Metric Learning as Convex Combinations of Local Models with Generalization Guarantees

Valentina Zantedeschi, Rémi Emonet, Marc Sebban
firstname.lastname@univ-st-etienne.fr
Univ Lyon, UJM-Saint-Etienne, CNRS, Institut d Optique Graduate School,
Laboratoire Hubert Curien UMR 5516, F-42023, SAINT-ETIENNE, France

Abstract

Over the past ten years, metric learning allowed the improvement of numerous machine learning approaches that manipulate distances or similarities. In this field, local metric learning has been shown to be very efficient, especially to take into account non linearities in the data and better capture the peculiarities of the application of interest. However, it is well known that local metric learning (i) can entail overfitting and (ii) face difficulties to compare two instances that are assigned to two different local models. In this paper, we address these two issues by introducing a novel metric learning algorithm that linearly combines local models (C2LM). Starting from a partition of the space in regions and a model (a score function) for each region, C2LM defines a metric between points as a weighted combination of the models. A weight vector is learned for each pair of regions, and a spatial regularization ensures that the weight vectors evolve smoothly and that nearby models are favored in the combination. The proposed approach has the particularity of working in a regression setting, of working implicitly at different scales, and of being generic enough so that it is applicable to similarities and distances. We prove theoretical guarantees of the approach using the framework of algorithmic robustness. We carry out experiments with datasets using both distances (perceptual color distances, using Mahalanobis-like distances) and similarities (semantic word similarities, using bilinear forms), showing that C2LM consistently improves regression accuracy even in the case where the amount of training data is small.

1. Introduction

In many machine learning tasks, like classification, clustering or ranking, decisions are based on distance or similarity functions. In order to capture the peculiarities of the data of the applications at hand, a lot of work has gone during the past ten years into automatically optimizing those functions, topic referred to as *metric learning* [9, 3, 4]. Most of the time, a unique *global* metric is learned over the input space, typically taking the form of a (linear) geometric transformation. This is the case for most of the Mahalanobis-like metric learning approaches, such as LMNN [22] or ITML [5]. However, it turns out that for data that present multi-modalities and/or non-linearities, *local* metric learning has been shown to be very efficient because of its flexibility to capture well geometric variations of the input space. On the other hand, a major problem of local metric learning is that it can entail overfitting. Some recent solutions have been proposed based on feature space dimensionality reduction [8], manifold regularization [21] or generative models [15]. However, those approaches mainly focus on improving the results locally, i.e. while comparing instances of the "same region" of the input space. Therefore, they are not suited to compare points far from each other. This limitation is illustrated in Figure 1.

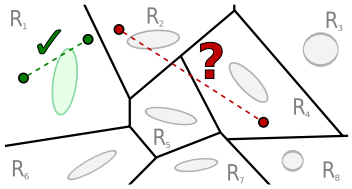


Figure 1: Limitation of local metric learning: While two points belonging to the same region (e.g. in R_1) can be managed by the corresponding locally-learned metric (depicted as an ellipse), two points from different regions (e.g. in R_2 and R_4) cannot be accurately compared using a single local metric.

One of the main objectives of our paper is to address this pitfall by learning convex combinations of local metrics that are not only good locally, but also globally relevant. Our algorithm, called Convex Combinations of Local Models (C2LM), basically optimizes for any pair of regions a vector of weights corresponding to the contribution of each local model while computing the distance or similarity between

two points of those regions (see Figure 2). By means of manifold and vector similarity regularization, we constrain the convex combinations to reflect the topological characteristics of the input space and to vary smoothly. Since our main aim is to learn the influence of each local metric, we will assume in the rest of this paper that the input space has been previously partitioned into regions and that on each region a local metric has been learned to express its underlying geometry.

Our approach has another particularity: unlike the current trend in metric learning, it lies in a regression setting rather than in a classification framework. Indeed, it is worth noticing that most metric learning methods use side information brought by pairs of training examples in the form of must-link/cannot-link constraints (also called positive/negative pairs) or relative constraints (also called training triplets). A metric learning method typically aims to optimize the parameters of the metric such that it best agrees with those constraints. It turns out that in some applications, the side information provided by the problem of interest simply relies on pairs of examples associated to a target score of (dis)similarity. This is the case in color distance perception (that will constitute one of our two series of experiments), where training data take the form of pairs of color patches and their reference perceptual distance ΔE_{00} [18]. This is also the case for databases made of pairs of strings and their corresponding semantic distance (see, e.g., the well known WordSim353 dataset[1]). A last example comes from temporal sequence alignments, where training data can be made of pairs of acoustic signals and their corresponding optimal alignment (e.g. see [10]). In such contexts, state of the art metric learning algorithms face difficulties to accurately capture the idiosyncrasies of the data. Indeed, the price to pay often implies a dramatic increase of the number of constraints to satisfy. Here, we overcome this issue by dealing with metric learning in a regression setting that allows us to directly fit the target scores.

When proposing a new algorithm for metric learning, it is fundamental to prove that it is theoretically well-founded. In this paper, a lot of work has gone into deriving theoretical guarantees for our method through the algorithmic robustness framework introduced in [25]. We show that this setting is particularly adapted to our framework because it is based on a partition of the input space as we defined it for our problem (see Section 3).

To recapitulate, our contributions are three fold:

1. We improve local metrics by learning linear combinations of local models that (i) allow one to accurately compare any pair of points, (ii) guarantees a certain continuity of the distances in the entire input space, and (iii) do not overfit;
2. We develop our metric learning approach in a regression setting that is not usual in this field;
3. We derive theoretical guarantees for our method through the algorithmic robustness framework.

