Matrix Variate Gaussian Mixture Distribution Steered Robust Metric Learning

Lei Luo, Heng Huang*
Electrical and Computer Engineering, University of Pittsburgh, USA
lel94@pitt.edu, heng.huang@pitt.edu

Abstract

Mahalanobis Metric Learning (MML) has been actively studied recently in machine learning community. Most of existing MML methods aim to learn a powerful Mahalanobis distance for computing similarity of two objects. More recently, multiple methods use matrix norm regularizers to constrain the learned distance matrix $\mathbf{M}$ to improve the performance. However, in real applications, the structure of the distance matrix $\mathbf{M}$ is complicated and cannot be characterized well by the simple matrix norm. In this paper, we propose a novel robust metric learning method with learning the structure of the distance matrix in a new and natural way. We partition $\mathbf{M}$ into blocks and consider each block as a random matrix variate, which is fitted by matrix variate Gaussian mixture distribution. Different from existing methods, our model has no any assumption on $\mathbf{M}$ and automatically learns the structure of $\mathbf{M}$ from the real data, where the distance matrix $\mathbf{M}$ often is neither sparse nor low-rank. We design an effective algorithm to optimize the proposed model and establish the corresponding theoretical guarantee. We conduct extensive evaluations on the real-world data. Experimental results show our method consistently outperforms the related state-of-the-art methods.

Introduction

Mahalanobis metric learning (MML) has been actively studied in machine learning community and successful applied to address numerous applications (Kuznetsova et al. 2016). MML methods target to learn a good Mahalanobis metric to effectively gauge the pairwise distance between data objects. Particularly, the distance matrix $\mathbf{M}$ plays a crucial role in MML. A well-learned $\mathbf{M}$ can precisely reflect domain-specific connections and relationships. Toward this end, many metric learning algorithms under various problem settings have been proposed, such as pairwise constrained component analysis (PCCA) (Mignon and Jurie 2012), neighborhood repulsed metric learning (NRML) (Lu et al. 2014), large margin nearest neighbor (LMNN) (Weinberger and Saul 2009), logistic discriminant metric learning (LDML) (Guillaumin, Verbeek, and Schmid 2009), and Hamming distance learning (Zheng, Tang, and Shao 2016; Zheng and Shao 2016). Although these supervised algorithms have shown great success in many applications, finding a robust distance metric from real-world data remains a big challenge.

To enhance the performance of metric learning models, some recent methods, represented by low-rank or sparse metric learning (Huo, Nie, and Huang 2016; Ying, Huang, and Campbell 2009), actively integrate different regularization terms associated with the matrix $\mathbf{M}$ in modeling. On one hand, using these regularization terms can effectively prevents the overfitting problem since it is equivalent to adding the prior knowledge into $\mathbf{M}$. On the other hand, these regularization terms can extract partial structure information of the matrix $\mathbf{M}$. The structure of a variable corresponds to its spatial distribution. However, for many practical applications, the spatial distribution of $\mathbf{M}$ is generally irregular and complicated. The simple matrix norm regularizers cannot characterize the structure information well. For instance, L_2 or L_1-norm regularization is based on the hypothesis that the elements of $\mathbf{M}$ are independently distributed with Gaussian distribution or Laplace distribution (Luo et al. 2016b). Obviously, they cannot capture the spatial structure of $\mathbf{M}$. The low-rank and Fantope regularizers (Law, Thome, and Cord 2014) overcome this limitation, but they lack the generalization because the distance matrix $\mathbf{M}$ may not be low-rank for real data.

Many previous works make use of multiple matrix norms to jointly characterize a matrix variate with complex distribution. Nevertheless, these strategies are limited to some special cases. For example, Least Soft-threshold Squares method (Wang, Lu, and Yang 2013) is only suitable for the variate following Gaussian-Laplace distribution, while Nuclear-L_1 joint regression model (Luo et al. 2015; 2016a) focuses on the structural and sparse matrix variate. To adapt more practical problems, some scholars carry out the variate estimation task under the framework of the Gaussian Mixture Regression (GMR) (Cao et al. 2015). This is originated from a basic fact: Gaussian Mixture distribution can construct a universal approximator to any continuous density function in theory (Bishop 2007). The experimental results show the advantages of this strategy, but most of these methods (related to GMR) are based on regression analysis. As we know, many practical problems cannot be formulated as regression-like models. Accordingly, it is expected to extend GMR to other formulations. Additionally, these GMR based

* To whom all correspondence should be addressed.

approaches assume the elements in a matrix variate are generated independently for the convenience, which overlooks the latent relationships between elements.

In this paper, we propose a novel metric learning model that utilizes the Gaussian Mixture Distribution (GMD) to automatically learn the structure information of the distance matrix $\mathbf{M}$ from the real data. To the best of our knowledge, this is the first work for depicting a matrix variate using GMD in metric learning model. To fully exploit the structure of $\mathbf{M}$, we partition $\mathbf{M}$ into blocks and view each block as a random matrix variate which is automatically fitted by GMD. Since each block represents the local structure of $\mathbf{M}$, our method integrates the structure information of different regions of $\mathbf{M}$ in modeling. On the basis of GMD, we construct a convex regularization with regard to $\mathbf{M}$ and use it to derive a robust metric learning model with triplet constraints. A new effective algorithm is introduced to solve the proposed model. Due to the promising generalization of GMD, our model does not rely on any assumption on $\mathbf{M}$, regardless of whether it holds the low-rank (or sparse) attribute or not. Therefore, comparing with existing metric learning methods using matrix norm regularizers, our method can effectively learn the structure information for the general distance matrix $\mathbf{M}$. Moreover, as the theoretical contribution of this paper, we analyze the convexity and generalization ability of the proposed model. A series of experiments on image classification and face verification demonstrate the effectiveness of our new method.

Learning A Robust Distance Metric Using Gaussian Mixture Distribution

In this section, we will first propose a robust objective for distance metric learning using grouped Gaussian mixture distribution. After that, we design an effective optimization algorithm to solve the proposed model.

Notation. Throughout this paper, we write matrices as bold uppercase characters and vectors as bold lowercase characters. Let $\mathbf{y} = \{y_1, y_2, \cdots, y_n\}$ be the label set of input (or training) samples $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$, where each $\mathbf{x}_i \in \mathcal{R}^d$ $(i = 1, 2, \cdots, n)$. For example, the label of sample $\mathbf{x}_i$ is y_i. Let $r(y_i, y_l) = 1$ if $y_i \neq y_l$ otherwise $r(y_i, y_l) = 0$. Suppose input samples $\mathbf{X}$ and labels $\mathbf{y}$ are contained in a input space $\mathcal{X}$ and a label space $\mathcal{Y}$, respectively. Meanwhile, we assume $\mathbf{z} := \{\mathbf{z}_i = (\mathbf{x}_i, y_i) : \mathbf{x}_i \in \mathcal{X}, y_i \in \mathcal{Y}, i \in \mathbf{N}_n\}$, where $\mathbf{N}_n = \{1, 2, \cdots, n\}$. For any $x \in \mathcal{R}$, the function $f(x) = [x]_+$ is equal to x if $x > 0$ and zero otherwise. $\mathcal{R}_+^{d\times d}$ defines the set of positive definite matrices on $\mathcal{R}^{d\times d}$. $\|\mathbf{M}\|_F$, $\mathbf{M}^T$ and Tr($\mathbf{M}$) denote the Frobenius-norm, transpose and trace of the matrix $\mathbf{M}$, respectively.

