Robust and Effective Metric Learning Using Capped Trace Norm

Zhouyuan Huo
Department of Computer Science and Engineering
University of Texas at Arlington
Texas, USA
huozhouyuan@gmail.com

Feiping Nie
Department of Computer Science and Engineering
University of Texas at Arlington
Texas, USA
feipingnie@gmail.com

Heng Huang[*]
Department of Computer Science and Engineering
University of Texas at Arlington
Texas, USA
heng@uta.edu

ABSTRACT

Metric learning aims at automatically learning a metric from pair or triplet based constraints in data, and it can be potentially beneficial whenever the notion of metric between instances plays a nontrivial role. In Mahalanobis distance metric learning, distance matrix M is in symmetric positive semi-definite cone, and in order to avoid overfitting and to learn a better Mahalanobis distance from weakly supervised constraints, the low-rank regularization has been often imposed on matrix M to learn the correlations between features and samples. As the approximations of the rank minimization function, the trace norm and Fantope have been utilized to regularize the metric learning objectives and achieve good performance. However, these low-rank regularization models are either not tight enough to approximate rank minimization or time-consuming to tune an optimal rank. In this paper, we introduce a novel metric learning model using the capped trace norm based regularization, which uses a singular value threshold to constraint the metric matrix M as low-rank explicitly such that the rank of matrix M is stable when the large singular values vary. The capped trace norm regularization can also be viewed as the adaptive Fantope regularization. We minimize singular values which are less than threshold value and the rank of M is not necessary to be k, thus our method is more stable and applicable in practice when we do not know the optimal rank of matrix M. We derive an efficient optimization algorithm to solve the proposed new model and the algorithm convergence proof is also provided in this paper. We evaluate our method on a variety of challenging benchmarks, such as LFW and Pubfig datasets. Face verification experiments are performed and results show that our method consistently outperforms the state-of-the-art metric learning algorithms.

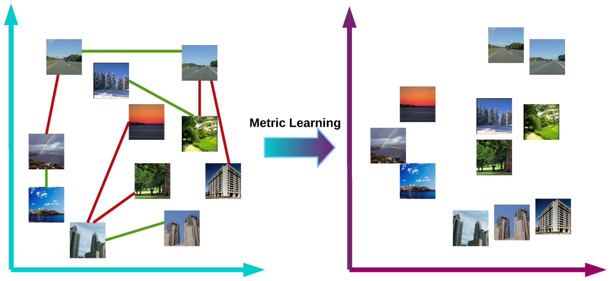


Figure 1: An illustration of metric learning on Outdoor Scene Recognition (OSR) dataset.

CCS Concepts

•Information systems → Data mining; •Computing methodologies → *Machine learning;*

Keywords

Metric Learning; Capped Trace Norm; Low-Rank Approximation; Face Verification

1. INTRODUCTION

Metric learning has shown promising results with learning the proper Mahalanobis distance for many data mining tasks.The goal of metric learning is to learn an optimal linear or nonlinear projection for original high-dimension features from supervised or weakly supervised constraints, and there have been a lot of works in this field [28, 1, 26, 9, 27, 24]. Metric learning has been widely used in applications where metric between samples plays an important role, such as image classification, face verification and recognition in computer vision [6, 17, 13, 14], learning to rank in information retrieval [16, 15], bioinformatics [25], *etc.*

In image mining and retrieval, there are many metric learning algorithms learning an optimal Mahalanobis distance from weakly supervised constraints between images. The main constraint paradigms include: pair constraint [1], triplet constraint [27], and quadruplet constraint [13]. To avoid overfitting and learn correlation among samples, many regularizations were proposed to impose on the projection matrix. Among these regularizations, the low-rank regularization is proved to be effective and efficient to learn potential correlations from training data, *e.g.* trace norm and Fantope regularization[14].

In this paper, we propose a novel metric learning model using the capped trace norm as the low-rank regularization for Mahalanobis

[*] To whom all correspondence should be addressed. This work was partially supported by US NSF-IIS 1117965, NSF-IIS 1302675, NSF-IIS 1344152, NSF-DBI 1356628, NIH R01 AG049371.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

KDD '16, August 13–17, 2016, San Francisco, CA, USA.
© 2016 ACM. ISBN 978-1-4503-4232-2/16/08. . . $15.00
DOI: http://dx.doi.org/10.1145/2939672.2939853

distance metric learning. Different from trace norm which minimizes sum of all singular values, or Fantope regularization which minimizes sum of k smallest singular values, the capped trace norm penalizes the singular values that are less than a threshold adaptively learned in the optimization. As a result, the non-relevant information (associated to smallest singular values) can be filtered out, such that the metric learning model is more robust. Meanwhile, we only need input an approximate rank value, thus our regularization term is tighter than trace norm and more stable and applicable in practical problems. We also derive an efficient optimization algorithm with rigorous convergence analysis. In the experiments, we impose our novel low-rank regularization on different metric learning formulations and compare with other the state-of-the-art metric learning methods. Experimental results show that our method outperforms other related methods on benchmark datasets.

2. RELATED WORK

The goal of metric learning is to learn an adaptive distance, such as Mahalanobis distance $d_M(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^T M (\mathbf{x}_i - \mathbf{x}_j)}$, for the problem of interest using the information brought by training examples. Most of metric learning methods use weakly-supervised constraints. There are three mainly paradigms of constraints, such as pairwise, triplet, or quadruplet constraint.

The pairwise constraint contains information whether two objects in a pair are similar or dissimilar, sometimes positive pairs or negative pairs. Pairwise constraint is represented by $\mathcal{D}$ and $\mathcal{S}$ as:

$$\begin{aligned} \mathcal{S} &= \{(\mathbf{x}_i, \mathbf{x}_j) : \mathbf{x}_i \text{and } \mathbf{x}_j \text{are similar}\} \\ \mathcal{D} &= \{(\mathbf{x}_i, \mathbf{x}_j) : \mathbf{x}_i \text{and } \mathbf{x}_j \text{are dissimilar}\} . \qquad (1) \end{aligned}$$

The information-Theoretic Metric Learning (ITML) is one of many methods using pairwise constraint training examples in metric learning field [28, 1, 11, 5], and it is formulated as follows:

$$\begin{aligned} \min_{M \in \mathbb{S}^d_+} \quad & \gamma \sum_{i,j} \xi_{ij} + D_{ld}(M, M_0) \\ s.t. \quad & d_M^2(\mathbf{x}_i, \mathbf{x}_j) \leq u + \xi_{ij}, \forall (\mathbf{x}_i, \mathbf{x}_j) \in \mathcal{S} \\ & d_M^2(\mathbf{x}_i, \mathbf{x}_j) \geq l - \xi_{ij}, \forall (\mathbf{x}_i, \mathbf{x}_j) \in \mathcal{D} , \qquad (2) \end{aligned}$$

where u and l are upper bound and lower bound for similar samples and dissimilar samples respectively. ξ_{ij} is a safety margin distance for each pair and $D_{ld}(M, M_0)$ is LogDet divergence and $D_{ld}(M, M_0) = \text{Tr}(M M_0^{-1}) - \text{logdet}(M M_0^{-1}) - d$ where d is the dimension of input space and M_0 is a positive definite matrix.