[1] http://alfonseca.org/eng/research/wordsim353.html

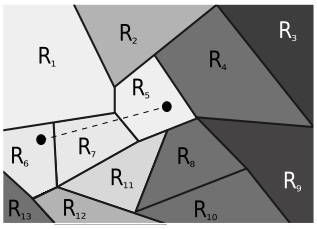


Figure 2: Illustration of the influence of the local models based on region distances: the more influent a local metric for the learned metric, the lighter the color of the associated region. For example, the local models of regions R_6, R_5, R_7, R_1 and R_{11} are more influent than those of the other regions, while computing the distance between the two points of regions R_6 and R_5.

The remainder of this paper is organized as follows: in Section 2, we introduce a short state of the art on metric learning; Section 3 is devoted to the presentation of our algorithm for which, in Section 4, we derive a generalization bound based on algorithmic robustness; In order to show that C2LM is able to deal with not only distance functions but also similarity functions, we instantiate the local models as Mahalanobis-like distances and as bilinear similarities; Lastly, Section 5 is dedicated to the experiments. We conduct two series of experiments: a first one in color distance perception, and a second one in string semantic similarities.

2. Related Work

A classic metric learning approach consists in learning a unique Mahalanobis-like metric of the form $d_A(x_1, x_2) = \sqrt{(x_1 - x_2)^T A (x_1 - x_2)}$, with A positive semi-definite ($A \succeq 0$) [4]. If $A = I$, the metric is an Euclidean distance. If a Cholesky decomposition is applied to A (then $A = L^T L$), the distance function corresponds to computing an Euclidean distance in a new space, where the data are linearly rescaled. For instance, the authors of [24], using pair-wise information, learn a metric that minimizes the distance between similar examples and maximizes the distance between dissimilar ones and show that it improves results in clustering tasks. Other common metric learning frameworks are LMNN (Large-Margin Nearest Neighbors) proposed in [22] for improving k-nearest neighbor (kNN) classification and ITML (Information-Theoric Metric Learning) introduced in [5] for handling constraints and prior knowledge on the metric by means of the LogDet regularization.

On the other hand, a global and linear metric may not necessarily perform well for all problems, especially for data that present multimodality and non-linearities. In those cases, non-linear methods are more suitable, such as kernel learning and local metric learning approaches. For instance, in [23], Weinberger et al. have shown that learning simultaneously a set of local metrics, one for each region of the input space or class label, improved their LMNN framework.

If local metric learning approaches can adapt well to variations on the input space, they are also quite sensitive to overfitting, especially when local metrics are learned independently from each other. In order to overcome this problem, linear or non-linear combinations of local metrics (instead of only one metric) or kernels (see Multiple Kernel Learning [1] and [7]) can be used to compare instances and auxiliary information can be taken into account by means of regularization terms. For instance, the authors of [21] proposed a regularization based on the geometric characteristics of the instance space: they learn jointly linear combinations of basis metrics (one local metric per region and one linear combination per input instance) and constrain them to vary smoothly over the instances. The weight vectors of close instances are then similar and reflect the geometric characteristics of the input space. However, the learned metrics are no longer symmetric and they are accurate only when comparing instances relatively close to each other. Another example of regularization is proposed in [8], where the authors control the rank of the matrix of the learned combinations of metrics, i.e. the total number of parameters of the problem. Doing so, they penalize too complex solutions, which are probably too specialized to the training instances and have lost generalization power on unseen instances. Their approach is based on the pair-wise information about the similarity between instances and the geometric structure of the input space is not taken into account.

Both frameworks [21, 8] are not suited for regression tasks and their choice of defining a linear combination of metrics for each input instance affects the complexity of their problems: the number of parameters to be learned increases with the size of the dataset. We claim that the potential gained accuracy is not enough to justify the computational cost and, in any case, it entails some approximations when testing on unseen data (they both assign the weight vector of the closest training instance in term of Euclidean distance).

As we will see, our approach (C2LM) is simple, theoretically founded, and accurate: it makes use of the geometric characteristics of the input space and weight vectors are learned on each pair of regions instead of each input instance. Moreover, it can be applied for modeling both distances and similarities; it is theoretically robust and has good performances in practice.

3. Learning Convex Combinations of Local Models

In this section, we present our optimization problem for learning convex combinations of local models which takes the form of a least absolute errors regression problem. For the sake of clarity, we first give the few notations we will employ in the rest of this paper.

3.1. Notations

Let X be the instance-pair space, i.e. the set of pairs $(x_1, x_2) \in U^2$, and $y : X \to Y \subset \mathbb{R}$ a metric function (the ground truth metric that can be a distance or a similarity function). We assume that U is a compact [6] convex metric space w.r.t. a norm $\|.\|$ so that $U \subset \mathbb{R}^d$. Thus, there exists a constant R such that $\forall x \in U,\ \|x\| \leq R$. We will refer to $\mathcal{Z} = X \times Y$ as the set of all possible valued pairs $p = (x_1, x_2, y(x_1, x_2))$, where $(x_1, x_2) \in X$ is a pair of instances and $y(x_1, x_2)$ is the associated target value. We also denote $P = \{p_i\}_{i=1}^n \subset \mathcal{Z}$ the set of n training pairs.