Problem statement. Given any two data points $\mathbf{x}_i$ and $\mathbf{x}_j$, a Mahalanobis distance between them can be calculated as following:

$$d_{\mathbf{M}}(\mathbf{x}_i,\ \mathbf{x}_j) = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^T \mathbf{M} (\mathbf{x}_i - \mathbf{x}_j)}. \tag{1}$$

The core task of metric learning is to learn an optimal positive semi-definite matrix $\mathbf{M}$ such that the distance between similar samples should be relatively smaller than between dissimilar samples. Meanwhile, a desired $\mathbf{M}$ is able to provide robustness to noise. To this end, the label information can be fully exploited, which leads to different weakly-supervised constraints, including pairwise, triplet and quadruplet constraints. However, the metric learning models with different constraints can be unified as the following form:

$$\mathbf{M}_{\mathbf{z}} = \underset{\mathbf{M}\in\mathcal{M}}{\operatorname{argmin}} \left(\varepsilon_{\mathbf{z}}(\mathbf{M}) + \lambda\Omega(\mathbf{M})\right), \tag{2}$$

where $\varepsilon_{\mathbf{z}}(\cdot)$ is called loss function, $\Omega(\mathbf{M})$ is a regularization term, the balance parameter $\lambda > 0$ and $\mathcal{M}$ denotes the domain of $\mathbf{M}$. In general, $\mathcal{M} \subseteq \mathcal{R}_+^{d\times d}$.

The loss function $\varepsilon_{\mathbf{z}}(\cdot)$ is generally induced by different constraints. Therefore, the minimization of $\varepsilon_{\mathbf{z}}(\cdot)$ will result in minimizing the distances between the data points with similar constraints and maximizing the distances between the data points with dissimilar constraints. Recently, the metric learning model: large margin nearest neighbor (LMNN) (Weinberger and Saul 2009) has attracted wide attention. It uses triplet constraints on training examples and the corresponding loss function can expressed as:

$$\begin{aligned}\varepsilon_{\mathbf{z}}^{lmnn}(\mathbf{M}) = (1-\mu) &\sum_{i,j\rightsquigarrow i} \mathcal{D}_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) \\ &+ \mu \sum_{i,j\rightsquigarrow i,l} r(y_i, y_l)[1 + \mathcal{D}_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) \\ &- \mathcal{D}_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_l)]_+,\end{aligned} \tag{3}$$

where $\mathcal{D}_{\mathbf{M}}(\mathbf{x}_i, \mathbf{x}_j) = d^2_{\mathbf{M}}(\mathbf{x}_i,\ \mathbf{x}_j)$ and the notation $j \rightsquigarrow i$ indicates that input $\mathbf{x}_j$ is a target neighbor (*i.e.*, the same labeled inputs) of input $\mathbf{x}_i$. When $r(y_i, y_l) = 1$, $\mathbf{x}_l$ is an impostor neighbor (*i.e.*, differently labeled inputs) of input $\mathbf{x}_i$. And the parameter $\mu \in (0\ \ 1)$ defines a trade-off between the above two objectives.

Although LMNN significantly improves the performance of traditional kNN classification, it is often confront with the overfitting problem. Then, some regularized LMNN models (Li, Tian, and Tao 2016; Lim, McFee, and Lanckriet 2013) have emerged, which enhance the generalization and robustness of LMNN. It should be noted that these methods only utilize some special matrix norm regularizers (*e.g.*, L_1-norm or nuclear norm) to constrain $\mathbf{M}$. In many practical problems, however, the distance matrix $\mathbf{M}$ of real data may be neither sparse nor low-rank.

The robust metric learning model induced by GMD. In the following, we provide a general regularization to characterize distance matrix $\mathbf{M}$ to adapt more practical applications. This regularization is induced by matrix variate Gaussian mixture distribution.

First, as shown in Fig.1, we partition the matrix $\mathbf{M}$ $(\in \mathcal{R}^{d\times d})$ into $p \times q$ blocks, where any two matrices do not contain the same elements. For each block $\mathbf{M}_{uv}$ $(\in \mathcal{R}^{d_u\times d_v})$, we preserve its matrix form. Each block $\mathbf{M}_{uv}$ $(\in \mathcal{R}^{d_u\times d_v})$ can be regarded as the local structure of $\mathbf{M}$, so our strategy actually merges different regions of local structures of $\mathbf{M}$.

Next, we use matrix variate Gaussian mixture distribution to fit each block. For the convenience, it is assumed that all blocks possess the same size. As a result, the Probability

$$\begin{pmatrix} m_{11} & m_{12} & \cdots & m_{1d} \\ m_{21} & m_{22} & \cdots & m_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ m_{d1} & m_{d2} & \cdots & m_{dd} \end{pmatrix} \Rightarrow \begin{pmatrix} \mathbf{M}_{11} & \mathbf{M}_{12} & \cdots & \mathbf{M}_{1q} \\ \mathbf{M}_{21} & \mathbf{M}_{22} & \cdots & \mathbf{M}_{2q} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{M}_{p1} & \mathbf{M}_{p2} & \cdots & \mathbf{M}_{pq} \end{pmatrix}$$

(a) (b)

Figure 1: Partition a matrix $\mathbf{M} = (m_{ij})_{d\times d}$ into $p \times q$ non-overlapping blocks: (a) the original matrix $\mathbf{M}$; (b) the partitioned result, where each $\mathbf{M}_{uv} \in \mathcal{R}^{d_u \times d_v}(u = 1, \cdots, p, v = 1, \cdots, q)$.

Density Function (PDF) of each block $\mathbf{M}_{uv}$ has the following form:

$$\mathcal{P}(\mathbf{M}_{uv}) = \sum_{k=1}^{R} \varrho_k \mathcal{N}(\mathbf{M}_{uv}|0, \mathbf{\Sigma}_k), \tag{4}$$

where R is the number of Gaussian components, ϱ_k denotes the mixing weight with $\varrho_k > 0$ and $\sum_{k=1}^{R} \varrho_k = 1$ and $\mathcal{N}(\mathbf{M}_{uv}|0, \mathbf{\Sigma}_k)$ is the zero-mean matrix variate Gaussian distribution with Σ_k denoting the covariance matrix, *i.e.*,

$$\mathcal{N}(\mathbf{M}_{uv}|0, \mathbf{\Sigma}_k) = \frac{1}{(2\pi)^{\frac{d_u d_v}{2}} |\mathbf{\Sigma_k}|^{d_v}} \exp(-\frac{1}{2}\text{tr}(\mathbf{M}_{uv}^T \mathbf{\Sigma}_k^{-1} \mathbf{M}_{uv})). \tag{5}$$

Finally, we assume all random matrix variates $\{\mathbf{M}_{uv}: u = 1, 2, \cdots, p, v = 1, 2, \cdots, q\}$ are independently and identically distributed. The PDF of $\mathbf{M}$ can be written as:

$$\mathcal{P}(\mathbf{M}) = \prod_{u=1}^{p} \prod_{v=1}^{q} \mathcal{P}(\mathbf{M}_{uv}). \tag{6}$$

The main task of traditional GMR is to search for a group of mixing weights $\boldsymbol{\varrho} = \{\varrho_1, \varrho_2, \cdots, \varrho_R\}$ and a group of covariance matrices $\mathbf{\Sigma} = \{\Sigma_1, \Sigma_2, \cdots, \Sigma_R\}$ such that the PDF of the regression error is maximized. In our problem, this means that the log likelihood function associated with the distance matrix $\mathbf{M}$

$$-\ln\mathcal{P}(\mathbf{M}) = -\sum_{u=1}^{p}\sum_{v=1}^{q} \ln\left(\sum_{k=1}^{R} \varrho_k \mathcal{N}(\mathbf{M}_{uv}|0, \mathbf{\Sigma}_k)\right) \tag{7}$$

is minimized.

