Pattern Recognition 000 (2017) 1–12

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Learning local metrics from pairwise similarity data

Julien Bohné[a,b,*], Yiming Ying[c], Stéphane Gentric[a], Massimiliano Pontil[b,d]

[a] Safran Identity & Security, 11 boulevard Galliéni, 92130 Issy-les-Moulineaux, France
[b] Department of Computer Science, University College London, London WC1E 6BT, UK
[c] SUNY Albany, Department of Mathematics and Statistics, 1400 Washington Avenue, Albany, NY, 12222, USA
[d] Istituto Italiano di Tecnologia, Via Morego 30, 16163 Genoa, Italy

ARTICLE INFO

Article history:
Received 22 July 2016
Revised 13 March 2017
Accepted 4 April 2017
Available online xxx

Keywords:
Similarity function learning
Local metric learning
Nearest neighbors classification
Face verification

ABSTRACT

We study the problem of learning a similarity function from a set of binary labeled data pairs. A common approach is to learn a similarly function which is a bilinear form associated to the pair of data points. We argue that this class may be too restrictive when handling heterogeneous datasets. To overcome this limitation local metric learning techniques have been advocated in the literature. However, they are subject to certain constraints preventing their usage in many applications. For example, they require knowledge of the class label of the training points. In this paper, we present a local metric learning method, which overcomes these limitations. The method first initializes a Gaussian mixture model on the training data. Then it estimates a set of local metrics and simultaneously refines the mixture's parameters. Finally, a similarity function is obtained by aggregating the local metrics. We also introduce a novel regularization term, which works well in a transfer learning setting. Our experiments show that the proposed method achieves state-of-the-art results on several real datasets.

1. Introduction

Metric learning is at the core of many applications and hence is a widely studied problem in machine learning. A popular class of similarity functions are linear metrics of the form $(x_i - x_j)^\top M(x_i - x_j)$ where x_i and x_j are two data points which we wish to compare and M is a positive semidefinite matrix (PSD). This approach has been shown to be quite effective on various tasks but suffers a strong limitation: it makes use of a single linear metric to compare data over all the input space. This simple approach might lead to unsatisfactory results when used on heterogeneous data. In principle, one could use global metric learning with a nonlinear kernel to be able to address more complex data. However, finding the appropriate kernel may be challenging and this type of approach may be computationally expensive, especially at test time when speed matters the most.

This observation is the root of development of local metric learning methods which combine several linear metrics to adapt the similarity function to the local specificities of the data. For illustrative purpose let us consider two examples. Each local metric is associated with a PSD matrix M_i. If L_i is a square root of M_i, that is $M_i = L_i L_i^\top$, then the mapping $x \mapsto L_i^\top x$ is the feature vector associated with the ith local metric. These feature mappings may vary considerably across regions. For example, it is well known that in digit classification some digits are easily mistaken for another such as "1" and "7" or "3" and "8", it seems therefore reasonable to reduce the number of misclassification by focusing on different features to discriminate digits in the "1-7" region and in the "3-8" one. Yet one more example, is face verification: should we put the emphasis on the very same features to compare two pictures of Caucasian males and two pictures of Asian females? We claim the answer is "no" and our experiments show that local metric learning improves the performance for these two applications.

Similarity functions can be employed in nearest neighbor classifiers or to take decisions based on the thresholding of the similarity. Those two situations are different as the former depends only on the ranking of the nearest neighbors whereas the latter is concerned with the absolute meaning of the similarity. In this work we propose a flexible local metric learning method called Large Margin Local Metric Learning (LMLML) which can be employed in both settings and can handle an arbitrarily large number of classes. Its training procedure does not need one to know the class labels but only a set of pairs labeled "similar" (both points belong to the same class) or "dissimilar" (the two points belong to different classes). Our method computes a set of local metrics which are combined into an adaptive similarity function with the help of a soft partitioning of the input space based on Gaussian

* Corresponding author.
E-mail addresses: julien.bohne@safrangroup.com, jbohne78-ucl@yahoo.fr (J. Bohné), yying@albany.edu (Y. Ying), stephane.gentric@safrangroup.com (S. Gentric), massimiliano.pontil@iit.it (M. Pontil).

http://dx.doi.org/10.1016/j.patcog.2017.04.002
0031-3203/

Mixture Models (GMM). The optimization of the local metrics is formulated as a convex problem which favors a large margin solution. The problem also involves a novel regularization term encouraging matrices which are close to a simple baseline solution. Our experiments show that LMLML outperforms or matches state-of-the-art results on various datasets.

The paper is organized in the following manner. In the next section we review related work. In Section 3, we present our large margin local metric learning (LMLML) approach and in Section 4 we extend this method with a new regularizer. Section 5 presents implementation details contributing to LMLML's good performance which are demonstrated by experiments on both synthetic and real datasets in Section 6. Finally, in Section 7 we summarize our findings.

We note that this article is a longer version of the conference paper by the authors [1] which includes new methodological and experimental results. Specifically, (i) we provide a full optimization scheme which alternates gradient steps with respect to the local metrics and the GMM partitioning the feature space; (ii) we give a new formulation of the regularizer with no auxiliary variables; (iii) finally we present extended experiments on three additional real datasets and illustrate the benefit of the new regularizer for out of sample / transfer learning scenarios.