Triplet constraint is also widely used in metric learning, and it is denoted by $\mathcal{R}$ as:

$$\mathcal{R} = \{(\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k) : \mathbf{x}_i \text{ is more similar to } \mathbf{x}_j \text{ than to } \mathbf{x}_k\} . \qquad (3)$$

Large Margin Nearest Neighbor (LMNN) [27] is one of the most widely used metric learning methods which uses triplet constraint on training examples. The LMNN model is to solve:

$$\begin{aligned} \min_{M \in \mathbb{S}^d_+} \quad & (1-\mu) \sum_{(\mathbf{x}_i, \mathbf{x}_j) \in S} d_M^2(\mathbf{x}_i, \mathbf{x}_j) + \mu \sum_{(i,j,k) \in \mathcal{R}} \xi_{ijk} \qquad (4) \\ s.t. \quad & d_M^2(\mathbf{x}_i, \mathbf{x}_k) - d_M^2(\mathbf{x}_i, \mathbf{x}_j) \geq 1 - \xi_{ijk} \forall (\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k) \in \mathcal{R} , \end{aligned}$$

where $\mu \in [0, 1]$ controls relative weight between two terms, and

$$\begin{aligned} &\mathcal{S} = \{(\mathbf{x}_i, \mathbf{x}_j) : y_i = y_j \text{ and } \mathbf{x}_j \text{ belongs to the } k\text{-neighborhood of } \mathbf{x}_i\} \\ &\mathcal{R} = \{(\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k) : (\mathbf{x}_i, \mathbf{x}_j) \in \mathcal{S}, y_i \neq y_k\} . \qquad (5) \end{aligned}$$

It is proved to be very effective to learn Mahalanobis distance in practice, and is extended to many methods for different applications [20, 10, 9]. However, LMNN is prone to be overfitting sometimes, and it is also sensitive to Euclidean distance when it computes neighbors of each sample at the beginning.

In [13], a novel quadruplet constraint was proposed to model similarity from complex semantic label relations, for example, the degree of presence of smile, from least smiling to most smiling. The scheme of quadruplet constraint is as follows:

$$\mathcal{A} = \{(\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k, \mathbf{x}_l) : d_M^2(\mathbf{x}_k, \mathbf{x}_l) \geq d_M^2(\mathbf{x}_i, \mathbf{x}_j) + \delta\} , \qquad (6)$$

where δ is a soft margin. Quadruplet is able to encompass pair and triplet constraint. Pairwise constraint can be represented as $(\mathbf{x}_i, \mathbf{x}_i, \mathbf{x}_i, \mathbf{x}_j)$, and set $\delta = l$, so $d_M^2(\mathbf{x}_i, \mathbf{x}_j) \geq l$ and $\mathbf{x}_i, \mathbf{x}_j$ are from dissimilar set; or $(\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_i, \mathbf{x}_i)$, set $\delta = -u$, then $d_M^2(\mathbf{x}_i, \mathbf{x}_j) \leq u$, $(\mathbf{x}_i, \mathbf{x}_j)$ are from similar set. Similarly, triplet constraint can also be represented as $(\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_i, \mathbf{x}_k)$.

To solve the problem of overfitting, many regularizations over matrix M were proposed in the past. In [22], they impose squared Frobenius norm on M, and form an SVM like structure to do metric learning:

$$\begin{aligned} \min_{W} \quad & ||M||_F^2 + C \sum_{i,j,k} \xi_{ijk} \qquad (7) \\ s.t. \quad & d_M^2(\mathbf{x}_i, \mathbf{x}_k) - d_M^2(\mathbf{x}_i, \mathbf{x}_j) \geq 1 - \xi_{ijk}, \forall (\mathbf{x}_i, \mathbf{x}_j, \mathbf{x}_k) \in \mathcal{R} , \end{aligned}$$

where $M = A^T W A$, matrix A is fixed and known, and the diagonal matrix W is learned.

There are some works imposing low-rank structure on M. The most direct way is let $M = L^T L$, where $M \in \mathbb{R}^{d \times d}$, $L \in \mathbb{R}^{k \times d}$ and k is smaller than d. So, M is a matrix of rank k.

In [16], they proposed a robust structure metric learning method, and used nuclear norm as convex approximation of low-rank regularization, and it can be expressed as a convex optimization problem:

$$\begin{aligned} \min_{M \in \mathbb{S}^d_+} \quad & \text{Tr}(M) + \frac{C}{n} \sum_{q \in \mathcal{X}} \xi_q \\ s.t. \quad & \forall q \in \mathcal{X}, y \in \mathcal{Y} : \\ & \langle M, \phi(q, y_q) - \phi(q, y) \rangle_F \geq \Delta(y_q, y) - \xi_q , \qquad (8) \end{aligned}$$

where $\mathcal{X} \subset \mathbb{R}^d$ is the training set of n points, $\mathcal{Y}$ is the set of all permutations over $\mathcal{X}$, $C > 0$ is a slack trade-off parameter, ϕ is a feature encoding of an input-output pair, and $\Delta(y_q, y)$ is the designed margin.

In [14], a tighter rank minimization approximation, Fantope regularization, was proposed and imposed on M, and holds an explicit control over rank of M. The formulation is $\text{Reg}(M) = \sum_{i=1}^{k} \sigma_i(M)$, where $\sigma_i(M)$ are k smallest singular values.

3. METRIC LEARNING USING CAPPED TRACE NORM

In this paper, we are going to introduce a novel low-rank regularization based metric learning method, so that we can avoid the problem of overfitting and learn an effective structure from limited training data. As we mentioned in last section, there have been already many different types of regularization over $M \in \mathcal{S}^d_+$ in metric learning literature. The Frobenius norm regularization proposed by [22] can avoid overfitting, however, the definition of M in this paper restricts the generation of M and it cannot learn correlation between features.

In weakly supervised metric learning, the algorithm does not have access to the labels of training data, and it is only provided with side information which is in the form of pair or triplet constraints. In this case, the low-rank regularization seems to be an effective way to learn correlations between data. Trace norm (also called as nuclear norm) has been used as the convex relaxation of the rank minimization, however, there still is a gap between trace norm and rank minimization. Because trace norm is the sum of all singular values, if one of the large singular values changes, the trace norm will also change correspondingly, but the rank of the original matrix keeps constant.

Setting $M = L^T L$ or imposing Fantope regularization are both explicit way to control the rank of matrix M. The performance could be good if we can find a good fitted rank. However, in practice, we do not know the rank of a matrix accurately, and we have to tune this parameter very carefully, because a small deviation of parameter k from optimal value may have large influence on the final performance. It is a tedious process to select the best k from a large range.

In this paper, we will use the capped trace norm as low-rank regularization [29, 30]. It can be represented as $\text{Reg}(M) = \sum_i \min\{\sigma_i(M), \varepsilon\}$, where ε is a threshold value. In this regularization, we only minimize the singular values that are smaller than ε, and we ignore other large singular values. Thus, when large singular values vary, our regularization behaves the same as low-rank regularization, and keeps constant too. In practical problems, it is difficult to estimate the rank of matrix M, but the ε value in capped trace norm can be easily decided [4, 8].

Because quadruplet constraint can encompass pair and triplet constraints, in this paper, we use quadruplet constraint to form a new robust metric learning model as:

$$\min_{M \in \mathbb{S}_+^d} \quad \sum_{q \in \mathcal{A}} \left[\xi_q + \left\langle M, \boldsymbol{x}_{ij}\boldsymbol{x}_{ij}^T - \boldsymbol{x}_{kl}\boldsymbol{x}_{kl}^T \right\rangle \right]_+ + \frac{\gamma}{2} \sum_i \min\{\sigma_i(M), \varepsilon\} \,. \tag{9}$$

We will also use this model in the optimization and convergence analysis sections, but the conclusions are the same when we use pair or triplet constraint.

3.1 Optimization Algorithm

Objective function (9) is non-smooth and non-convex, and it is hard to optimize. In this section, at first, we will use re-weighted method to transform the original objective function to a convex subproblem, then proximal gradient method is applied to solve this new subproblem. In next section, we will prove that our objective function will converge, and the values of original objective function (9) are non-increasing after each step, and a local optimum value is to be obtained.