The above function not only considers the two-dimensional group structure of $\mathbf{M}$, but also absorbs the advantages of GMD. If we set $R = p = q = 1$, $\pi_1 = 1$ and $\Sigma_1 = \mathbf{I}_{d\times d}$, where $\mathbf{I}_{d\times d}$ denotes a $d \times d$ identity matrix, then $-\ln\mathcal{P}(\mathbf{M})$ becomes the squared Frobenius-norm. As for other norms, *e.g.*, L_1-norm, group sparsity norm and nuclear norm (Li et al. 2016), there exist the corresponding segmentation strategies on $\mathbf{M}$ and parameters $\boldsymbol{\varrho}$ and $\mathbf{\Sigma}$ such that $-\ln\mathcal{P}(\cdot)$ can well approximate them (Maz'ya and Schmidt 1996). Therefore, $-\ln\mathcal{P}(\cdot)$ is more general as compared to these matrix norm regularizers.

Considering this merit of $-\ln\mathcal{P}(\cdot)$, we choose $\Omega(\mathbf{M}) = -\ln\mathcal{P}(\mathbf{M})$ and $\varepsilon_{\mathbf{z}}(\cdot) = \varepsilon_{\mathbf{z}}^{lmnn}(\cdot)$ in (2), which leads to the following metric learning model:

$$\begin{aligned} (\mathbf{M}, \boldsymbol{\varrho}, \mathbf{\Sigma}) &= \text{argmin}\left(\varepsilon_{\mathbf{z}}^{lmnn}(\mathbf{M}) - \lambda\ln\mathcal{P}(\mathbf{M})\right) \\ s.t.\ \mathbf{M} &\in \mathcal{M}, \boldsymbol{\varrho}_k > 0, \mathbf{\Sigma}_k \in \mathcal{R}_+^{d\times d}, k = 1, 2, \cdots, R \\ &\text{and} \sum_{k=1}^{R} \varrho_k = 1. \end{aligned} \tag{8}$$

Optimization algorithm for solving model (8). Comparing to the other metric learning models with matrix norm regularizers, solving model (8) is very challenging, because it needs to learn more model parameters and $-\ln\mathcal{P}(\cdot)$ is not separable for $\mathbf{\Sigma}$ and $\boldsymbol{\pi}$. Here, we regard the objective (8) as a Q-function (Zheng, Liu, and Ni 2014) in Bayesian learning and propose a simple yet effective Expectation-Maximization (EM) algorithm (Bishop 2007) to optimize it. This is divided into three steps as follows:

(1) **Initialization.** We need to initialize the distance matrix $\mathbf{M}$, mixing coefficients set $\boldsymbol{\varrho}$ and covariance matrices set $\mathbf{\Sigma}$.

(2) **E-step.** In this step, we compute the conditional expectation of latent variate $z_{uv,k}$ given $\mathbf{M}_{uv}$ by the Bayes' rule, *i.e.*,

$$z_{uv,k} = \frac{\varrho_k \mathcal{N}(\mathbf{M}_{uv}|\mathbf{0}, \mathbf{\Sigma}_k)}{\sum_{j=1}^{R} \varrho_j \mathcal{N}(\mathbf{M}_{uv}|\mathbf{0}, \mathbf{\Sigma}_j)}. \tag{9}$$

(3) **M-step.** We will find the optimal parameters $\mathbf{\Sigma}$, $\boldsymbol{\varrho}$ and $\mathbf{M}$ according to the current posterior probabilities.

Recalling our model (8), each optimal $\mathbf{\Sigma}_k$ can be obtained by solving the following problem:

$$\min_{\mathbf{\Sigma}_k \in \mathcal{R}_+^{d\times d}} -\sum_{u=1}^{p}\sum_{v=1}^{q} \ln\left(\sum_{k=1}^{R} \varrho_k \mathcal{N}(\mathbf{M}_{uv}|0, \mathbf{\Sigma}_k)\right). \tag{10}$$

Calculating the derivative objective (10) with respect to $\mathbf{\Sigma}_k$ and setting it as 0, we have

$$\mathbf{\Sigma_k} = \frac{1}{\sum_{u=1}^{p}\sum_{v=1}^{q} z_{uv,k}} (\sum_{u=1}^{p}\sum_{v=1}^{q} z_{uv,k} \mathbf{M}_{uv}\mathbf{M}_{uv}^T + \xi\mathbf{I}_{d_u\times d_v}), \tag{11}$$

where $\xi I_{d_u\times d_v}(\xi > 0)$ is a perturbed term insuring $\mathbf{\Sigma}_k$ is invertible.

To achieve the optimal $\varrho_k, k = 1, 2, \cdots, R$, we consider the optimization problem:

$$\min_{\varrho_k > 0} -\sum_{u=1}^{p}\sum_{v=1}^{q} \ln\left(\sum_{k=1}^{R} \varrho_k \mathcal{N}(\mathbf{M}_{uv}|0, \mathbf{\Sigma}_k)\right) + \alpha(\sum_{k=1}^{R} \varrho_k - 1), \tag{12}$$

where $\alpha > 0$ is a Lagrangian multiplier. Therefore,

$$\varrho_k = \frac{1}{pq}\sum_{u=1}^{p}\sum_{v=1}^{q} z_{uv,k}. \tag{13}$$

For the variate $\mathbf{M}$, we consider the following problem:

$$\mathbf{M} = \underset{\mathbf{M}\in\mathcal{M}}{\text{argmin}} \left(\varepsilon_{\mathbf{z}}^{lmnn}(\mathbf{M}) - \lambda\ln\mathcal{P}(\mathbf{M})\right). \tag{14}$$