3.2. Optimization Problem

Let us suppose that the instance space U has been decomposed in K clusters or regions (one could perform a Kmeans according to the Euclidean distance), denoted $\{R_z\}_{z=1}^K$ and, on each cluster, a local model $s_z : X \to \mathbb{R}$ has been defined in order to compare instances belonging to that specific cluster. Let $S = \{s_z(.)\}_{z=1}^K$ be the set of metric functions related to the local models (which can be distance functions, $s_z : U^2 \to \mathbb{R}^+$, or similarity functions, $s_z : U^2 \to \mathbb{R}$). Our aim is to define on each pair of regions $(R_i, R_j) = R_{ij}$ a metric function $t_{ij} : X \to \mathbb{R}$ as a convex combination of S and that is symmetric. The problem we are trying to solve is how to compare instances potentially belonging to different clusters. For each pair of regions R_{ij} we will learn a vector W_{ij} of positive weights representing the contribution of each local model while estimating the similarity between an instance $x_1 \in R_i$ and an instance $x_2 \in R_j$. Therefore, the new metric function $t_{ij}(x_1, x_2)$ related to that pair of regions can be expressed as follows:

$$t_{ij}(x_1, x_2) = \sum_{z=1}^{K} W_{ijz} s_z(x_1, x_2). \quad (1)$$

Notice that, as we want the new function to be a metric, $\forall i, j = 1, ..., K\ \ t_{ij}(x_1, x_2) = t_{ji}(x_2, x_1)$: the $K \times K$ matrix of vectors $W = [W_{11} W_{12} ... W_{KK}]$ is symmetric, thus $\forall i, j = 1...K,\ W_{ij} = W_{ji}$.
We define a loss function $l : \mathcal{Z} \to \mathbb{R}$ over the training set P, corresponding to the gap between t_{ij} and the ground truth metric valued on each pair $p = (x_1 \in R_i, x_2 \in R_j, y(x_1, x_2))$:

$$l(W, p) = l(W_{ij}, (x_1 \in R_i, x_2 \in R_j, y(x_1, x_2)))$$

$$= |t_{ij}(x_1, x_2) - y(x_1, x_2)| \,. \quad (2)$$

Among all the possible norms, we choose to define our loss function as a L1-norm, i.e. the least absolute deviations, because of its robustness to outliers. This loss is assumed to be uniformly upper-bounded by a constant B, i.e. for any pair $p \in \mathcal{Z}$ the deviation of the predicted value from the expected one is finite.

We define our optimization problem, called C2LM, as follows:

$$\arg\min_{W} F_P(W) = \hat{R}^l + \lambda_1 D(W) + \lambda_2 S(W)$$
$$s.t.\ \forall i, j = 1, ..., K : \sum_{z=1}^{K} W_{ijz} = 1 \ and \ W_{ij} \geq 0 \quad (3)$$

where

$$\hat{R}^l = \frac{1}{n} \sum_{i,j,p \in R_{ij}} l(W, p) =$$
$$= \frac{1}{n} \sum_{i=1}^{K} \sum_{j=1}^{i} \sum_{p \in R_{ij}} \left| \sum_{z=1}^{K} W_{ijz} s_z(x_1, x_2) - y(x_1, x_2) \right| \quad (4)$$

is the mean loss over all training pairs, and

$$D(W) = \sum_{i=1}^{K} \sum_{j=1}^{i} \left\| E_{ij}^T W_{ij} \right\|_F^2 \quad (5)$$

$$S(W) = \sum_{i=1}^{K} \sum_{j=1}^{i} \sum_{i'=1}^{K} \sum_{j'=1}^{i'} K_{iji'j'} \left\| W_{ij} - W_{i'j'} \right\|_2^2 \quad (6)$$

are two regularizers used to avoid overfitting and λ_1 and λ_2 are the corresponding regularization parameters that have to be tuned by cross-validation.

The first term, $D(W)$ takes into account the prior influence of each local model in the computing of a weight vector. For instance, for a vector W_{ij} related to the pair of regions (R_i, R_j), we penalize a solution that has big weights associated to the local models that should not be influent in the computing of the associated metric. As a matter of fact, E_{ij} is a $1 \times K$ vector whose component E_{ijz} represents the prior influence of the metric s_z. E_{ijz} can be estimated in different ways. In our work, we base this estimation on the topological characteristics of the decomposition of the space U. As we can see in Figure 2, a local model defined on a region close to the pair of regions is more influent than one far from it.

The second term, $S(W)$, expresses the correlations between different weights' vectors. Through it, we force the space of weights' vectors to be smooth. In other words, we constrain the vectors defined on close pairs of regions to be similar. As for the prior influence, we base the estimation of the similarity between two vectors W_{ij} and $W_{i',j'}$, expressed by the parameter $K_{iji'j'}$, on the geometric characteristics of the instance space U (see Figure 3).

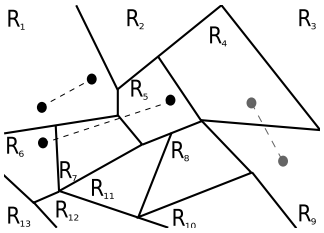


Figure 3: Similarity of a pair of regions: based on proximity, the vector W_{56} should be more similar to the vector W_{11} than to the vector W_{49}.

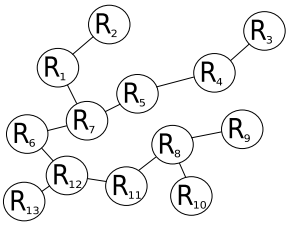


Figure 4: Minimum Spanning Tree: the distance between two regions corresponds to the number of edges of the shortest path connecting them. E.g., $dist(R_5, R_7) = 1$, $dist(R_{56}, R_4) = dist(R_5, R_4) + dist(R_6, R_4) = 4$ and $dist(R_{56}, R_{49}) = 5$.

In order to evaluate the prior influence of local models and the similarity between vectors of weights, we need to define a distance function between regions. We chose to build the Minimum Spanning Tree of the complete graph of region centroids (computed using the Euclidean distance), then to express the distance between two regions as the number of edges of the shortest path connecting them (see Figure 4). Therefore, for our experiments, we will consider E_{ijz} directly proportional to $dist(R_{ij}, R_z) = dist(R_i, R_z) + dist(R_j, R_z)$ and the similarity $K_{iji'j'} = \exp(-dist(R_{ij}, R'_{i'j}))$ exponentially decreasing with $dist(R_{ij}, R_{i'j'}) = min(dist(R_i, R_{i'}) + dist(R_j, R_{j'}), dist(R_i, R_{j'}) + dist(R_j, R_{i'}))$.