2. Related work

Metric learning has been the subject of several papers. Most of them deal with global metric learning; important developments include ITML [2], LDML [3], LMNN [4] or DML_eig [5]. They all formulate metric learning as the optimization of an objective function which decreases the distances of similar pairs while increasing those of dissimilar pairs. Some of these methods also include a regularization term which aims at alleviating the risk of overfitting.

In practice, the discriminating property of the input features might vary between different neighborhoods, and thus a global metric cannot reflect the local specificities of the data. A more general approach is to learn a metric on each local neighborhood. This approach, which is often referred to as *local metric learning*, has been investigated from several angles. We briefly review some of the key developments below.

Local metric learning has sometimes been linked to semi-supervised clustering such as in [6] where labeled data are used to find local transformations of the data points in order to improve a clustering process. This kind of method cannot compute similarity measures between pairs of never seen points which is the goal of our work.

Metric learning is often used to improve nearest neighbors classification. Several local metric learning algorithms have been developed to improve nearest neighbors classifiers. Weinberger and Saul proposed an extension of LMNN to local metric (MM-LMNN [7]), in which a specific metric is associated to each class and all the metrics are jointly learned to optimize a classification criterion. LMNN has also been extended to the setting of multi-tasks learning [8], where multiple metrics are jointly learned [9]. Our work can be considered to be a generalization of these methods as it uses a weighted combination metrics instead of activating a single metric for each comparison (see Section 3).

KISSME [10] has also been extended to local metric in [11] where one KISSME metric is learned separately for each class. These class-specific metrics are averaged with a global one to alleviate the risk of overfitting due to the fact that each metric might be learned using only a limited number of training samples. The method of GLML [12] employs local metrics to reduce the performance bias due to finite sampling using the class conditional probability distribution.

Most previously presented methods suffer from the same drawback, namely they need enough training samples per class to estimate the metrics. Therefore, they cannot directly be employed for applications in which there are a large number of classes with only few training data points in some classes. PLML, a local metric learning method based on finite number of linear metrics, is introduced in [13] to overcome this problem. The number of metrics is different from the number of classes and hence the method can scale to a larger number of classes. However, this method is specifically designed for nearest neighbors classification as it can only compute the similarity of pairs for which at least one data point is in the training set. This strongly limits the practical range of tasks that PLML can deal with. In particular, it prevents the application of PLML to the problem of face verification.

All the methods mentioned above cannot deal with datasets having a large number of classes or are unable to compute a similarity function for pairs of two points which do not belong to the training dataset. Up to our knowledge LMLML [1] has been the first method to overcome these limitations. A further development is provided by CLML [14]; it jointly learns many locally linear projections such that any pair of projected points can be effectively compared using the Euclidean distance. Like in our work, the input space is soft-partitioned using a GMM but, as opposed to what we propose, the GMM parameters are learned during an initial step and are regarded as fixed during the projections optimization.

At last we note that local metric is one way to extend linear metric but other directions have been explored to combined multiple metrics. For example, [15] learns several metrics and selects the one giving the smallest distance for each comparison. They also show how to make video-to-video comparisons using additional latent variables to select which video frames to compare.

3. Large margin local metric learning

In this section we present our large margin local metric learning (LMLML) approach. Let $\mathcal{S}_+^n$ be the set of $n \times n$ PSD matrices. The usual squared Mahalanobis distance associated with a matrix $M \in \mathcal{S}_+^n$ and evaluated on a pair of data points $(x_i, x_j) \in \mathbb{R}^n \times \mathbb{R}^n$ is given by $(x_i - x_j)^\top M(x_i - x_j)$. In LMLML, the matrix M is replaced by a matrix-valued function $\mathcal{M}_\theta : \mathbb{R}^n \times \mathbb{R}^n \mapsto \mathcal{S}_+^n$ which is defined, for every $(x_i, x_j) \in \mathbb{R}^n \times \mathbb{R}^n$, as a convex combination of $K+1$ matrices

$$\mathcal{M}_\theta(x_i, x_j) = \sum_{k=0}^{K} w_\theta^k(x_i, x_j) M_k, \tag{1}$$

where $w_\theta^k(x_i, x_j)$ are nonnegative weights which will be defined below. The resulting similarity function is given, for every $(x_i, x_j) \in \mathbb{R}^n \times \mathbb{R}^n$, by the formula

$$d^2(x_i, x_j, \mathcal{M}_\theta) = (x_i - x_j)^\top \mathcal{M}_\theta(x_i, x_j)(x_i - x_j). \tag{2}$$

The smoothness of the matrix-valued function $\mathcal{M}$ is a desirable property because it guarantees that the similarity function is local and prevents abrupt changes which, as we observe in our experiments below, degrade performance. In order to ensure this property, we use weights w_θ^k which vary smoothly across the input space. As we want the similarity function to be local, it makes sense to use a soft partitioning of the input space to compute the weights $w_\theta^k(x_i, x_j)$. To this end, we employ a Gaussian Mixture Model (GMM) with K components of parameters $\theta = \{\alpha_k, \mu_k, S_k\}_{1 \le k \le K}$, where α_k are the prior of each Gaussian and μ_k and S_k the corresponding means and covariance matrices, respectively. The weights are defined by the formula

$$w_\theta^k(x_i, x_j) = \begin{cases} \beta & \text{if } k = 0 \\ P(k|x_i, \theta) + P(k|x_j, \theta) & \text{otherwise} \end{cases} \tag{3}$$

Please cite this article as: J. Bohné et al., Learning local metrics from pairwise similarity data, Pattern Recognition (2017), http://dx.doi.org/10.1016/j.patcog.2017.04.002