It is difficult to find a closed-form solution of the problem (14). However, we can compute a sub-gradient of objective (14) with regard to $\mathbf{M}$:

$$\begin{aligned}\bigtriangledown_{\mathbf{M}} = (1-\mu)\sum_{i,j\rightsquigarrow i}\mathbf{C}_{ij} + \mu\sum_{(i,j,l)\in\mathcal{H}}(\mathbf{C}_{ij}-\mathbf{C}_{ij}) \\ + \frac{d(-\lambda \ln\mathcal{P}(\mathbf{M}))}{d\mathbf{M}},\end{aligned} \tag{15}$$

where $\mathbf{C}_{ij} = (\mathbf{x}_i - \mathbf{x}_j)(\mathbf{x}_i - \mathbf{x}_j)^T$ and the set of triples $\mathcal{H}$ is defined as: $(i,j,l) \in \mathcal{H}$ if and only if the indices (i,j,l) trigger the hinge loss in the second part of Eq. (3). In addition, the each block of $\frac{d(-\lambda \ln\mathcal{P}(\mathbf{M}))}{d\mathbf{M}}$ can be expressed as:

$$\frac{d(-\lambda \ln\mathcal{P}(\mathbf{M}))}{d\mathbf{M}_{uv}} = \frac{\sum_{k=1}^{R} GD_{uv,k}\boldsymbol{\Sigma}_k^{-1}}{\sum_{k=1}^{R} GD_{uv,k}}, \tag{16}$$

where $GD_{uv,k} = \frac{\varrho_k}{(2\pi)^{\frac{d_u d_v}{2}}|\boldsymbol{\Sigma_k}|^{d_v}}\exp(-\frac{1}{2}\mathrm{tr}(\mathbf{M}_{uv}^T\boldsymbol{\Sigma}_k^{-1}\mathbf{M}_{uv}))$.

Then, when $\boldsymbol{\Sigma}$ and $\boldsymbol{\varrho}$ are fixed, the optimal $\mathbf{M}$ can be found by solving model (14) using subgradient method (Boyd, Xiao, and Mutapcic 2003):

$$\mathbf{M} \Leftarrow P_{\mathcal{R}_+^{d\times d}}(\mathbf{M} - \gamma \bigtriangledown \mathbf{M}), \tag{17}$$

where $P_{\mathcal{R}_+^{d\times d}}(\cdot)$ denotes the projection operator and $\gamma > 0$ is a step size.

The whole iterative procedure for solving problem (8) is summarized in Algorithm 1.

Remark 1. Computing the sub-gradient (15) is extremely expensive since the set of triples $\mathcal{H}$ is potentially very large. To circumvent this problem, we can use the similar technique as in (Weinberger and Saul 2009) to reduce the computation complexity of (17).

There have been many results to show the convergence of EM algorithm. Here we refer the readers to (Gupta, Chen, and others 2011)). In Algorithm 1, we use the following convergence conditions:

$$\| \boldsymbol{\Sigma}_k^{new} - \boldsymbol{\Sigma}_k^{old} \|_F \leq \theta \text{ and } |\varrho_k^{new} - \varrho_k^{old}| \leq \theta, \tag{18}$$

where θ is a sufficiently small positive number.

If the final distance matrix $\mathbf{M}$ is obtained by Algorithm 1, then by Eq. (1), we can construct the Mahalanobis distance to compute similarity of two samples in the experiments.

Theoretical Analysis

In this section, we investigate the convexity and the generalization ability of model (14).

Theorem 1. The optimization problem (14) is convex.

Proof. By (Parameswaran and Weinberger 2010), we know that $\varepsilon_{\mathbf{z}}^{lmnn}(\mathbf{M})$ is convex. Thus, it suffices to show that $-\ln\mathcal{P}(\mathbf{M})$ is convex. Since each $f_k(\mathbf{M}) = -\mathrm{tr}(\mathbf{M}^T\boldsymbol{\Sigma}_k\mathbf{M})$ is a concave matrix function and the $g(x) = e^x$ is monotonically increasing and convex, then each $h_k(\mathbf{M}) = c_k e^{-\mathrm{tr}(\mathbf{M}^T\boldsymbol{\Sigma}_k\mathbf{M})}$ is a concave matrix function. Therefore, $h_1(\mathbf{M}) + h_2(\mathbf{M}) + \cdots + h_R(\mathbf{M})$ is still concave. Considering that $-\ln(\cdot)$ is a monotonically decreasing function, it is known that the regularization $-\ln\mathcal{P}(\mathbf{M})$ is convex. □

Algorithm 1 Solving Model (8) via EM

Input: training samples $\mathbf{X}$, parameters λ, components number R and threshold value ξ.
Initialization: covariance matrices set $\boldsymbol{\Sigma} = \{\boldsymbol{\Sigma}_1, \boldsymbol{\Sigma}_2, \cdots, \boldsymbol{\Sigma}_R\}$ coefficients $\boldsymbol{\varrho} = \{\varrho_1, \varrho_2, \cdots, \varrho_R\}$ and the initial distance matrix $\mathbf{M}$.
Output: the final parameters $\boldsymbol{\Sigma}$, $\boldsymbol{\varrho}$ and distance matrix $\mathbf{M}$.
repeat

1. (E-step for $z_{uv,k}$): Update the posterior probability $z_{uv,k}$ by

$$z_{uv,k}^{new} \Leftarrow \frac{\varrho_k^{old}\mathcal{N}(\mathbf{M}_{uv}^{old}|\mathbf{0}, \boldsymbol{\Sigma}_k^{old})}{\sum_{j=1}^{R}\varrho_j^{old}\mathcal{N}(\mathbf{M}_{uv}^{old}|\mathbf{0}, \boldsymbol{\Sigma}_j^{old})}.$$

2. (M-step for $\boldsymbol{\Sigma}_k$): Update each $\boldsymbol{\Sigma}_k$ by

$$\boldsymbol{\Sigma_k}^{new} \Leftarrow \frac{1}{\sum_{u=1}^{p}\sum_{v=1}^{q} z_{uv,k}^{old}}(\mathbf{W} + \xi I_{d_u\times d_v}),$$

where $\mathbf{W} = \sum_{u=1}^{p}\sum_{v=1}^{q} z_{uv,k}^{old}\mathbf{M}_{uv}^{old}\mathbf{M}_{uv}^{old^T}$.

3. (M-step for ϱ_k): Update each ϱ_k by

$$\varrho_k^{new} \Leftarrow \frac{1}{pq}\sum_{u=1}^{p}\sum_{v=1}^{q} z_{uv,k}^{old}.$$

4. (M-step for $\mathbf{M}$): Update $\mathbf{M}$ by using subgradient method (17) to solve problem (14).

until *converge*.

Remark 2. Connecting (Boyd, Xiao, and Mutapcic 2003) and Theorem 1, it is known that iteration (17) is convergent.

Before starting the generalization guarantee of model (14), some definitions and Lemmas needs to be introduced. We assume that the instance space $\mathcal{X}$ is a compact convex with respect to L_2-norm, *i.e.*, there exists a constant $\tau > 0$ such that $\forall\ \mathbf{x} \in \mathcal{X}, \|\mathbf{x}\|_2 \leq \tau$. Given a training sample $\mathcal{T} = \{\mathbf{z}_i = (\mathbf{x}_i, y_i)\}_{i=1}^n$ drawn i.i.d. from an unknown joint distribution P over the space $\mathcal{Z}$, we denote by $\mathcal{P}_\mathcal{T}$ the set of all possible pair built from $\mathcal{T}$:

$$\mathcal{P}_\mathcal{T} = \{(\mathbf{z}_1, \mathbf{z}_1), \cdots, (\mathbf{z}_1, \mathbf{z}_n), \cdots, (\mathbf{z}_n, \mathbf{z}_n)\}. \tag{19}$$

For the convenience, we simplify triplet constraints in (14) as the pairwise constraints, and consider the more general metric learning model:

$$\mathbf{M} = \underset{\mathbf{M}\in\mathcal{M}}{\mathrm{argmin}}\left(\frac{1}{|\mathcal{P}_\mathcal{T}|}\sum_{(\mathbf{z}_i,\mathbf{z}_j)\in\mathcal{P}_\mathcal{T}}\omega(\mathbf{M}, \mathbf{z}_i, \mathbf{z}_j) - \lambda\ln\mathcal{P}(\mathbf{M})\right), \tag{20}$$

where $|\mathcal{P}_\mathcal{T}|$ denotes the element number of $\mathcal{P}_\mathcal{T}$ and the loss function $\omega(\mathbf{M}, \mathbf{z}_i, \mathbf{z}_j)$ depends on the input examples and their labels.