According to the re-weighted algorithm described in [19, 18], let $M = U\Sigma V^T$ and singular value σ_i are in ascending order. We define:

$$D = \frac{1}{2} \sum_{i=1}^{k} \sigma_i^{-1} \boldsymbol{u}_i \boldsymbol{u}_i^T \,. \tag{10}$$

Therefore, original problem (9) can be transformed to:

$$\min_{M \in \mathbb{S}_+^d} \quad \sum_{q \in \mathcal{A}} \left[\xi_q + \left\langle M, \boldsymbol{x}_{ij}\boldsymbol{x}_{ij}^T - \boldsymbol{x}_{kl}\boldsymbol{x}_{kl}^T \right\rangle \right]_+ + \frac{\gamma}{2} \operatorname{Tr}(M^T D M) \,. \tag{11}$$

When we fix D, this problem is convex, and we use proximal gradient method to optimize it iteratively. In each iteration, M is updated by performing a subgradient descent, and the subgradient descent of problem (11) with respect to M is:

$$\nabla_M = \sum_{q \in \mathcal{A}^+} (\boldsymbol{x}_{ij}\boldsymbol{x}_{ij}^T - \boldsymbol{x}_{kl}\boldsymbol{x}_{kl}^T) + \gamma D M \tag{12}$$

where $\mathcal{A}^+$ denotes the subset of constraints in $\mathcal{A}$ that is larger than 0 in function (11). After each step, M is projected onto the positive semidefinite cone.

$$M = \Pi_{\mathcal{S}_+^d} (M - \eta \nabla_M) \,. \tag{13}$$

Optimization algorithm of our method is summarized in Algorithm (1) below.

Algorithm 1 Algorithm to solve problem (9).

```
Input: $\mathcal{A}$, $X \in \mathcal{R}^{d \times n}$
Output: $M \in \mathcal{S}_+^d$
while not converge do
    Update $D \Leftarrow \frac{1}{2}\sum_{i=1}^{k} \sigma_i^{-1}\boldsymbol{u}_i\boldsymbol{u}_i^T$ .
    Update $M \Leftarrow \Pi_{\mathcal{S}_+^d}(M - \eta\nabla_M)$
        $\nabla_M = \sum_{q \in \mathcal{A}^+}(\boldsymbol{x}_{ij}\boldsymbol{x}_{ij}^T - \boldsymbol{x}_{kl}\boldsymbol{x}_{kl}^T) + \gamma D M$
end while
```

3.2 Convergence Analysis

Using the algorithm above, we can solve our original non-smooth and non-convex objective function (9). In this section, we prove the convergence of our optimization algorithm, and a local solution can be obtained in the end. In capped trace norm optimization algorithm, the number of thresholded singular values varies for different iterations, thus the convergence proof is difficult.

THEOREM 1. *Through Algorithm 1, the objective function (9) will converge, or the values of objective function (9) are non-increasing monotonically.*

In order to prove Theorem 1, at first, we need the following Lemmas.

LEMMA 1. *According to [23], any two hermitian matrices $A, B \in R^{d \times d}$ satisfy the inequality ($\sigma_i(A)$, $\sigma_i(B)$ are singular values sorted in the same order)*

$$\sum_{i=1}^{d} \sigma_i(A)\, \sigma_{d-i+1}(B) \le \operatorname{Tr}\left(A^T B\right) \le \sum_{i=1}^{d} \sigma_i(A)\, \sigma_i(B) \,. \tag{14}$$

LEMMA 2. *Let $M = U\Sigma V^T$, σ_i are singular values of M in ascending order, and there are k singular values less than ε. $\hat{M} = \hat{U}\hat{\Sigma}\hat{V}^T$, $\hat{\sigma}_i$ are singular values of $\hat{M}$ in ascending order, and there are $\hat{k}$ singular values less than ε. $\hat{M}$ denotes the updated parameter after M. ε is a constant value. So it is true that,*

$$\begin{aligned} &\sum_i \min\{\hat{\sigma}_i(\hat{M}), \varepsilon\} - \operatorname{Tr}\left(\hat{M}^T D \hat{M}\right) \\ &\le \sum_i \min\{\sigma_i(M), \varepsilon\} - \operatorname{Tr}\left(M^T D M\right) \,, \end{aligned} \tag{15}$$

where D is defined as (10).

Proof: It's obvious that

$$\sigma_i - 2\hat{\sigma}_i + \sigma_i^{-1}\hat{\sigma}_i^{\,2} = \frac{1}{\sigma_i}(\sigma_i - \hat{\sigma}_i)^2 \ge 0 \,, \tag{16}$$

hence the following inequality holds:

$$\sum_{i=1}^{k}\left(\hat{\sigma}_i - \frac{1}{2}\sigma_i^{-1}\hat{\sigma}_i^2\right) \leq \frac{1}{2}\sum_{i=1}^{k}\sigma_i\,. \tag{17}$$

We know there are $\hat{k}$ singular values of $\hat{M}$ less than ε and they are in ascending order, the first $\hat{k}$ smallest singular values $\hat{\sigma}_i$ are less than ε, thus no matter whether $\hat{k} \geq k$ or $\hat{k} < k$, it holds that:

$$\sum_{i=1}^{\hat{k}}\hat{\sigma}_i - \hat{k}\varepsilon \leq \sum_{i=1}^{k}\hat{\sigma}_i - k\varepsilon\,. \tag{18}$$

Combining two inequalities (17) and (18), we get:

$$\sum_{i=1}^{\hat{k}}\hat{\sigma}_i - \frac{1}{2}\sum_{i=1}^{k}\sigma_i^{-1}\hat{\sigma}_i^2 - \hat{k}\varepsilon \leq \frac{1}{2}\sum_{i=1}^{k}\sigma_i - k\varepsilon\,. \tag{19}$$

Then, summing $d\varepsilon$ on both sides of inequality (19), where d is the dimension of matrix M, we are able to get the following inequality:

$$\begin{aligned} &\textstyle\sum_{i=1}^{\hat{k}}\hat{\sigma}_i + \left(d-\hat{k}\right)\varepsilon - \frac{1}{2}\sum_{i=1}^{k}\sigma_i^{-1}\hat{\sigma}_i^2 \\ &\quad \leq \textstyle\sum_{i=1}^{k}\sigma_i + (d-k)\,\varepsilon - \frac{1}{2}\sum_{i=1}^{k}\sigma_i\,. \end{aligned} \tag{20}$$

As per the definition of matrix $D = \frac{1}{2}\sum_{i=1}^{k}\sigma_i^{-1}\boldsymbol{u}_i\boldsymbol{u}_i^T$, we know that:

$$\begin{aligned} \operatorname{Tr}\left(M^TDM\right) &= \frac{1}{2}\operatorname{Tr}\left(\sum_{i=1}^{k}\sigma_i^{-1}\boldsymbol{u}_i\boldsymbol{u}_i^TMM^T\right) \\ &= \frac{1}{2}\operatorname{Tr}\left(U\Lambda U^TU\Sigma^2U^T\right) \\ &= \frac{1}{2}\operatorname{Tr}\left(U\Lambda\Sigma^2U^T\right) \\ &= \frac{1}{2}\sum_{i=1}^{k}\sigma_i\,. \end{aligned} \tag{21}$$

Via Lemma 1, we know that:

$$\begin{aligned} \frac{1}{2}\operatorname{Tr}\left(\sum_{i=1}^{k}\sigma_i^{-1}\boldsymbol{u}_i\boldsymbol{u}_i^T\hat{M}\hat{M}^T\right) &= \frac{1}{2}\operatorname{Tr}\left(U\Lambda U^T\hat{U}\hat{\Sigma}^2\hat{U}^T\right) \\ &\geq \frac{1}{2}\sum_{i=1}^{k}\sigma_i^{-1}\hat{\sigma}_i^2\,. \end{aligned} \tag{22}$$

Combining Eq. (21), inequalities (20) and (22), we have:

$$\begin{aligned} &\sum_{i=1}^{\hat{k}}\hat{\sigma}_i + \left(d-\hat{k}\right)\varepsilon - \frac{1}{2}\operatorname{Tr}\left(\sum_{i=1}^{k}\sigma_i^{-1}\boldsymbol{u}_i\boldsymbol{u}_i^T\hat{M}\hat{M}^T\right) \\ &\leq \sum_{i=1}^{k}\sigma_i + (d-k)\,\varepsilon - \frac{1}{2}\operatorname{Tr}\left(\sum_{i=1}^{k}\sigma_i^{-1}u_iu_i^TMM^T\right)\,. \end{aligned} \tag{23}$$

Finally, inequality holds that:

$$\begin{aligned} &\sum_i \min\{\hat{\sigma}_i(\hat{M}),\varepsilon\} - \operatorname{Tr}\left(\hat{M}^TD\hat{M}\right) \\ &\leq \sum_i \min\{\sigma_i(M),\varepsilon\} - \operatorname{Tr}\left(M^TDM\right)\,. \end{aligned} \tag{24}$$

LEMMA 3. *Function (11), is convex with domain $\mathcal{S}_+^d$, and gradient of this function is clearly Lipschitz continuous with a large enough constant L. Secondly, positive semidefinite cone is a closed convex cone. As per [2], when we use proximal gradient method and select a proper learning rate, the value of this function converges in each iteration.*

Right now, we are able to prove Theorem 1 by using the Lemmas above.