The learned combinations of local models are convex, as we fix their weights to be non-negative and to sum up to 1, and the resulting optimization problem is convex. Note that the number of parameters to learn depends on the number of regions K defined on the input space and is directly proportional to K^3, then the number of constraints is also directly proportional to K^3. This is a main advantage of applying C2LM to problems providing pairs of instances and their target score, if we consider the fact that $K \ll n$: in order to adapt the state of the art approaches (meant for classification tasks) to this kind of problems, a number of constraints directly proportional to the number of instances

of the dataset has to be added.

4. Robustness and Generalization Bound

In this section, we study the generalization ability of our algorithm according to the notion of algorithmic robustness introduced in [25]. This framework allows us to derive generalization bounds when the variation in the loss associated with two nearby training and testing examples is bounded. The closeness of two examples is based on the notion of covering number. By making use of the Bretagnolle-Huber-Carol inequality and proving that the metric functions $s_z(.)$ are lipschitz continuous, we can derive a PAC generalization bound for C2LM.

4.1. Theoretical guarantees

Let us define a partition of the space $\mathcal{Z} = X \times Y$ of all possible valued pairs $p = (x, x', y(x, x'))$ in order to establish if two pairs of instances are close. The partition is based on the notion of covering number.

Definition 1 (Covering Number [20]) For a metric space (S, ρ), and $T \subset S$, we say that $\hat{T} \subset T$ is a γ-cover of T if $\forall t \in T$, $\exists \hat{t} \subset \hat{T}$ such that $\rho(t, t') \leq \gamma$. The γ-covering number of T is

$$\mathcal{N}(\gamma, T, \rho) = \min\{|\hat{T}| : \hat{T} \text{ is a } \gamma\text{-covering of } T\}. \quad (7)$$

In other words, the γ-covering number of a metric space corresponds to the minimum number of regions of radius at most $\gamma > 0$ needed to cover it.
In order to define the closeness between instances of a metric space $\mathcal{Z} = X \times Y$, both the input X and the target Y spaces have to be partitioned. In most works [2, 13, 12, 14], Y is the finite set of labels, so its covering number is exactly equal to $|Y|$ and two instances are considered close if they have the same label. In our setting, we partition the space X into $\mathcal{N}(\gamma_1/2, X, \|.\|_2)$ subsets and the space Y into $\mathcal{N}(\gamma_2/2, Y, |.|)$, so that any region of X (resp. Y) has a diameter smaller than γ_1 (resp. γ_2). In this way, if $p = (x_1, x_2, y(x_1, x_2))$ and $p' = (x'_1, x'_2, y(x'_1, x'_2))$ belong to the same subset of $\mathcal{Z}$, then $\|x_1 - x'_1\|_2 \leq \gamma_1$, $\|x_2 - x'_2\|_2 \leq \gamma_1$ and $|y(x_1, x_2) - y(x'_1, x'_2)| \leq \gamma_2$. In the rest of this paper, we will refer to $H = \mathcal{N}(\gamma_1/2, X, \|.\|_2)\mathcal{N}(\gamma_2/2, Y, |.|)$ as the covering number of $\mathcal{Z}$.

Definition 2 (Algorithmic Robustness [25]). An algorithm A is said $(H, \epsilon(.))$-robust, for $H \in \mathbb{N}$ and $\epsilon : \mathcal{Z} \to \mathbb{R}$ if $\mathcal{Z}$ can be partitioned into H disjoint subsets, denoted by $\{C_i\}_{i=1}^H$, such that the following holds for all samples $P \in \mathcal{Z}$:

$$\begin{aligned} &\forall p \in P, \forall p' \in \mathcal{Z}, \forall i = 1, ..., H \\ &if\ p, p' \in C_i\ then\ |l(A, p) - l(A, p')| \leq \epsilon(P). \end{aligned} \quad (8)$$

The following concentration inequality provides a probability bound on the deviation of a multinomial random variable from its expected value. We will use it for obtaining information about the theoretical distribution of the valued pairs $p \in \mathcal{Z}$ over the regions of the partition.

Proposition 1([20]) Let $(|N_1|), ..., |N_H|)$ an IID (Independent and Identically Distributed) multinomial random variable with parameters n and $(p(C_1), ..., p(C_H))$. By the Bretagnolle-Huber-Carol inequality we have: $\mathbb{P}(\sum_{i=1}^H \left|\frac{|N_i|}{n} - p(C_i)\right| \geq \lambda) \leq 2^H \exp \frac{-n\lambda^2}{2}$, hence with probability at least $1 - \delta$,

$$\sum_{i=1}^{H} \left| p(C_i) - \frac{|N_i|}{n} \right| \leq \sqrt{\frac{2H \ln 2 + 2\ln(1/\delta)}{n}}. \quad (9)$$

We denote R^l the true loss $R^l = \mathbb{E}_{p\sim\mathcal{Z}} l(W, p)$ and $\hat{R}^l$ the empirical loss $\hat{R}^l = \mathbb{E}_{p\sim P} l(W, p)$.

We can now derive a PAC generalization bound for C2LM. We first prove that our algorithm is robust that requires to prove that $\forall z = 1, ..., K : s_z(.)$ is θ_z-lipschitz. According to the nature of the local metric functions $s_z(.)$, the proof of θ_z-lipschitzness varies. In Sections 4.2 and 4.3, we will instantiate $s_z(.)$ with Mahalanobis-like distances and bilinear similarities.