We call optimization problem (20) as the metric learning algorithm $\mathcal{A}$, which takes as input a finite set of pairs from $(\mathcal{Z}\times\mathcal{Z})^n$ and outputs a metric.

Definition 1. (Bellet 2013) The model (20) (or $\mathcal{A}$) is $(S, \zeta(\cdot))$-robust for $S \in \mathbf{N}$ and $\zeta(\cdot) : (\mathcal{Z}\times\mathcal{Z})^n \to \mathcal{R}$ if $\mathcal{Z}$ can be partitioned into S disjoints sets, denoted by $\{C_i\}_{i=1}^S$, such that the following holds for all $\mathcal{T} \in \mathcal{Z}^n$:

Databases	1NN	SVM	LMNN	LMNN_Trace	LMNN_Fantope	LMNN_Cap	Our method
FGNET	10.412	42.647	47.653	54.711	55.286	57.817	**59.812**
VINIR	33.474	68.850	69.750	69.832	73.600	73.731	**74.428**
OSR	65.474	75.850	75.315	76.021	76.471	76.511	**77.493**
PubFig	67.556	83.797	86.353	86.411	86.576	87.125	**87.883**

Table 1: The classification accuracies (%) of all methods on the FGNET Aging, VINIR, OSR, and PubFig databases

$\forall(\mathbf{z}_1, \mathbf{z}_2) \in \mathcal{P}_\mathcal{T}, \mathbf{z}, \mathbf{z}' \in \mathcal{Z}$ and $i, j \in [S]$: if $\mathbf{z}_1, \mathbf{z} \in C_i$ and $\mathbf{z}_2, \mathbf{z}' \in C_j$ then

$$|\omega(\mathbf{M}_{\mathcal{P}_\mathcal{T}}, \mathbf{z}_1, \mathbf{z}_2) - \omega(\mathbf{M}_{\mathcal{P}_\mathcal{T}}, \mathbf{z}_1, \mathbf{z}_2)| \leq \zeta(\mathcal{P}_\mathcal{T}), \quad (21)$$

where $\mathbf{M}_{\mathcal{P}_\mathcal{T}}$ is learned by model (20) and $\zeta(\mathcal{P}_\mathcal{T}) > 0$.

We further assume that $\omega(\mathbf{M}, \mathbf{z}_i, \mathbf{z}_j) = g(y_i y_j[1 - \mathcal{D}_\mathbf{M}(\mathbf{x}_i, \mathbf{x}_j)])$, where $g(\cdot)$ is nonnegative and Lipschitz continuous with Lipschitz constant L.

Lemma 1. If σ_k is the minimum eigenvalue of $\mathbf{\Sigma}_k$, then $\text{tr}(\mathbf{M}^T\mathbf{\Sigma}_k\mathbf{M}) \geq \sigma_k \text{tr}(\mathbf{M}^T\mathbf{M})$.

Proof. Let the eigenvalue decomposition of $\mathbf{\Sigma}_k$ be $\mathbf{U}_k^T\mathbf{\Delta}\mathbf{U}_k$, it can be seen that $\text{tr}[\mathbf{M}\mathbf{M}^T\mathbf{U}_k^T(\mathbf{\Delta}_k - \sigma_k\mathbf{I}_{d\times d})\mathbf{U}_k] \geq 0$. This implies that $\text{tr}(\mathbf{M}^T\mathbf{\Sigma}_k\mathbf{M}) \geq \sigma_k\text{tr}(\mathbf{M}^T\mathbf{M})$. □

Theorem 2. Model (20) is $(|\mathcal{Y}|\mathcal{N}(\gamma/2, \mathcal{X}, \|\cdot\|_2), \frac{8L\varepsilon_0\tau\gamma}{\sqrt{\lambda\varrho}})$-robust, where $\mathcal{N}(\gamma/2, \mathcal{X}, \|\cdot\|_2)$ is the $\gamma/2$-covering number of $\mathcal{X}$ (Bellet 2013), $\varepsilon_0 = \varepsilon_\mathbf{z}(\mathbf{0}) - \lambda\ln\mathcal{P}(\mathbf{0})$ and $\gamma > 0$.

Proof. Let $\varepsilon(\mathbf{M}) = \frac{1}{|\mathcal{P}_\mathcal{T}|}\sum_{(\mathbf{z}_i,\mathbf{z}_j)\in\mathcal{P}_\mathcal{T}} \omega(\mathbf{M}, \mathbf{z}_i, \mathbf{z}_j)$ and $\mathbf{M}^*$ be the optimal solution of model (20), then we have

$$\varepsilon_\mathbf{z}(\mathbf{M}^*) - \lambda\ln\mathcal{P}(\mathbf{M}^*) \leq \varepsilon_\mathbf{z}(\mathbf{0}) - \lambda\ln\mathcal{P}(\mathbf{0}) = \varepsilon_0, \quad (22)$$

which leads to $-\ln\mathcal{P}(\mathbf{M}^*) \leq \frac{\varepsilon_0}{\lambda}$.

We partition $\mathcal{Z}$ as $|\mathcal{Y}|\mathcal{N}(\gamma/2, \mathcal{X}, \|\cdot\|_2)$ sets such that if $\mathbf{z}$ and $\mathbf{z}'$ belong to the same set then $y = y'$ and $\|\mathbf{x} - \mathbf{x}'\| \leq \gamma$. Now, for $\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_1', \mathbf{z}_2' \in \mathcal{Z}$, if $y_1 = y_1'$ and $y_2 = y_2'$, then $\|\mathbf{x}_1 - \mathbf{x}_1'\| \leq \gamma$ and $\|\mathbf{x}_2 - \mathbf{x}_2'\| \leq \gamma$. Thus,

$$\begin{aligned}
&|g(y_1y_2[1 - \mathcal{D}_{\mathbf{M}^*}(\mathbf{x}_1, \mathbf{x}_2)]) - g(y_1'y_2'[1 - \mathcal{D}_{\mathbf{M}^*}(\mathbf{x}_1', \mathbf{x}_2')])| \\
\leq& L(\|\mathbf{x}_1 - \mathbf{x}_2\|_2\|\mathbf{M}\|_F\|\mathbf{x}_1 - \mathbf{x}_1'\|_2 \\
&+ \|\mathbf{x}_1 - \mathbf{x}_2\|_2\|\mathbf{M}\|_F\|\mathbf{x}_2 - \mathbf{x}_2'\|_2 \\
&+ \|\mathbf{x}_1 - \mathbf{x}_1'\|_2\|\mathbf{M}\|_F\|\mathbf{x}_1 - \mathbf{x}_2'\|_2 \\
&+ \|\mathbf{x}_2 - \mathbf{x}_2\|_2\|\mathbf{M}\|_F\|\mathbf{x}_1 - \mathbf{x}_2'\|_2).
\end{aligned} \quad (23)$$