Proof: Via Lemma 3, after we use proximal gradient descent method to minimize function (11) in Algorithm 1, it is guaranteed that:

$$\begin{aligned} &\sum_{q\in\mathcal{A}}\left[\xi_q + \left\langle\hat{M},\boldsymbol{x}_{ij}\boldsymbol{x}_{ij}^T - \boldsymbol{x}_{kl}\boldsymbol{x}_{kl}^T\right\rangle\right]_+ + \frac{\gamma}{2}\operatorname{Tr}\left(\hat{M}^TD\hat{M}\right) \\ &\leq \sum_{q\in\mathcal{A}}\left[\xi_q + \left\langle M,\boldsymbol{x}_{ij}\boldsymbol{x}_{ij}^T - \boldsymbol{x}_{kl}\boldsymbol{x}_{kl}^T\right\rangle\right]_+ + \frac{\gamma}{2}\operatorname{Tr}\left(M^TDM\right)\,. \end{aligned} \tag{25}$$

Via Lemma 2, we can easily know that:

$$\begin{aligned} &\gamma\sum_i \min\{\hat{\sigma}_i(\hat{M}),\varepsilon\} - \operatorname{Tr}\left(\hat{M}^TD\hat{M}\right) \\ &\leq \gamma\sum_i \min\{\sigma_i(M),\varepsilon\} - \operatorname{Tr}\left(M^TDM\right)\,. \end{aligned} \tag{26}$$

Finally, we combine inequalities (25) and (26) to achieve:

$$\begin{aligned} &\sum_{q\in\mathcal{A}}\left[\xi_q + \left\langle\hat{M},\boldsymbol{x}_{ij}\boldsymbol{x}_{ij}^T - \boldsymbol{x}_{kl}\boldsymbol{x}_{kl}^T\right\rangle\right]_+ \\ &\quad +\frac{\gamma}{2}\sum_i \min\{\hat{\sigma}_i(\hat{M}),\varepsilon\} \\ \leq\;&\sum_{q\in\mathcal{A}}\left[\xi_q + \left\langle M,\boldsymbol{x}_{ij}\boldsymbol{x}_{ij}^T - \boldsymbol{x}_{kl}\boldsymbol{x}_{kl}^T\right\rangle\right]_+ \\ &\quad +\frac{\gamma}{2}\sum_i \min\{\sigma_i(M),\varepsilon\}\,. \end{aligned} \tag{27}$$

So far, it is clear that the value of our proposed objective function will not increase by using our optimization algorithm, thus we prove the Theorem 1 that our optimization algorithm is non-increasing monotonically. Because M is a positive semidefinite matrix, we know that the objective function (9) is at least larger than zero. So our objective function is also lower bounded. Therefore we can conclude that our optimization algorithm converges, and a local optimum value is to be obtained in the end.

4. EXPERIMENTAL RESULTS

We evaluate our proposed model on different datasets, including synthetic dataset, widely used face recognition datasets, and some other datasets in image data mining. There are two main goals in our experiment: first, we will show that our model is able to outperform the state-of-the-art metric learning methods; second, our proposed capped trace norm is more stable to be applied to solve practical problems than Fantope regularization.

4.1 Synthetic Data

In this experiment, we evaluate our proposed metric learning on a synthetic dataset, and each constraint is quadruplet.

Dataset: We follow the setting of experiment in [14] to generate a synthetic dataset. We define a target symmetric positive semidefinite matrix $T \in \mathbb{S}_+^d$, and $T = \begin{pmatrix} A & 0 \\ 0 & 0 \end{pmatrix}$, where $A \in \mathbb{S}_+^e$ is a random symmetric positive definite matrix with $\operatorname{rank}(A) = e$ and $e < d$. Matrix A is a multiplication of one random symmetric matrix and its transpose. So, $\mathrm{rank}(T) = \mathrm{rank}(A) = e$. $X \in \mathbb{R}^{d\times n}$ is a feature matrix, each element is generated from gaussian distribution in $[0, 1]$, and each sample is a feature vector $\boldsymbol{x}_i \in \mathbb{R}^d$. The Mahalanobis distance between two feature vectors $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ is given by: $d_T^2(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\boldsymbol{x}_i - \boldsymbol{x}_j)^T T(\boldsymbol{x}_i - \boldsymbol{x}_j)$.

To build a training constraint set $\mathcal{A}$, we randomly sample pairs of distance using quadruplets and get the ground truth using d_T^2, so that: $\forall(\boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{x}_k, \boldsymbol{x}_l) \in \mathcal{A}, d_T^2(\boldsymbol{x}_i, \boldsymbol{x}_j) < d_T^2(\boldsymbol{x}_k, \boldsymbol{x}_l)$, and it denotes that distance between sample pair (i, j) is smaller than sample pair (k, l). Training set $\mathcal{A}$ is used as training data to learn Mahalanobis metric matrix M. Validation set $\mathcal{V}$ and test set $\mathcal{T}$ are generated in the same way as $\mathcal{A}$, and they are used to tune parameters and test evaluation respectively.

Setting: In the experiment, we set $e = 10$, $d = 100$, $n = 10^6$, $|\mathcal{A}| = |\mathcal{V}| = |\mathcal{T}| = 10^4$. After we learn a metric M, we evaluate these metrics on test set $\mathcal{T}$, and measure accuracy of satisfying the constraints. In this experiment, we compare with four other methods: metric learning with no regularization, metric learning with trace norm regularization and metric learning with fantope regularization. Parameter γ are tuned in the range of $\{10^{-2}, 10^{-1}, 1, 10, 10^2\}$, and rank of Mahalanobis metric M are tuned from $[5, 20]$.

Compared Methods: Because we use quadruplet constraints in this experiment, the general model we use to solve this problem is:

$$\sum_{q\in\mathcal{A}} \left[\xi_q + \left\langle M, \boldsymbol{x}_{ij}\boldsymbol{x}_{ij}^T - \boldsymbol{x}_{kl}\boldsymbol{x}_{kl}^T \right\rangle\right]_+ + \frac{\gamma}{2}\mathrm{Reg}(M)\,. \qquad (28)$$

- Metric learning with no regularization. We set $\gamma = 0$ in problem (28).

- Metric learning with trace norm regularization. In this model, $\gamma > 0$ and $\mathrm{Reg}(M) = \sum_i \sigma_i(M)$, where $\sigma_i(M)$ are singular values of M.

- Metric learning with Fantope regularization. In this model, $\gamma > 0$ and $\mathrm{Reg}(M) = \sum_{i=1}^{k} \sigma_i(M)$, where $\sigma_i(M)$ are k smallest singular values of M.

- Metric learning with capped trace norm regularization. In our model, $\gamma > 0$ and $\mathrm{Reg}(M) = \sum_i \min\{\sigma_i(M), \varepsilon\}$, where $\sigma_i(M)$ are singular values of M.

Evaluation Metrics: After learning a metric matrix M from training constraints $\mathcal{A}$, we evaluate it by computing the number of dominant singular values, namely $\mathrm{rank}(M)$. Then we test it on testing constraints $\mathcal{T}$ and compute the accuracy of satisfied constraints.