Lemma 1 If $\forall z = 1, ..., K$, $s_z(.)$ is θ_z-lipschitz w.r.t. the norm $\|.\|_2$, the optimization problem (3) is $(H, \theta\sqrt{2}\gamma_1 + \gamma_2)$-robust, with $\theta = \max_{z=1..K} \theta_z$.

Proof. We can partition $\mathcal{Z}$ into $H = \mathcal{N}(\gamma_1/2, X, \|.\|_2)\mathcal{N}(\gamma_2/2, Y, |.|)$ disjoint subsets, such that if $p = (x_1, x_2, y(x_1, x_2))$ and $p = (x'_1, x'_2, y(x'_1, x'_2))$ belong to the same subset C_h, then $x_1, x'_1 \in R_i$ so $\|x_1 - x'_1\|_2 \leq \gamma_1$, also $x_2, x'_2 \in R_j$ so $\|x_2 - x'_2\|_2 \leq \gamma_1$ and $|y(x_1, x_2) - y(x'_1, x'_2)| \leq \gamma_2$. We have, then:

$$|l(W_{ij}, p) - l(W_{ij}, p')| = \quad (10)$$

$$\left| \left| \sum_{z=1}^{K} W_{ijz} s_z(x_1, x_2) - y(x_1, x_2) \right| - \left| \sum_{z=1}^{K} W_{ijz} s_z(x'_1, x'_2) - y(x'_1, x'_2) \right| \right|$$

$$\leq \left| \sum_{z=1}^{K} W_{ijz} s_z(x_1, x_2) - \sum_{z=1}^{K} W_{ijz} s_z(x'_1, x'_2) - y(x_1, x_2) + y(x'_1, x'_2) \right| \quad (11)$$

$$\leq \left| \sum_{z=1}^{K} W_{ijz} (s_z(x_1, x_2) - s_z(x'_1, x'_2)) \right| + \left| y(x_1, x_2) - y(x'_1, x'_2) \right|$$

$$\leq \sum_{z=1}^{K} |W_{ijz}| \left| s_z(x_1, x_2) - s_z(x'_1, x'_2) \right| + \gamma_2 \quad (12)$$

$$\leq \sum_{z=1}^{K} |W_{ijz}| \, \theta_z \left\| \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} x'_1 \\ x'_2 \end{pmatrix} \right\|_2 + \gamma_2 \quad (13)$$

$$\leq \theta \left\| \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} x'_1 \\ x'_2 \end{pmatrix} \right\|_2 \sum_{z=1}^{K} W_{ijz} + \gamma_2 \quad (14)$$

$$\leq \theta\sqrt{2}\gamma_1 + \gamma_2 \,. \tag{15}$$

Eq. 11 is due to the reverse triangle inequality. Inequality 13 is valid because s_z is multi-variate θ_z-lipschitz continuous w.r.t. the norm $\|.\|_2$ (see below). In Eq. 14, we define $\theta = \max_{\forall z=1..K} \theta_z$ and recall that $\forall i, j = 1, ..., K : W_{ij} \geq 0$. Eq. 15 is due to $\sum_{z=1}^{K} W_{ij} = 1$. $\sqrt{2}\gamma_1$ is the maximum $\|.\|_2$ distance between the two vectors. □

In the previous proof, we made use of the notion of Multi-variate Lipschitz continuity.

Definition 3 (Multi-variate Lipschitz continuity). A function $f : U^2 \subset \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, with U a convex space, is said θ-lipschitz w.r.t. the norm $\|.\|_2$ if $\exists\, \theta \in \mathbb{R}, \theta > 0$ that $\forall x_1, x_2, x_1', x_2' \in U$:

$$\|f(x_1, x_2) - f(x_1', x_2')\|_2 \leq \theta \left\| \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \begin{pmatrix} x_1' \\ x_2' \end{pmatrix} \right\|_2 . \tag{16}$$

Roughly speaking, a function that is lipschitz continuous varies slightly within a certain interval. This property is fundamental for the robustness of our algorithm: the fact that the functions $S = \{s_z(.)\}_{z=1}^{K}$ are θ_z-lipschitz continuous implies that any linear combination of them returns similar values when evaluated on instances belonging to the same region of the partition. According to [26], the constant θ can be estimated considering the fact that

$$\theta = \max_{\forall x_1, x_2, x_1', x_2' \in U} \left(\frac{\|f(x_1, x_2) - f(x_1', x_2')\|_2}{\left\| \binom{x_1}{x_2} - \binom{x_1'}{x_2'} \right\|_2} \right) = \\ = \max_{\forall x_1, x_2 \in U} \|\nabla f(x_1, x_2)\|_2 \,. \tag{17}$$

We can now derive the generalization bound of C2LM.

Lemma 2 As $F_P(W)$ is $(H, \theta\sqrt{2}\gamma_1 + \gamma_2)$-robust and the training set P is obtained from n IID draws according to a multinomial random variable, for any $\delta > 0$ with probability at least $1 - \delta$, we have:

$$|R^l - \hat{R}^l| \leq \theta\sqrt{2}\gamma_1 + \gamma_2 + B\sqrt{\frac{2H \ln 2 + 2 \ln 1/\delta}{n}} \,. \tag{18}$$

Proof: See Supplementary Material.

It is worth noting that this bound tends to zero as the covering number H increases ($\gamma_1 \to 0$ and $\gamma_2 \to 0$) and the number of samples $n \to \infty$. In the following subsections, we will instantiate $s_z(.)$ with two different metric functions: first as a Mahalanobis-like distance and then as a bilinear similarity. For both of them, we will need to prove their θ_z-lipschitz continuity and estimate their constant θ_z as defined in Def. 3.