On the other hand, denoting $\sigma = \min\{\sigma_1, \sigma_2, \cdots, \sigma_k\}$ and $\varrho = \min\{\varrho_1, \varrho_2, \cdots, \varrho_k\}$ and considering Lemma 1, we have

$$\begin{aligned}
-\ln\mathcal{P}(\mathbf{M}^*) &\geq -\ln R\varrho e^{-\sigma\text{tr}(\mathbf{M}^T\mathbf{M})} \\
&= -\ln R\varrho + \sigma\text{tr}(\mathbf{M}^T\mathbf{M}).
\end{aligned} \quad (24)$$

Noticing $-\ln R\varrho \geq 0$, we further get

$$\sqrt{-\frac{1}{\sigma}\ln\mathcal{P}(\mathbf{M}^*)} \geq \sqrt{\text{tr}(\mathbf{M}^T\mathbf{M})} = \|\mathbf{M}\|_F. \quad (25)$$

Combining (23) and (25), we have

$$\begin{aligned}
&|g(y_1y_2[1 - \mathcal{D}_{\mathbf{M}^*}(\mathbf{x}_1, \mathbf{x}_2)]) - g(y_1'y_2'[1 - \mathcal{D}_{\mathbf{M}^*}(\mathbf{x}_1', \mathbf{x}_2')])| \\
\leq& \frac{8L\varepsilon_0\tau\gamma}{\sqrt{\lambda\varrho}}.
\end{aligned} \quad (26)$$

According to Definition 1, we can complete the proof of Theorem 2. □

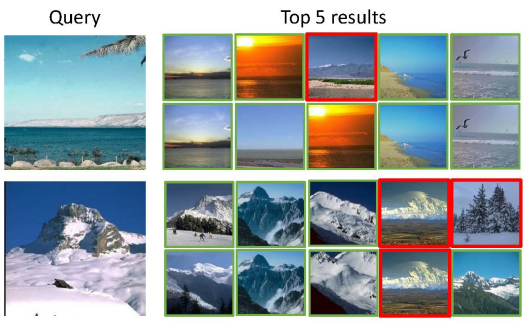
In this paper, this loss function $\omega(\mathbf{M}, \mathbf{z}_i, \mathbf{z}_j)$ is assumed to be nonnegative and uniformly bounded by a constant B ($B > 0$). Denote $G^\omega_{\mathcal{P}_\mathcal{T}} = \frac{1}{|\mathcal{P}_\mathcal{T}|}\sum_{(\mathbf{z}_i,\mathbf{z}_j)\in\mathcal{P}_\mathcal{T}} \omega(\mathbf{M}, \mathbf{z}_i, \mathbf{z}_j)$, $\tau = \frac{8L\varepsilon_0\tau\gamma}{\sqrt{\lambda\varrho}}$, $K = |\mathcal{Y}|\mathcal{N}(\gamma/2, \mathcal{X}, \|\cdot\|_2)$ and $G^\omega = \mathcal{E}_{\mathbf{z},\mathbf{z}'\sim\mathcal{P}}[\omega(\mathbf{M}, \mathbf{z}_i, \mathbf{z}_j)]$, where $\mathcal{E}(\cdot)$ defines the Expectation function. Using Theorem 2 and Theorem 7.3 in (Bellet 2013), we can easily obtain the generalization bound of model (20), *i.e.*, for any $\delta > 0$, with probability at least $1-\delta$ we have:

$$|G^\omega(\mathbf{M}_{\mathcal{P}_\mathcal{T}}) - G^\omega_{\mathcal{P}_\mathcal{T}}(\mathbf{M}_{\mathcal{P}_\mathcal{T}})| \leq \left[\tau + 2B\sqrt{\frac{2K\ln 2 + 2\ln(1/\delta)}{n}}\right], \quad (27)$$

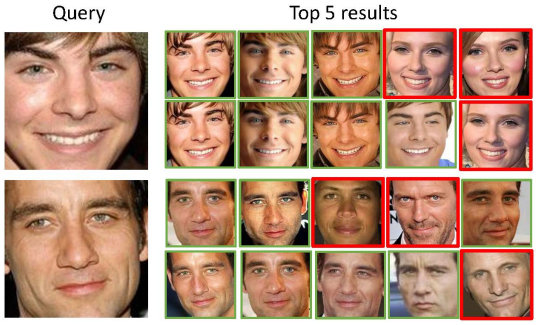
where n denotes the number of training samples.

Experiments

In this section, five standard databases, including FGNET Aging database (Lanitis, Taylor, and Cootes 2002), Visible (VI) and Near-Infrared (NIR) Face Database (Shen et al. 2011), OSR database (Parikh and Grauman 2011), PubFig database (Kumar et al. 2009) and LFW database (Huang et al. 2007), are selected to evaluate the effectiveness of our method. Especially, we implement image classification experiments on FGNET Aging, VI and NIR Face, OSR and PubFig databases, and compare our method with kNN, SVM (Fan et al. 2008), LMNN (Weinberger and Saul 2009), LMNN_Trace (Huo, Nie, and Huang 2016), LMNN_Fantope (Law, Thome, and Cord 2014) and LMNN_Cap (Huo, Nie, and Huang 2016). Meanwhile, we carry out face verification experiments on PubFig and LFW databases, and compare our method with LMNN_Cap, LMNN_Fantope, KISSME (Koestinger et al. 2012), ITML (Davis et al. 2007), LDML (Guillaumin, Verbeek, and Schmid 2009), IDENTITY (Huo, Nie, and Huang 2016) and MAHAL (Huo, Nie, and Huang 2016). For our method, we set $\lambda = 0.001, \xi = 10^{-5}$ and $R = 5$. For other compared methods, we follow the authors' suggestions to choose the optimal parameters.



(a) Results of 5 nearest neighbors when we query an image on OSR dataset. The first row shows the results of LMNN_Cap, and the second row shows the results of our method.



(b) Results of 5 nearest neighbors when we query an image on Pubfig dataset. The first row shows the results of LMNN_Cap, and the second row shows the results of our method.

Figure 2: Results of 5 nearest neighbors when we query an image. Green line means this neighbor is in the same class with query image, and red line denotes they are different.

