Neural Process Lett (2018) 48:1335–1345
https://doi.org/10.1007/s11063-017-9654-y

# Document Classification via Nonlinear Metric Learning

Xin Li[1,2] · Yanqin Bai[3] · Siyun Zhou[3] · Ying Li[4]

Published online: 14 June 2017

**Abstract** Learning a proper distance metric is an important problem in document classification, because the similarities of samples in many problems are usually measured by distance metric. In this paper, we address the nonlinear metric leaning problem with applying in the document classification. First, we propose a new representation about nonlinear metric by using a linear combination of some basic kernels. Second, we give a linear metric learning method by a triplet constraint and k-nearest neighbors, and then we develop it to a nonlinear method based on multiple kernel by above nonlinear metric. Further, the corresponding problem can be rewritten as an unconstrained optimization problem on positive definite matrices groups. At last, to ensure the learned distance matrix must be a positive definite matrix, we provide an improved intrinsic steepest descent algorithm with adaptive step-size to solve this unconstrained optimization. The experimental results show that our proposed method is effective on some document classification problems.

**Keywords** Nonlinear distance metric learning · Triplet constraint · Symmetric positive definite matrix · Intrinsic steepest descent

✉ Yanqin Bai
yqbai@t.shu.edu.cn

Xin Li
li_xin1129@163.com

Siyun Zhou
zhousiyun.eiko@outlook.com

Ying Li
yinglotus@t.shu.edu.cn

[1] School of Economics, Shanghai University, Shanghai 200444, China

[2] School of Mathematics and Statistics, Nanyang Normal University, Nanyang 473061, Henan, China

[3] Department of Mathematics, Shanghai University, Shanghai 200444, China

[4] School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China

## 1 Introduction

Document classification is an important research problem in information retrieval. Many classification techniques based on the category of machine learning have been proposed [1–7]. Support Vector Machine (SVM) [8] and k-Nearest Neighbor (k-NN) [9] are two popular methods, which have been widely applied in the classification of documents, such as text, image, etc. [10–13], yet they deeply rely on a good representation of the distance metric in the feature space. So it is important to find a desirable distance metric from the training data rather than only using Euclidean metric in document classification, Metric learning is proposed to address this issue.

Metric learning (ML) aims to learn a proper distance metric from the structure of training data to measure the distance or similarity between samples. Mahalanobis distance [14] is a most well-studied metric, which is usually learned by learning a symmetric positive semi-definite matrix. Xing et al. [15] propose the first metric learning method about Mahahabinous distance by pairwise constraints, the optimization problem is a convex optimization problem and solved by a projected gradient descent method. Weinberger and Saul [16] provide a method Large Margin Nearest Neighbors (LMNN), which is inspired by Neighbourhood Component Analysis (NCA) [17] and designed by using pairwise constraints and target neighbors. The optimization problem of LMNN can be reformulated to a semi-definite programming problem by introducing slack variables, while the introduced slack variables lead to much more constraints. These methods can efficiently improve the classification accuracy, however, the learned distance matrix at each iteration is not always a positive semi-definite matrix, so it needs to be projected onto a subspace of the positive semi-definite matrices for next iteration. Otherwise, the learned metrics are all linear metrics, which generally do not apply in the nonlinear problems, such as document classification problems.

Many nonlinear methods are proposed for metric learning. The kernel method is a widely-used method, and the main idea is the kernelization of linear metric learning methods. When a kernel function given, it will induce a Hilbert space, in which a linear metric can be learned by the metric learning based on kernel method. Some popular metric learning methods based kernel include Kernel Discriminative Component Analysis (KDCA) [18], Large Margin Component Analysis (LMCA) [19], Information-theoretic metric learning (ITML) [20]. The kernel method is also widely adopted in document classification [21,22]. However, their performance are affected by the selection of kernel function, then Multiple Kernel Learning (MKL) is proposed to address this problem. In MKL, a proper kernel is learned by a certain combination of some given basic kernels [23–25]. Inspired by MKL for SVM, a framework combining ML and MKL for nonlinear metric learning is proposed by Wang et al. [26], In this framework, a desirable Mahalanobis distance will be learned in a reproducing kernel Hilbert space (RKHS) induced by basic kernels. This framework is general, yet the efficiency of this framework is still deeply influenced by the adopted metric learning method in real world application.