4.2. Derivation for Mahalanobis-like Local Models

The Mahalanobis distance of a pair (x_1, x_2) valued for a local model z can be written as $s_z(x_1, x_2) = d_{M_z}(x_1, x_2) = \sqrt{(x_1 - x_2)^T M_z (x_1 - x_2)}$ with M_z the corresponding (learned) PSD matrix. Thus, our objective function takes the following form:

$$F_P(W) = \frac{1}{n} \sum_{i=1}^{K} \sum_{j=1}^{i} \sum_{p \in R_{ij}} \left| \sum_{z=1}^{K} W_{ijz} d_{M_z}(x_1, x_2) - y(x_1, x_2) \right| \\ + \lambda_1 D(W) + \lambda_2 S(W) \tag{19}$$

where $M = \{M_1, .., M_K\}$ is a set of Mahalanobis metrics.

Lemma 3 $\forall z = 1, ..., K$ the Mahalanobis distance $d_{M_z}(x_1, x_2)$ is θ_z-lipschitz w.r.t. the norm $\|.\|_2$, with $\theta_z = \sqrt{2}\,\|L_z\|_2$.

Proof: See [26].

Lemma 4 $F_P(W)$ is $(H, 2\gamma_1 \|L\|_2 + \gamma_2)$-robust and for any $\delta > 0$ with probability at least $1 - \delta$, we have:

$$|R^l - \hat{R}^l| \leq 2\gamma_1 \|L\|_2 + \gamma_2 + B\sqrt{\frac{2H \ln 2 + 2 \ln 1/\delta}{n}}. \tag{20}$$

The constant $\|L\|_2$ corresponds to $\max_{\forall z=1..K} \|L_z\|_2$ so that $\theta = \sqrt{2}\,\|L\|_2$, because $\theta_z = \sqrt{2}\,\|L_z\|_2$.

4.3. Derivation for Bilinear Similarity Local Models

The bilinear similarity of a pair (x_1, x_2) can be written as $s_z(x_1, x_2) = x_1^T M_z x_2$. Thus, our problem becomes:

$$F_P(W) = \frac{1}{n} \sum_{i=1}^{K} \sum_{j=1}^{i} \sum_{p \in R_{ij}} \left| \sum_{z=1}^{K} W_{ijz} x_1^T M_z x_2 - y(x_1, x_2) \right| \\ + \lambda_1 D(W) + \lambda_2 S(W) \tag{21}$$

where $M = \{M_1, .., M_K\}$ is a set of bilinear similarities.

Lemma 5 $\forall z = 1, ..., K$ the bilinear similarity $s_z(x_1, x_2) = x_1^T M_z x_2$ is θ_z-lipschitz w.r.t. the norm $\|.\|_2$, with $\theta_z = \sqrt{2}\,\|M_z\|_2\, R$.

Proof: See [26].

Lemma 6 $F_P(W)$ is $(H, 2\gamma_1 \|M\|_2 R)$-robust and for any $\delta > 0$ with probability at least $1 - \delta$, we have:

$$|R^l - \hat{R}^l| \leq 2\gamma_1 \|M\|_2 R + \gamma_2 + B\sqrt{\frac{2H \ln 2 + 2 \ln 1/\delta}{n}} \tag{22}$$

$\|M\|_2 = \max_{\forall z=1..K} \|M_z\|_2$ so that $\theta = \sqrt{2}\,\|M\|_2\, R$, because $\theta_z = \sqrt{2}\,\|M_z\|_2\, R$.

5. Experiments

In this section, we aim at showing that C2LM is well suited to deal with both distance and similarity functions. Therefore, we empirically evaluate our method on two applications: first on the estimation of perceptual color distances and then on the estimation of semantic similarities between words.

5.1. Applications and Datasets

Modeling perceptual color distances It is known that a human observer cannot distinguish all the shades corresponding to the different mixtures of light wavelengths. He is more sensitive to medium wavelengths (to green/yellow colors) than to short and large wavelengths of the visible spectrum. Moreover, human perception strongly depends on variations of visual conditions, such as brightness, luminance, background changes, and so on. The perceived difference between colors cannot be modeled using an additive color space as the RGB space, because the corresponding distance is not proportional to the Euclidean distance on that space.

In the past, several perceptual color spaces have been proposed to better model the human color perception : CIELuv and CIELab (see [19]) are two examples of such efforts to model uniform perceptual spaces. However, these spaces are still sensitive to some visual variations and can be used only under standard image acquisition conditions. This is because the camera configuration, such as white balance, demosaicing and gamma correction, have a huge impact on the final perception of the color distances.

We claim that, by means of C2LM, we can model a perceptual color distance that is invariant to acquisition conditions. For our experiments, we use the dataset built by Perrot et al. [17]. We have at our disposal 29580 color patches, expressed in their RGB coordinates and uniformly distributed in the RGB cube, and 41800 pairs of color patches, taken under several viewing conditions and with 4 different cameras, with their reference perceptual distance ΔE_{00}. Such a target distance corresponds to the perceptual color distance and has been computed using the CIEDE2000 color-difference formula [18] based on CIELab space. However, it is reliable only under standard viewing conditions (illuminant D65, illuminance of 1000 lx, etc. defined by the International Commission on Illumination CIE) so it cannot be used in all circumstances. Our proposal is to approximate the true perceptual distance between two colors no matter the viewing conditions. For this aim, the color patches are clustered using k-means (using the Euclidean distance on the RGB space) and on each so-found region a local model is learned as a Mahalanobis-like distance (using the color pairs whose patches both belong to that region). We then apply our method for learning linear combinations of those distance functions with manifold regularization, as detailed in section 3. We compare our method to [17], where the authors learn a set of Mahalanobis-like metrics independently from each other: they cluster the color patches using k-means and learn a local metric on each cluster and a global one with the color pairs whose patches belong to different clusters; they compute the distance between two colors using the local distance if they belong to the same cluster or the global distance if they do not. As [17], we evaluate our method on two different tasks (testing on unseen colors and on color pairs from unseen cameras).