Experiments on the FGNET Aging Database

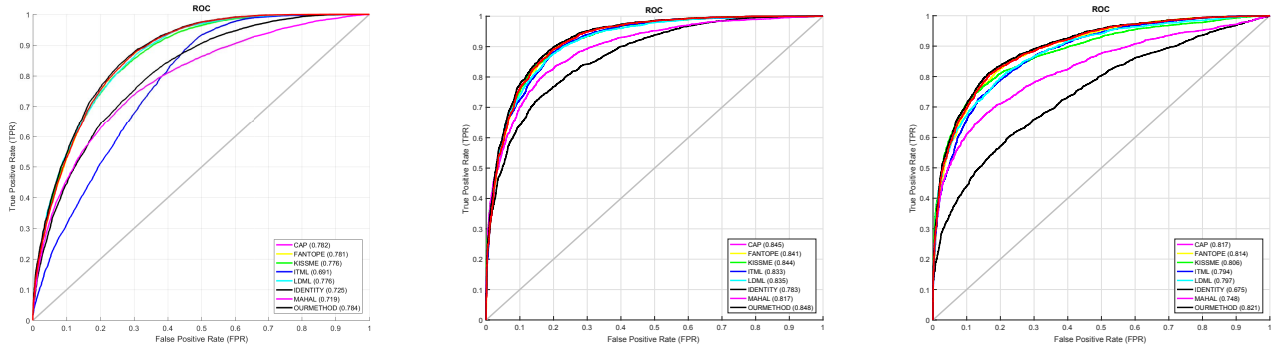
We implement an experiment on the FGNET Aging Database. There are 1, 002 face images from 82 subjects in this database. Each subject has 6-18 face images at different ages. Each image is labeled by its chronological age. The ages are distributed in a wide range from 0 to 69. Besides age variation, most of the age-progressive image sequences display other types of facial variations, such as significant changes in pose, illumination, expression, *etc*. To adapt some metric learning methods, a subset of FGNET database is chosen. It includes 680 face images from 68 subjects. Each subject has 10 face images. We manually cropped the face portion of the image and then normalized it to 32$\times$ 32 pixels. Then, we choose randomly five face images from each person as training samples, while the remaining five images are utilized as test samples. This experiment is repeated ten times. We calculate the average recognition rates (%) and standard deviations of all methods as shown in the second line of Table 1. For$k$NN, we set $k = 1$ as in (Huo, Nie, and Huang 2016). From Table 1, we can find that the results of some metric learning methods with regularization terms such as LMNN_Trace (54.711%), LMNN_Fantope (55.29%) and LMNN_Cap (57.817%) perform better than the other methods. Since FGNET Aging database not only includes age variance, but also face pose changes, it is very challenging to correctly characterize the distance matrix $\mathbf{M}$. However, our method (59.812%) still has some advantages in handling these variances. As a result, comparing to simple matrix norm regularizers, the proposed regularization (7) can fit the distance matrix $\mathbf{M}$ better.

Experiments on the VI and NIR Face Database

An experiment is performed on the Visible (VI) and Near-Infrared (NIR) Face Database (Shen et al. 2011). The NIR and VI database is collected using developed dual camera system from 215 subjects. Subjects faces were captured in four different settings, *i.e.*, expression variations, pose variations and illumination/time variations. In my experiment, 800 face images from 80 subjects on this database are adopted. Each subject has 10 face images which include five visible and five near-infrared face images. We manually cropped the face portion of the image and then normalized it to 32$\times$ 32 pixels. Then, we choose randomly five face images from each person as training samples, while the remaining five images are utilized as test samples. This experiment is repeated ten times. We calculate the average recognition rates (%) of all methods as shown in the third line of Table 1. From this table, it is observed that the results of all methods are similar. Particularly, the result of SVM (68.850%) is encouraging. The classification accuracies of other LMNN based methods are less than 74%, while the proposed method achieves the highest accuracy: 74.428%. This means that our method can effectively handle face images from two sensor types.

Experiments on the OSR Database

In this experiment, we utilize Outdoor Scene Recognition (OSR) dataset. It includes 2688 images from 8 scene categories, which are described by high level attribute features. The same experimental setting as (Huo, Nie, and Huang 2016) is adopted. That is, 30 images for each category are chosen as training data, and other images are used as testing data. To verify the robustness of our methods, we randomly select 30 images for each category as training data, and other images are used as testing data. This procedure is repeated 5 times. We compute the average accuracies of the compared methods, including kNN, SVM, LMNN, LMNN_Trace norm, LMNN_Fantope, LMNN_capped norm and our method, which are listed in the forth line of Table 1. It is observed that the performance of 1NN is poor since it only involves an Euclidean metric. The results of the other methods associated with LMNN are comparative, because they take advantage of Mahalanobis distance, which admits arbitrary linear scalings and rotations of the feature space. Some regularized LMNN methods, *e.g.*, LMNN_Trace norm, LMNN_Fantope



(a) Pubfig dataset (b) Attribute Feature (LFW dataset) (c) SIFT Feature (LFW dataset)

Figure 3: ROC curves of face verification on Pubfig and LFW datasets

and LMNN_capped norm, are superior to LMNN. However, our method (77.493%) performs better than other methods. Accordingly, using Gaussian mixture distribution induced regularization to characterize distance matrix $\mathbf{M}$ is more appropriate than matrix norm regularizers in real applications.

Experiments on the PubFig Database

We design two experiments on the PubFig Face Dataset. In the first experiment, we choose a subset of Pubfig database, which includes 771 images from 8 face categories. The experimental setting is similar to (Law, Thome, and Cord 2014). But here, a 512-dimensional DSIFT (Cheung and Hamarneh 2009) descriptor is used for the better performance. This experiment is run 5 times, where 30 images per person in training data are selected randomly each time, and the average classification accuracies are utilized as the evaluation criterion. We compare our method with some related methods, such as 1NN, SVM, LMNN, LMNN_Trace, LMNN_Fantope and LMNN_Cap. The results of all method are exhibited in the fifth line of Table 1. It can be seen that our method achieves the best result: 88.083%. Meanwhile, some methods based on LMNN, including LMNN_trace (86.411%), LMNN_fantope (86.576%) and LMNN_cap (87.125%), also give the good performance. The above results shows that our method is effective to the face images under uncontrolled environment.

The second experiment focuses on face verification task using face verification benchmark dataset, which consists of 20,000 pairs of images of 140 people from PubFig Face Dataset. The data is divided into 10 folds with mutually disjoint sets of 14 people, and each fold contains 1,000 intra and 1,000 extra-personal pairs. Our method is compared with some recent methods and the Equal Error Rates (EER) of all method are computed. The ROC curves of all methods are shown in Fig.3(a). It is evident that our method has a leading performance. For example, the 1-EER values of LMNN_Cap (Huo, Nie, and Huang 2016), Fantope, KISSME (Koestinger et al. 2012), ITML (Davis et al. 2007), LDML (Guillaumin, Verbeek, and Schmid 2009), IDENTITY (Huo, Nie, and Huang 2016), MAHAL (Huo, Nie, and Huang 2016) and our method are 0.782, 0.781, 0.766, 0.725, 0.719 and 0.784, respectively. Thus, how to fit the distance $\mathbf{M}$ has an important effect on the face verification performance.