Results: Table 1 shows the results of our experiment. As we can see, metric learning with Fantope regularization and capped norm regularization performs much better than other two methods. Our method can get a comparable results to metric learning with Fantope regularization, and this is the result when we tune parameter k for Fantope regularization very carefully.

Figure 2 represents the accuracies of metric learning with Fantope regularization and our method when selection of rank changes. It is obvious that our method always outperforms Fantope regularization when we select parameter rank randomly except 10. Our method performs more stable when we do not have enough time to tune the rank of Mahalanobis metric M. In practice, it common that we do not know the exact rank of a matrix, our method is more applicable to solve practical problems.

Method	Accuracy	rank(M)
ML	85.62%	53
ML + Trace	88.44%	41
ML + Fantope	**95.50%**	10
ML + capped	**95.43%**	10

Table 1: Synthetic experiment results.

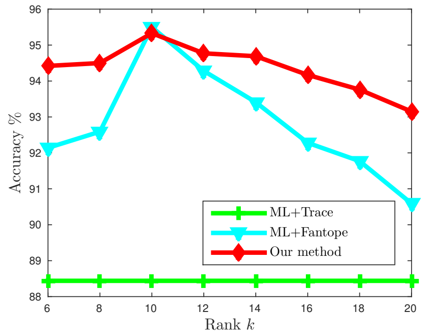


Figure 2: Accuracy vs the number of rank k

4.2 Face Verification

In this section, we evaluate our method, pairwise constraints with capped norm regularization, on two challenging face recognition datasets: Labeled Faces in the Wild (LFW) [7] and Public Figures Face Database (PubFig) [12]. In our experiments, we focus on face verification task, namely deciding if two face images are from the same person, and results show that our method can outperform the state-of-the-art metric learning algorithm.

4.2.1 Labeled Faces in The Wild

Dataset: The Labeled Faces in the Wild dataset is considered as the current state-of-the-art face recognition benchmark. It contains 13,233 unconstrained face images of 5749 individuals, and 1680 of these pictured people appear in two or more distinct photos in this data set.

There are two different feature representations in this experiment, LFW SIFT feature dataset and LFW Attribute feature dataset. We use the face representation proposed by [5], it extracts SIFT descriptors [6] at 9 automatically detected facial landmarks over three scales. Each image is a feature vector of size 3,456. To make this dataset tractable for distance metric learning algorithm and save time, we perform principle component analysis to reduce the dimension, and we select 100 largest principle components in this experiment [11]. We also use ‘high-level describable visual attributes (gender, race, age, hair color) in [12]. These features of face image are insensitive to pose, illumination, expression and other imaging conditions, and can avoid some obvious mistakes, for example men are confused for women or child for middle-aged. Each image is represented by a vector $\mathbf{x} \in \mathbb{R}^d$ where d is the number of attributes to describe the image. Each entry in vector $\mathbf{x}$ means the score of presence of each specific attribute.

Setting: To compare with other state-of-the-art methods on face verification, we follow the experiment setting in [11]. Data are organized in 10 folds, and each fold consists of 300 similar constraints (two faces in a pair are from the same person) and 300 dissimilar constraints (two faces in a pair are from different person). The average over results of 10 folds is used as final evaluation metric. In the experiment, we an only access pairwise constraints given by similar or dissimilar pairs, and labels or more training data are not allowed. In the experiment, we tune parameter μ from range $[10^{-2}, 10^{-1}, 1, 10, 10^{2}]$, and parameter rank k of matrix M from $\{30, 35, 40, 45, 50, 55, 60, 65, 70\}$ in LFW SIFT dataset and from $\{10, 15, 20, 25, 30, 35, 40, 45, 50\}$ in LFW Attribute dataset. For other methods, we follow their already tuned parameter setting in [11].

To prove the stableness of our method, we split LFW SIFT feature dataset and LFW attribute feature dataset into 5 folds, and evaluate our metric learning with capped trace norm model and metric learning with Fantope norm on these datasets. We run experiments 5 times using different training/testing splits and compute mean value and standard variance for validation.

Compared Methods:

- IDENTITY: We compute Euclidean distance directly as a baseline.
- MAHALANOBIS: Traditional Mahalanobis distance between images in a pair is computed, where the metric matrix is inverse of covariance between two vectors.
- KISSME: A metric learning methods based on a statistical inference perspective. It learns a distance metric from equivalence constraints and can be used in large scale dataset [11].
- ITML: Metric learning methods proposedd in [1]. They use LogDet divergence as regularization so that they do not need do explicit positive semi-definite projection.
- LDML: [5] offers a metric learning method that uses logistic discriminant to learn a metric from a set of labeled image pairs.

  Because of the setting of this experiment, for method Fantope and Cap, the general function for their models on this task is:

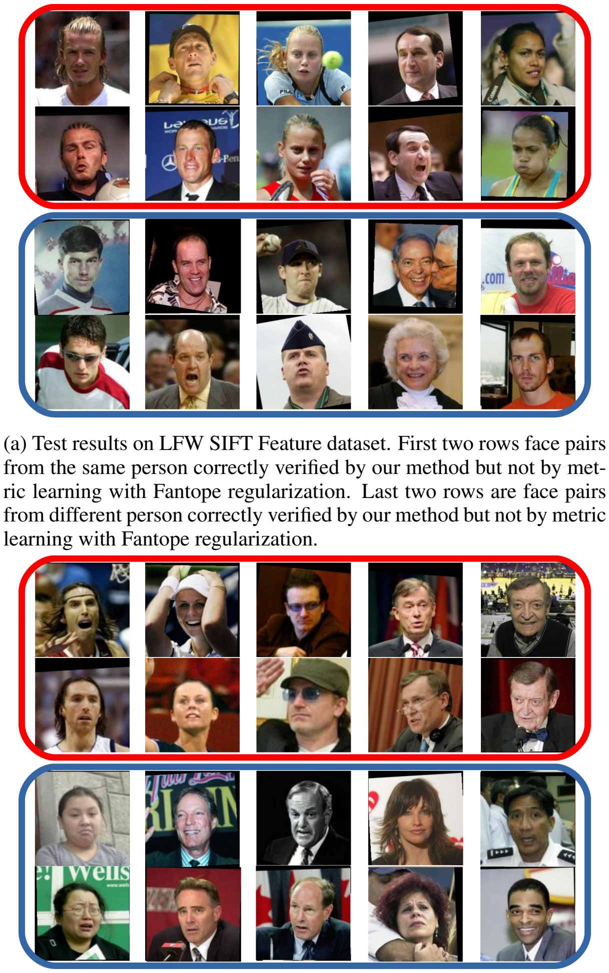
$$\sum_{(i,j)\in\mathcal{S}} \left[\langle M, \mathbf{x}_{ij}\mathbf{x}_{ij}^T\rangle - u\right]_+ + \sum_{(i,j)\in\mathcal{D}} \left[l - \langle M, \mathbf{x}_{ij}\mathbf{x}_{ij}^T\rangle\right]_+ + \frac{\gamma}{2}\mathrm{Reg}(M)\,, \qquad (29)$$

  where $\mathcal{D}$ and $\mathcal{S}$ denote dissimilar pair set and similar pair set respectively. u is the upper bound for Mahalanobis distance between two samples in similar pair set, and l represents the distance lower bound between two samples in dissimilar pair set.

- ML+Fantope: It denotes metric learning with Fantope regularization [14]. As per its definition, $\mathrm{Reg}(M) = \sum_{i=1}^{k} \sigma_i(M)$.
- ML+Cap: Pairwise constraints metric learning with capped trace norm regularization. $\mathrm{Reg}(M) = \sum_i \min\{\sigma_i(M), \varepsilon\}$.

