2005)
9. Guo, Z.C., Ying, Y.: Guaranteed classification via regularized similarity learning. CoRR abs/1306.3108 (2013)
10. Hoi, S.C.H., Liu, W., Chang, S.F.: Semi-supervised distance metric learning for collaborative image retrieval and clustering. TOMCCAP 6(3) (2010)
11. Hust, A.: Learning Similarities for Collaborative Information Retrieval. In: Machine Learning and Interaction for Text-Based Information Retrieval Workshop, TIR-04. pp. 43–54 (2004)
12. Kolmogorov, A., Tikhomirov, V.: ϵ-entropy and ϵ-capacity of sets in functional spaces. American Mathematical Society Translations 2(17), 277–364 (1961)
13. Qamar, A.M., Gaussier, É.: Online and batch learning of generalized cosine similarities. In: ICDM. pp. 926–931 (2009)
14. Shalev-Shwartz, S., Singer, Y., Ng, A.Y.: Online and batch learning of pseudo-metrics. In: ICML. ACM, New York, NY, USA (2004)
15. van der Vaart, A., Wellner, J.: Weak Convergence and Empirical Processes. Springer series in statistics, Springer (1996)
16. Weinberger, K., Saul, L.: Distance metric learning for large margin nearest neighbor classification. JMLR 10, 207–244 (2009)
17. Xu, H., Mannor, S.: Robustness and generalization. In: COLT. pp. 503–515 (2010)