Experiments on the LFW Database

In this subsection, the face verification experiments are carried out on the Labeled Faces in the Wild (LFW) dataset. It contains 13,233 unconstrained face images of 5749 individuals, and 1680 of these pictured people appear in two or more distinct photos. We adopt two different feature representations, *i.e.*, LFW Attribute feature dataset and LFW SIFT feature dataset. The experiment setting is similar to (Huo, Nie, and Huang 2016). We plot ROC curves of all methods, including LMNN_Cap, Fantope, KISSME, ITML, LDML, Identity, MAHAL in Figure 2(a) and 1(b). Meanwhile, the Equal Error Rate for each method is calculated and the 1-EER value is used to evaluate the performance of these methods. Figure 3(b) shows the results on LFW Attribute feature dataset. It is seen that Mahalanobis distance based methods perform better than Euclidean distance. Comparing with Identity and Mahalanobis methods, the advantage of KISSME is obvious. The performance of Mahalanobis distance with structural regularizers is competitive. For example, LMNN_Cap and Fantope reach 84.5% and 84.1%, respectively. For SIFT Feature dataset, the similar phenomenon can be found (see Figure 3(c)). In a word, our method consistently outperforms other methods.

Conclusions

In this paper, we propose a robust metric learning model with triplet constraints. Our method partitions the distance matrix $\mathbf{M}$ into several blocks and uses Matrix Variate Gaussian Mixture Distribution to fit each block. Due to the outstanding generalization of Gaussian Mixture Distribution, the proposed method can deal with more practical cases, where the distribution of $\mathbf{M}$ is complex and irregular. The proposed model is solved via EM algorithm. Additionally, we provide the theoretical analysis for our method. Empirical experiments on several real-world datasets demonstrate the robustness of the proposed model.

Acknowledgement

This work was partially supported by the following grants: NSF-IIS 1302675, NSF-IIS 1344152, NSF-DBI 1356628, NSF-IIS 1619308, NSF-IIS 1633753, NIH R01 AG049371.

References

Bellet, A. 2013. Supervised metric learning with generalization guarantees. *arXiv preprint arXiv:1307.4514*.

Bishop, C. 2007. Pattern recognition and machine learning (information science and statistics), 1st edn. 2006. corr. 2nd printing edn. *Springer, New York*.

Boyd, S.; Xiao, L.; and Mutapcic, A. 2003. Subgradient methods. *lecture notes of EE392o, Stanford University, Autumn Quarter* 2004.

Cao, X.; Chen, Y.; Zhao, Q.; Meng, D.; Wang, Y.; Wang, D.; and Xu, Z. 2015. Low-rank matrix factorization under general mixture noise distributions. In *Proceedings of the IEEE International Conference on Computer Vision*, 1493–1501.

Cheung, W., and Hamarneh, G. 2009. n-sift: n-dimensional scale invariant feature transform. *IEEE Transactions on Image Processing* 18(9):2012–2021.

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

Fan, R.-E.; Chang, K.-W.; Hsieh, C.-J.; Wang, X.-R.; and Lin, C.-J. 2008. Liblinear: A library for large linear classification. *Journal of machine learning research* 9(Aug):1871–1874.

Guillaumin, M.; Verbeek, J.; and Schmid, C. 2009. Is that you? metric learning approaches for face identification. In *Computer Vision, 2009 IEEE 12th international conference on*, 498–505. IEEE.

Gupta, M. R.; Chen, Y.; et al. 2011. Theory and use of the em algorithm. *Foundations and Trends® in Signal Processing* 4(3):223–296.

Huang, G. B.; Ramesh, M.; Berg, T.; and Learned-Miller, E. 2007. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report, Technical Report 07-49, University of Massachusetts, Amherst.

Huo, Z.; Nie, F.; and Huang, H. 2016. Robust and effective metric learning using capped trace norm: Metric learning via capped trace norm. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1605–1614. ACM.

Koestinger, M.; Hirzer, M.; Wohlhart, P.; Roth, P. M.; and Bischof, H. 2012. Large scale metric learning from equivalence constraints. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2288–2295. IEEE.

Kumar, N.; Berg, A. C.; Belhumeur, P. N.; and Nayar, S. K. 2009. Attribute and simile classifiers for face verification. In *Computer Vision, 2009 IEEE 12th International Conference on*, 365–372.

Kuznetsova, A.; Hwang, S. J.; Rosenhahn, B.; and Sigal, L. 2016. Exploiting view-specific appearance similarities across classes for zero-shot pose prediction: A metric learning approach. In *AAAI*, 3523–3529.

Lanitis, A.; Taylor, C. J.; and Cootes, T. F. 2002. Toward automatic simulation of aging effects on face images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24(4):442–455.

Law, M. T.; Thome, N.; and Cord, M. 2014. Fantope regularization in metric learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1051–1058.

Li, J.; Kong, Y.; Zhao, H.; Yang, J.; and Fu, Y. 2016. Learning fast low-rank projection for image classification. *IEEE Transactions on Image Processing* 25(10):4803–4814.

Li, Y.; Tian, X.; and Tao, D. 2016. Regularized large margin distance metric learning. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on*, 1015–1022. IEEE.

Lim, D.; McFee, B.; and Lanckriet, G. R. 2013. Robust structural metric learning. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*, 615–623.

Lu, J.; Zhou, X.; Tan, Y.-P.; Shang, Y.; and Zhou, J. 2014. Neighborhood repulsed metric learning for kinship verification. *IEEE transactions on pattern analysis and machine intelligence* 36(2):331–345.

Luo, L.; Yang, J.; Qian, J.; and Tai, Y. 2015. Nuclear-l1 norm joint regression for face reconstruction and recognition with mixed noise. *Pattern Recognition* 48(12):3811–3824.

Luo, L.; Chen, L.; Yang, J.; Qian, J.; and Zhang, B. 2016a. Tree-structured nuclear norm approximation with applications to robust face recognition. *IEEE Transactions on Image Processing* 25(12):5757–5767.

Luo, L.; Yang, J.; Qian, J.; Tai, Y.; and Lu, G.-F. 2016b. Robust image regression based on the extended matrix variate power exponential distribution of dependent noise. *IEEE Transactions on Neural Networks and Learning Systems*.

Maz'ya, V., and Schmidt, G. 1996. On approximate approximations using gaussian kernels. *IMA Journal of Numerical Analysis* 16(1):13–29.

Mignon, A., and Jurie, F. 2012. Pcca: A new approach for distance learning from sparse pairwise constraints. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2666–2672. IEEE.

Parameswaran, S., and Weinberger, K. Q. 2010. Large margin multi-task metric learning. In *Advances in neural information processing systems*, 1867–1875.

Parikh, D., and Grauman, K. 2011. Relative attributes. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, 503–510. IEEE.

Shen, L.; He, J.; Wu, S.; and Zheng, S. 2011. Face recognition from visible and near-infrared images using boosted directional binary code. In *International Conference on Intelligent Computing*, 404–411. Springer.

Wang, D.; Lu, H.; and Yang, M.-H. 2013. Least soft-threshold squares tracking. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2371–2378.

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

Ying, Y.; Huang, K.; and Campbell, C. 2009. Sparse metric learning via smooth optimization. In *Advances in neural information processing systems*, 2214–2222.

Zheng, F., and Shao, L. 2016. Learning cross-view binary identities for fast person re-identification. In *IJCAI*, 2399–2406.

Zheng, J.; Liu, S.; and Ni, L. M. 2014. Robust bayesian inverse reinforcement learning with sparse behavior noise. In *AAAI*, 2198–2205.

Zheng, F.; Tang, Y.; and Shao, L. 2016. Hetero-manifold regularisation for cross-modal hashing. *IEEE transactions on pattern analysis and machine intelligence*.