Evaluation Metrics: To measure the face verification accuracy for all these compared methods, we report a Receiver Operator Characteristic (ROC) curve. To compare the performance of each method, we compute Equal Error Rate (EER) of the respective method, and use $1 -$ EER as evaluation criterion, and the method with the lowest EER, or the highest $1 -$ EER is the most accurate one.

Results: We plot ROC curve for each method in Figure 4, and $1 - EER$ values are also computed to evaluate their performance. Figure 4a shows the results on LFW SIFT feature dataset. Mahalanobis distance between two similar pairs performs quite well comparing with Euclidean distance, it increases the performance from 67.5% to 74.8%. KISSME method is the state-of-the-art method on this feature type and reaches an $1 - EER$ at 80.06%, it outperforms widely used metric learning methods ITML and LDML. It is clear that pairwise contraint with low-rank approximation methods, Fantope and CAP, perform better than KISSME, and reach 81.4%, and 81.7% respectively. Our method increases the performance of Euclidean distance by 14.2% and KISSME method by 0.9%. Figure 4b presents the performance of each method on LFW Attribute feature dataset. Our method outperforms KISSME method, and also works better than metric learning than Fantope regularization by 0.4%. 3 shows the result samples on this dataset.

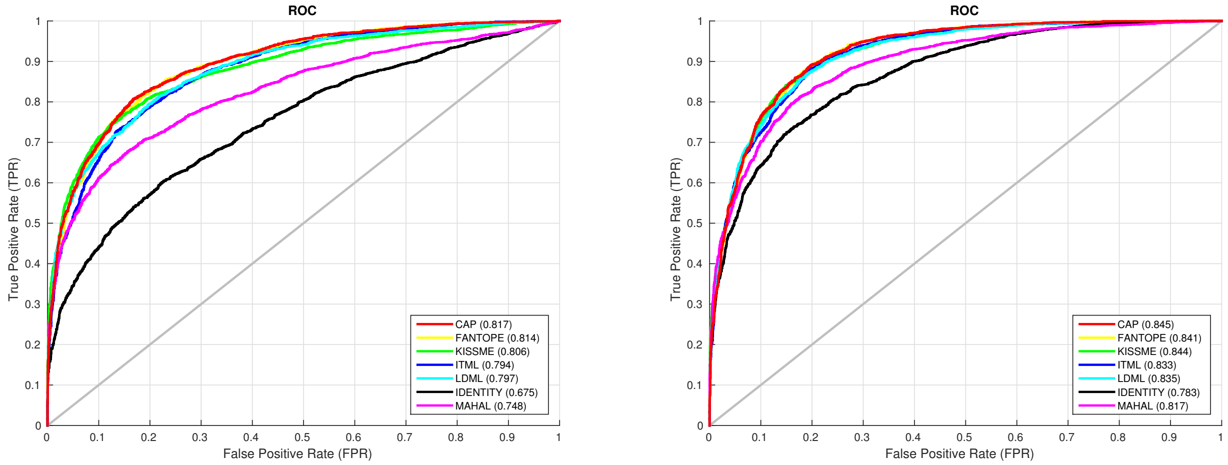


(a) Test results on LFW SIFT Feature dataset. First two rows face pairs from the same person correctly verified by our method but not by metric learning with Fantope regularization. Last two rows are face pairs from different person correctly verified by our method but not by metric learning with Fantope regularization.

(b) Test results on LFW Attribute Feature dataset. First two rows face pairs from the same person correctly verified by our method but not by metric learning with Fantope regularization. Last two rows are face pairs from different person correctly verified by our method but not by metric learning with Fantope regularization.

Figure 3: Sample results on LFW dataset.

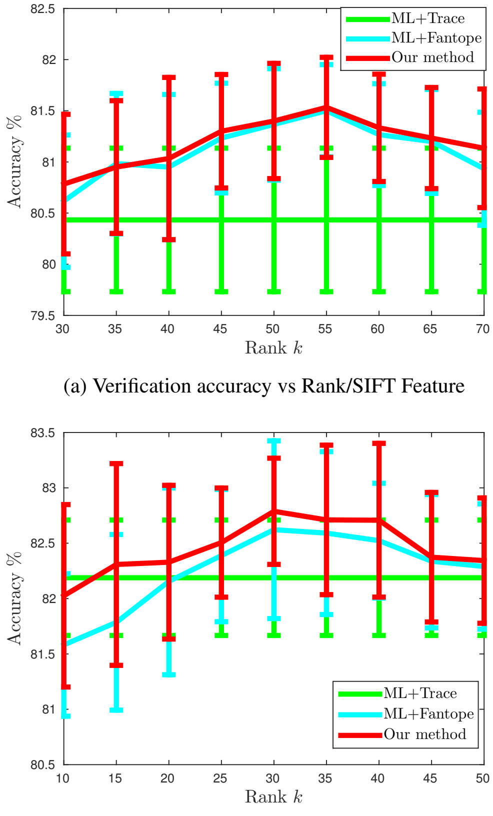
We run 5-fold cross-validation experiments on Fantope method and our method. In Figure 5, we plot accuracy result with respect to rank selection. When we select rank parameter k roughly, it is clearly that our method can get better results than metric learning with Fantope regularization. In large scale dataset, it not piratical to tune parameters as carefully as in small dataset, and sometimes, there is not exact rank at all. So, our method is much more applicable to this kind of situation, and performs better when we inputs an approximation of matrix rank.



(a) ROC curves on SIFT Feature dataset

(b) ROC curves on Attribute Feature dataset

Figure 4: Face verification results on LFW dataset: Figure (4a) ROC curve for different methods on LFW SIFT feature dataset. Figure (4b) ROC curve for different methods on LFW Attribute dataset.



(a) Verification accuracy vs Rank/SIFT Feature

(b) Verification accuracy vs Rank/Attribute Feature

Figure 5: Verification accuracy with respect to rank selection on LFW SIFT feature dataset 5a and LFW attribute feature dataset 5b

4.2.2 *Public Figures Face Database*

Dataset: The PubFig database is a large, real-world face dataset consisting of 58,797 images of 200 people collected from the internet. Images in the data set are downloaded from the internet using search query on a variety of image search engines. Compared to LFW, there are larger number of images per person, and more variation on different poses, lighting conditions, and expressions. Served as a complementary to LFW dataset, PubFig dataset consists "high-level" features of visual face traits that are not sensitive to pose, illumination or other imaging conditions.

Setting: Face verification benchmark dataset consists of 20,000 pairs of images of 140 people. The data is divided into 10 folds with mutually disjoint sets of 14 people each, and each fold contains 1,000 intra and 1,000 extra-personal pairs.

Similar to experiment setting in LFW dataset, to prove stableness of our method, we split Pubfig feature dataset into 5 folds, and evaluate our metric learning with capped trace norm model and metric learning with Fantope norm on these datasets. We run experiments 5 times using different training/testing splits and compute average and standard variance.

Compared Methods: We use the same compared methods in LFW experiment.

Evaluation Metrics: ROC figure is plotted for each method and we also compare verification accuracy when we tune parameter k for methods Fantope and Cap.

Results: Figure 6 represents the performance of each compared method on Pubfig dataset. We calculate Equal Error Rate for them, and it is clear that our method outperforms other correlated metric learning methods. By imposing capped trace norm regularization, our method increases the performance of traditional Mahalanobis distance by about 6%, and the state-of-the-art KISSME method by over 1%.

We also perform another experiments to show the process of parameter tuning procedure. In 7, we tune parameter rank k from 5 to 40, and we can see that our method works more stable than metric learning with Fantope regularization. The performance of Fantope regularization is greatly subject to the selection of rank k. In this figure, we use trace norm regularization as baseline.

4.3 Image Classification

In this section, we evaluate our methods on image classification,, and the task is assigning an image to a predefined class. We can also look as this task as object recognition. In the experiments, we use image dataset with attribute features. Attribute is an important type of semantic properties shared among different objects or activities.