Modeling semantic similarities The semantic similarity between words is defined as the measure of closeness in meaning between two terms. It is a measure defined by human perception and it cannot be expressed by exact rules. Nevertheless, it can be estimated by representing the words as vectors of a continuous space (word embedding) and computing their distance or similarity, for instance the Euclidean distance or the cosine similarity. We show how a word embedding can be enhanced using our method. As in the previous application, we learn a local model on each cluster of words (the clustering procedure accomplished using k-means with the Euclidean distance on the word embedding) and then we apply C2LM on the learned local models, which, in this case, are bilinear forms (see 4.3) computed independently using the following optimization problem:

$$\arg\min_{B_z} \frac{1}{n} \sum_{p \in R_{zz}} \left| x_1^T B_z x_2 - y(x_1, x_2) \right| + \|B_z\|_{\mathcal{F}}. \qquad (23)$$

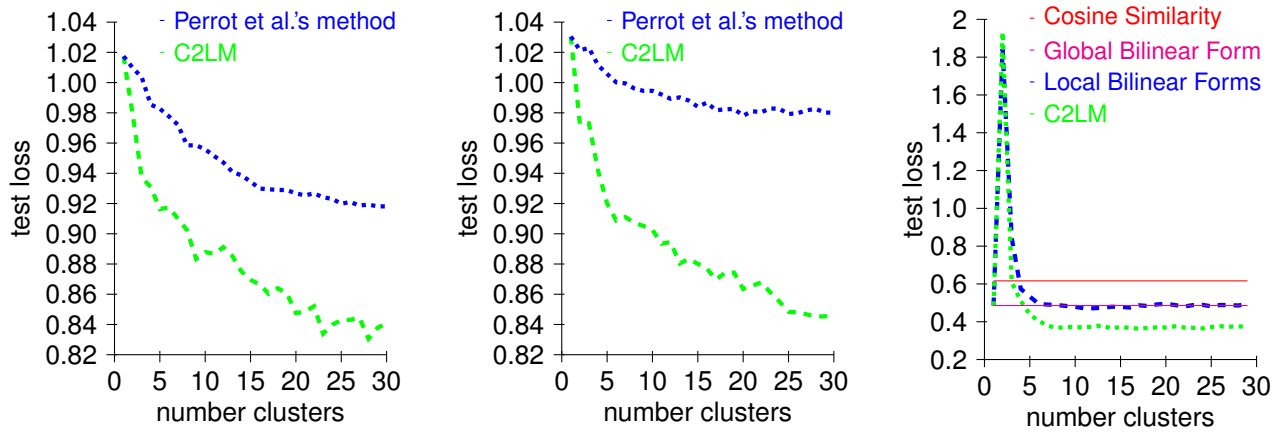
For our experiments, we extracted the word embedding from the Reuters News stories[2] text corpus using the Hellinger PCA as presented in [11]. We then evaluate different methods on the WordSim353-similarity dataset: it is composed of 353 pairs of english words and for each pair we have at our disposal its semantic similarity as estimated by a human expert. We will compare our method with computing the cosine similarity directly on the embedding and with learning a set of local bilinear similarities and a global one. Because the cosine similarity is capable of predicting scores only in the interval $[-1, 1]$ and the similarity scores of the dataset are between 0 and 10, we first normalized the target scores into the interval $[-1, 1]$.

5.2. Implementation and results

We implemented our algorithm using the Cvxpy library[3] and its SCS solver (see [16]). For our experiments, we computed the best values for parameters λ_1 and λ_2 executing a grid search hyperparameter optimization by cross-validation: we fixed them to $\lambda_1 = 0.01$ and

[2] http://about.reuters.com/researchandstandards/corpus/

[3] cvxpy.readthedocs.org/en/latest/



(a) Results on unseen colors

(b) Results on unseen cameras

(c) Results on wordsim353 dataset

Figure 5: Comparison of our method and local metric learning approaches, such as Perrot et al.'s method, for the application on perceptual color distances (5a and 5b) and for the application on word semantic similarities (5c). The used criterion is the loss over the test instances.

$\lambda_2 = 10000$ for the first application and to $\lambda_1 = 0.0001$ and $\lambda_2 = 100$ for the second one.

For the application on unseen colors, we show the mean results of a 6-fold cross validation of the color patches set, iterated five times. In Figure 5a, we represent the variation of the test loss over the number of clusters. We notice that as the number of clusters increases the empirical test loss decreases: a set of local metrics captures much better the underlying geometry of the color space than a unique global metric ($K = 1$). Moreover, with a small number of clusters, the learned linear combinations are more expressive than the local metrics: thanks to the prior influence and similarity regularizations, we successfully prevent the model from overfitting the training instances. This trend is more and more important as the number of clusters grows. For the application on unseen cameras, Figure 5b shows the mean results of a 4-fold cross validation (leave one camera out) of the color pairs set, iterated 3 times. Once again, our method outperforms the state of the art. For both tasks, we can note that with a very limited number of clusters, that is only 5, our test loss is always smaller than every test loss the approach of [17] could attain, even with 30 clusters. In addition, we use the learned color metrics to perform image segmentation and provide illustrations in the supplementary material.

Concerning the application on semantic similarity, Figure 5c presents the mean results of a 6-fold cross validation, iterated five times. We can note that learning metrics on the word embedding gives better results than applying directly the cosine similarity, but also that the local metrics fail to improve the test error with respect to a global bilinear form. On the contrary, C2LM converges with a limited number of clusters to an enhanced test error. We also notice that, against the trend, the test error increases when passing from one to two clusters. This can be explained by the fact that the quality of the local models is so poor that the learned convex combinations of them cannot be good.

6. Conclusion

In this paper, we proposed a new method for learning convex combinations of local models given a prior knowledge on their correlations. We proved that our learning algorithm is theoretically founded w.r.t. the algorithmic robustness framework. Empirically, our approach has better results than the state of the art to estimate perceptual color distances and semantic word similarities.
So far, we assumed that the local models were provided. A possible perspective of this work is to jointly learn the local metrics and their linear combinations. The optimization problem would take the form of a double regression, one over the points belonging to the same region and one for all the others. In this way, we could guarantee that the local models perform well both locally and globally speaking by means of regularization.

7. Acknowledgments

We wish to thank the reviewers for their valuable comments and suggestions contribute to improve the final version of the paper. We also thank the ANR projects SOLSTICE (ANR-13-BS02-01) and LIVES (ANR-15-CE23-0026-03).

References

[1] F. R. Bach, G. R. Lanckriet, and M. I. Jordan. Multiple kernel learning, conic duality, and the smo algorithm. In *Proceedings of the twenty-first international conference on Machine learning*, page 6. ACM, 2004.

[2] A. Bellet and A. Habrard. Robustness and Generalization for Metric Learning. *Neurocomputing*, 151(1):259–267, 2015.

[3] A. Bellet, A. Habrard, and M. Sebban. A survey on metric learning for feature vectors and structured data. *arXiv preprint arXiv:1306.6709*, 2013.

[4] A. Bellet, A. Habrard, and M. Sebban. *Metric Learning*. Morgan & Claypool, 2015.

[5] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning. In *Proceedings of the 24th international conference on Machine learning*, pages 209–216. ACM, 2007.

[6] M. Fréchet. Sur les fonctionnelles continues. In *Annales Scientifiques de L'Ecole Normale Superieure*, volume 27, pages 193–216, 1910.

[7] S. Hauberg, O. Freifeld, and M. J. Black. A geometric take on metric learning. In *Advances in Neural Information Processing Systems*, pages 2024–2032, 2012.

[8] Y. Huang, C. Li, M. Georgiopoulos, and G. C. Anagnostopoulos. Reduced-rank local distance metric learning. In *Machine Learning and Knowledge Discovery in Databases*, pages 224–239. Springer, 2013.

[9] B. Kulis. Metric learning: A survey. *Foundations and Trends in Machine Learning*, 5(4):287–364, 2012.

[10] R. Lajugie, D. Garreau, F. R. Bach, and S. Arlot. Metric learning for temporal sequence alignment. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 1817–1825, 2014.

[11] R. Lebret and R. Collobert. Word emdeddings through hellinger pca. *arXiv preprint arXiv:1312.5542*, 2013.

[12] Y. Mansour and M. Schain. Robust domain adaptation. *Annals of Mathematics and Artificial Intelligence*, 71(4):365–380, 2014.

[13] E. Morvant, A. Habrard, and S. Ayache. Parsimonious unsupervised and semi-supervised domain adaptation with good similarity functions. *Knowledge and Information Systems*, 33(2):309–349, 2012.

[14] M.-I. Nicolae, M. Sebban, A. Habrard, É. Gaussier, and M.-R. Amini. Algorithmic robustness for semi-supervised (ϵ, γ, τ)-good metric learning. *arXiv preprint arXiv:1412.6452*, 2014.

[15] Y.-K. Noh, B.-T. Zhang, and D. D. Lee. Generative local metric learning for nearest neighbor classification. In *Advances in Neural Information Processing Systems*, pages 1822–1830, 2010.

[16] B. ODonoghue, E. Chu, N. Parikh, and S. Boyd. Conic optimization via operator splitting and homogeneous self-dual embedding. 2015.

[17] M. Perrot, A. Habrard, D. Muselet, and M. Sebban. Modeling perceptual color differences by local metric learning. In D. J. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, editors, *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, volume 8693, pages 96–111. Springer, 2014.

[18] G. Sharma, W. Wu, and E. N. Dalal. The ciede2000 color-difference formula: Implementation notes, supplementary test data, and mathematical observations. *Color research and application*, 30(1):21–30, 2005.

[19] M. Tkalcic, J. F. Tasic, et al. Colour spaces: perceptual, historical and applicational background. In *Eurocon*, pages 304–308, 2003.

[20] A. W. Van Der Vaart and J. A. Wellner. Weak convergence. In *Weak Convergence and Empirical Processes*, pages 16–28. Springer New York, 1996.

[21] J. Wang, A. Kalousis, and A. Woznica. Parametric local metric learning for nearest neighbor classification. In *Advances in Neural Information Processing Systems*, pages 1601–1609, 2012.

[22] K. Q. Weinberger, J. Blitzer, and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. In *Advances in neural information processing systems*, pages 1473–1480, 2005.

[23] K. Q. Weinberger and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. *The Journal of Machine Learning Research*, 10:207–244, 2009.

[24] E. P. Xing, M. I. Jordan, S. Russell, and A. Y. Ng. Distance metric learning with application to clustering with side-information. In *Advances in neural information processing systems*, pages 505–512, 2002.

[25] H. Xu and S. Mannor. Robustness and generalization. *Machine learning*, 86(3):391–423, 2012.

[26] V. Zantedeschi, R. Emonet, and M. Sebban. Lipschitz continuity of mahalanobis distances and bilinear forms. *arXiv preprint arXiv:1604.01376*, 2016.