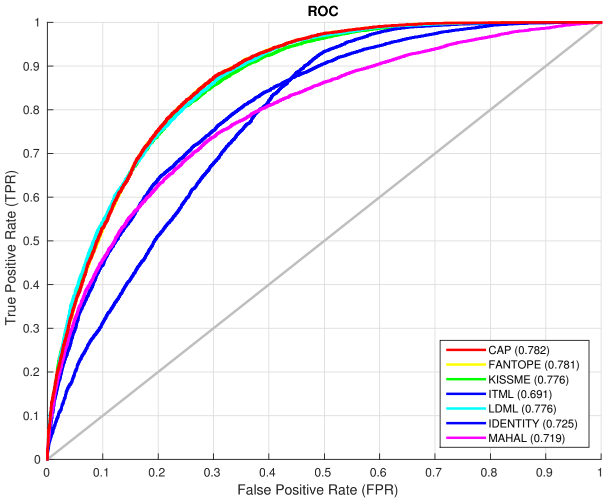


Figure 6: ROC curves of different methods on the Pubfig datasets.

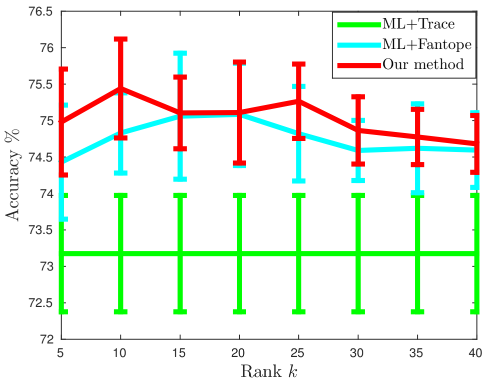


Figure 7: Verification accuracy vs rank k on Pubfig dataset

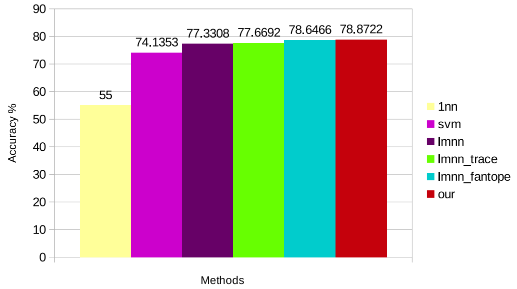


Figure 8: Classification accuracy of each method on Pubfig dataset.

It is a representation in a higher level than the raw feature representation directly extracted from images or videos. In recent years, several attribute datasets are used by various researchers for the study of utilizing attributes for different vision applications. There are two image classification datasets, PubFig dataset and Outdoor Scene Recognition (OSR) dataset.

4.3.1 Public Figures Face Database

Dataset: We use a subset face images of Pubfig dataase, and there are 771 images from 8 face categories.

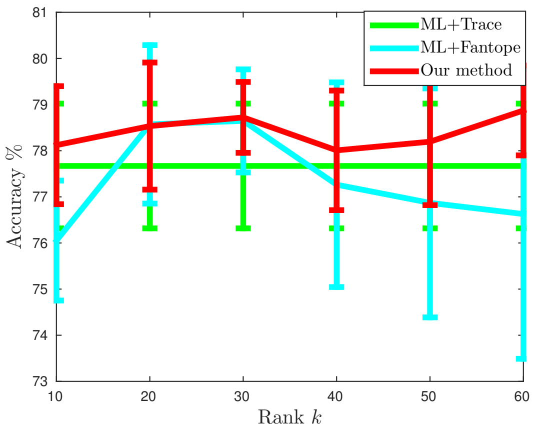
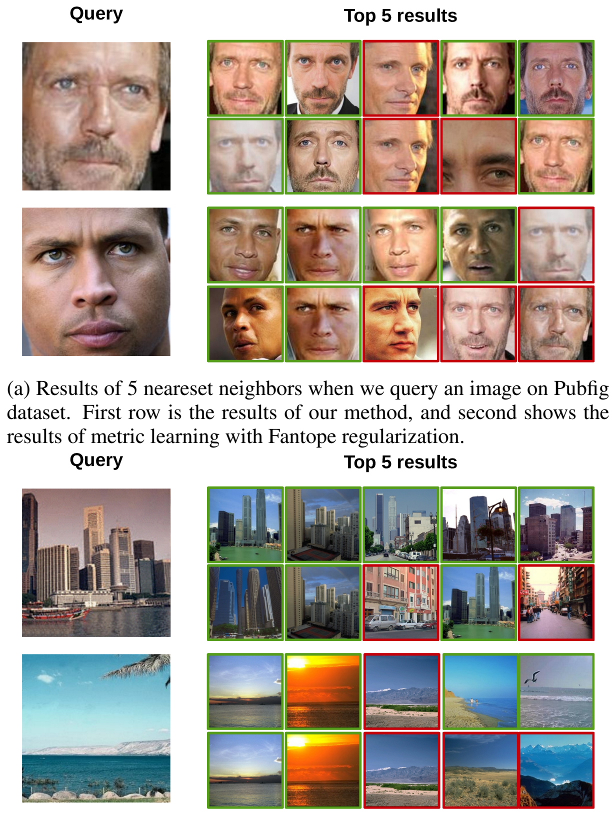


Figure 9: Classification accuracy vs rank k on Pubfig dataset.



(a) Results of 5 neareset neighbors when we query an image on Pubfig dataset. First row is the results of our method, and second shows the results of metric learning with Fantope regularization.

(b) Results of 5 neareset neighbors when we query an image on Pubfig dataset. First row is the results of our method, and second shows the results of metric learning with Fantope regularization.

Figure 10: Results of 5 neareset 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.

Setting: We use the same experiment setup as [14]. Each person contributes 30 images as training data to learn Mahalanobis metric matrix M and build classifier, other images are used as testing data to evaluate classification performance. We run this experiment 5 times, and 30 images per person in training data are selected randomly each time, and average performance is used as evaluation criterion.

Compared Methods:

- KNN: In this method, we use k-nearest neighbor method as classifier and compute Euclidean distance to measure the similarity between any two images. This method works as a baseline.

- SVM: Support vector machine (SVM) is a widely used classifier. In this method, we compare our method with SVM, and the code is from LibLinear [3].

- LMNN: It is one of the most widely-used Mahalanobis distance metric learning methods. In this method, they use labeled information to generate triplet constraints.

  We impose low-rank constraint on matrix M in LMNN method, and the general form is,

$$\begin{aligned}&\tfrac{\gamma}{2}\text{Reg}(M) + (1-\mu)\sum_{(i,j)\in\mathcal{S}} \left\langle M, \mathbf{x}_{ij}\mathbf{x}_{ij}^T\right\rangle \\ &+\mu \sum_{(i,j,k)\in\mathcal{R}} \left[1+\left\langle M, \mathbf{x}_{ij}\mathbf{x}_{ij}^T - \mathbf{x}_{ik}\mathbf{x}_{ik}^T\right\rangle\right]_+ \qquad (30)\end{aligned}$$

- LMNN+Trace norm: We impose trace norm as low-rank regularization approximation and $\text{Reg}(M) = \sum_i \sigma_i(M)$, where $\sigma_i(M)$ is singular value of matrix M.

- LMNN+Fantope: Instead of using trace norm regularization as low-rank approximation, we impose rank of matrix explicitly, and $\text{Reg}(M) = \sum_{i=1}^{k} \sigma_i(M)$, where $\sigma_i(M)$ are k smallest singular values of M.

- LMNN+Capped trace norm: We minimize singular value of matrix M smaller than a value learned in the optimization ε, so $\text{Reg}(M) = \sum_i \min\{\sigma_i(M), \varepsilon\}$.

Evaluation Metrics: In this experiment, we compute classification accuracy for each method as evaluation criterion.

Results: Figure 8 represents the performance of compared methods. 1NN method using Euclidean distance works really bad on this task, and its accuracy is just 55%. SVM method increases the performance of 1NN method greatly by about 20%. When we use LMNN method to learn Mahalanobis distance for this task, and use 1NN classifier, the accuracy reaches 77% and is better than SVM. We impose low-rank regularization, trace norm, Fantope regularization, and capped trace norm respectively, it is clear that our method works best and increases the performance of LMNN method by about 1.5%. Sample query results are presented in Figure 10a, we plot 5 nearest neighbors for each query, and we can find out that our method does a better job than metric learning with Fantope regularization in this task.

In Figure 9, we shows the performance of metric learning with Fantope regularization and capped trace norm method when we tune parameter rank k. The performance of Fantope regularization is very sensitive to the choice of parameter rank k, it is obvious that sometimes, the performance of Fantope regularization is worse than trace norm regularization. It is clear that when we select k from 40 to 100, our method performs much stable than Fantope regularization, because it explicitly controls the rank of matrix M to be k. Our method can learn an adaptive threshold in the optimization and the rank of matrix M is not necessary to be k.

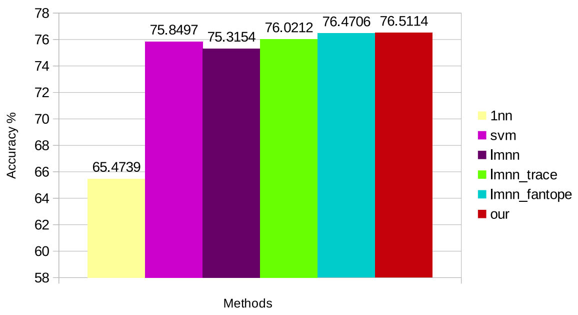


Figure 11: Classification accuracy of each method on OSR dataset.

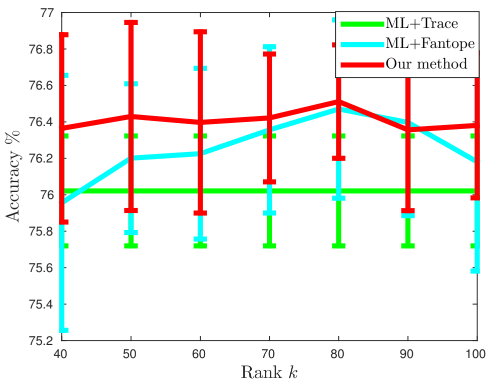


Figure 12: Classification accuracy vs rank k on OSR dataset.

4.3.2 *Outdoor Scene Recognition Dataset*

Dataset: We use Outdoor Scene Recognition (OSR) dataset from [21], and there are 2688 images from 8 scene categories, and it is described by high level attribute features.

Setting: In the experiment, we also use 30 images for each category as training data, and other images are used as testing data. Each time, we select training data randomly and we repeat this procedure 5 times, and use average accuracy as performance of each method.

Compared Methods: There are six compared methods as in last section, namely KNN, SVM, LMNN, LMNN+Trace norm, LMNN+Fantope and LMNN+capped trace norm.

Evaluation Metrics: Classification accuracy is compute as the performance of each method.

Results: In Figure 11, 1NN method reaches accuracy at about 65%, and SVM method perform a large increase by about 10%. Although the result of 1NN method using Mahalanobis distance learned by LMNN is a little worse than SVM method, LMNN method with low-rank regularization always outperform the result of SVM. We can also find out that our LMNN with capped trace regularization method works better than trace norm and Fantope regularization, and reaches an accuracy at about 76.5%. It improves LMNN method by about 1%. Figure 10b shows two sample query results of two compared methods. It is a little hard to distinguish coast and mountain when both of them has sky in the back, it is clear that our method learns a better Mahalanobis distance to do classification and image search.

We also plot figure to compare the performance of LMNN method with low-rank regularization, and we use LMNN with trace norm regularization as a baseline. In Figure 12, it is clear that our method always works better than metric learning with Fantope regularization. When we make $k = 40$, we can find that the performance of Fantope regularization is even worse than the baseline.

5. CONCLUSION

In this paper, we propose to use a novel low-rank regularization, capped trace norm regularization, to impose on metric learning method. Capped trace norm regularization is a better rank minimization approximation than trace norm. It works more stable than Fantope regularization and can be seen as an adaptive Fantope regularization as well. We also introduce an efficient optimization algorithm, and prove the convergence of our objective function. Experimental results show that our method outperforms the state-of-the-art metric learning methods.

6. REFERENCES

[1] 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.

[2] L. V. EE236C. 6. proximal gradient method.

[3] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. Liblinear: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874, 2008.

[4] H. Gao, F. Nie, W. Cai, and H. Huang. Robust capped norm nonnegative matrix factorization. *24th ACM International Conference on Information and Knowledge Management (CIKM 2015)*, pages 871–880, 2015.

[5] M. Guillaumin, J. Verbeek, and C. Schmid. Is that you? metric learning approaches for face identification. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 498–505. IEEE, 2009.

[6] M. Guillaumin, J. Verbeek, and C. Schmid. Multiple instance metric learning from automatically labeled bags of faces. In *Computer Vision–ECCV 2010*, pages 634–647. Springer, 2010.

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

[8] W. Jiang, F. Nie, and H. Huang. Robust dictionary learning with capped l1 norm. *Twenty-Fourth International Joint Conferences on Artificial Intelligence (IJCAI 2015)*, pages 3590–3596, 2015.

[9] D. Kedem, S. Tyree, F. Sha, G. R. Lanckriet, and K. Q. Weinberger. Non-linear metric learning. In *Advances in Neural Information Processing Systems*, pages 2573–2581, 2012.

[10] D. Kedem, Z. E. Xu, and K. Q. Weinberger. Gradient boosted large margin nearest neighbors.

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

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

[13] M. Law, N. Thome, and M. Cord. Quadruplet-wise image similarity learning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 249–256, 2013.

[14] M. T. Law, N. Thome, and M. Cord. Fantope regularization in metric learning. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1051–1058. IEEE, 2014.

[15] D. Lim, G. Lanckriet, and B. McFee. Robust structural metric learning. In *Proceedings of the 30th International Conference on Machine Learning*, pages 615–623, 2013.

[16] B. McFee and G. R. Lanckriet. Metric learning to rank. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 775–782, 2010.

[17] T. Mensink, J. Verbeek, F. Perronnin, and G. Csurka. Metric learning for large scale image classification: Generalizing to new classes at near-zero cost. In *Computer Vision–ECCV 2012*, pages 488–501. Springer, 2012.

[18] F. Nie, H. Huang, X. Cai, and C. Ding. Efficient and robust feature selection via joint $\ell_{2,1}$-norms minimization. *NIPS*, 2010.

[19] F. Nie, J. Yuan, and H. Huang. Optimal mean robust principal component analysis. In *Proceedings of the 31st International Conference on Machine Learning (ICML-14)*, pages 1062–1070, 2014.

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

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

[22] M. Schultz and T. Joachims. Learning a distance metric from relative comparisons. *Advances in neural information processing systems (NIPS)*, page 41, 2004.

[23] C. Theobald. An inequality for the trace of the product of two symmetric matrices. In *Mathematical Proceedings of the Cambridge Philosophical Society*, volume 77, pages 265–267. Cambridge Univ Press, 1975.

[24] H. Wang, F. Nie, and H. Huang. Robust distance metric learning via simultaneous l1-norm minimization and maximization. *The 31st International Conference on Machine Learning (ICML 2014)*, pages 1836–1844, 2014.

[25] J. Wang, X. Gao, Q. Wang, and Y. Li. Prodis-contshc: learning protein dissimilarity measures and hierarchical context coherently for protein-protein comparison in protein database retrieval. *BMC bioinformatics*, 13(Suppl 7):S2, 2012.

[26] 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.

[27] 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.

[28] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell. Distance metric learning with application to clustering with side-information. *Advances in neural information processing systems*, 15:505–512, 2003.

[29] T. Zhang. Multi-stage convex relaxation for learning with sparse regularization. *NIPS*, 2008.

[30] T. Zhang. Analysis of multi-stage convex relaxation for sparse regularization. *JMLR*, pages 1081–1107, 2010.