As mentioned above, in this paper, we propose a nonlinear metric learning method to address document classification problem. The proposed optimization can be decomposed into two subproblems. The first problem is to learn a linear metric in kernel space induced by basic kernels, in which, we will learn a positive definite matrix rather than a positive semi-definite matrix on positive definite matrices groups to avoid the projection onto a subspace of the positive semi-definite matrices. The second problem is to learn the combination coefficients of basic kernels to provide a proper kernel function. By solving these two subproblems alternately, our optimization problem is resolved.

Our work is inspired by LMNN [16] and the methods of Wang et al. [26] and Ying et al. [27]. The main contributions of our work are summarized as follows. First, the proposed new formulation of nonlinear metric in our method is represented by a weighted linear combination of basic kernels, and this weighted linear combination can better reflects the relationships of basic kernels and expresses the distribution of data. Second, our optimization problem is proposed by adopting triplet constraint rather than pairwise constraint, which makes it can be solved without introducing the slack variables mentioned in [16], especially it can be transformed to an unconstrained optimization on positive define matrices groups. Last, our proposed algorithm is performed with an adaptive step-size rather than the constant step-size in Ying et al. [27], then it ensures that the searching step-size in each iteration must be the optimal step, which can avoid the problem that too large constant step-size will lead to reduce classification accuracy, or the problem that too small step-size will increase iteration times.

The rest of this paper is organized as follows. Section 2 reviews some related work, and introduces our nonlinear metric learning method, Sect. 3 proposes an improved intrinsic steepest descent algorithm, Sect. 4 presents our experimental results on document classification, and Sect. 5 concludes this paper.

## 2 Nonlinear Metric Learning Method with Multiple Kernel

### 2.1 Mahalanobis Distance via Multiple Kernel

Let $\mathcal{X} = \{x_1, \ldots, x_n\} \subseteq \mathbb{R}^n$ be the samples data, where $x_i \in \mathbb{R}^d$, $i = 1, \ldots, n$, $X = [x_1, \ldots, x_n]$ be data matrix corresponding to $\mathcal{X}$. Then the Mahalanobis distance is defined by

$$D(x_i, x_j) = \sqrt{(x_i - x_j)^T \mathbf{M} (x_i - x_j)} \tag{1}$$

where $\mathbf{M}$ is a symmetric positive semi-definite matrix, denoted by $\mathbf{M} \succeq 0$. $\mathbf{M}$ can be decomposed as $\mathbf{M} = L^T L$, $L : \mathbb{R}^d \rightarrow \mathbb{R}^d$ is a linear transformation, then learning $\mathbf{M}$ is equal to learning $L$.

Let $\mathcal{H}$ be the feature space of $\mathcal{X}$. A kernel function in $\mathcal{H}$ is defined by

$$k(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle = \phi(x_i)^T \phi(x_j), x_i, x_j \in \mathcal{X}$$

where $\phi : x \rightarrow \phi(x) \in \mathcal{H}$ is a nonlinear map. Kernel matrix is $\mathbf{K} = (k_{ij})_{n \times n}, k_{ij} = k(x_i, x_j)$, which is a symmetric positive semi-definite matrix. Then the Mahalanobis distance in $\mathcal{H}$ is

$$D(\phi(x_i), \phi(x_j)) = \sqrt{(\phi(x_i) - \phi(x_j))^T \mathbf{M} (\phi(x_i) - \phi(x_j))} \tag{2}$$

It is difficult to find a proper mapping $\phi$ for Eq. (2). Jain et al. [28] present an form $\mathbf{M} = \eta \mathbf{I} + \Phi(\mathbf{X}) \mathbf{A} \Phi(\mathbf{X})^T$, where $\mathbf{A} \succeq 0$, $\Phi(\mathbf{X}) = [\phi(x_1), \ldots, \phi(x_n)]$, $\mathbf{I}$ is the identity matrix and $\eta$ is a constant, usually $\eta = 0$. Then learning $\mathbf{M}$ in $\mathcal{H}$ only needs leaning $\mathbf{A}$. Let $K^i$ be the $i$-th column of $\mathbf{K}$, the square of (2) can be rewritten as

$$\begin{aligned} D^2(\phi(x_i), \phi(x_j)) &= (\phi(x_i) - \phi(x_j))^T \Phi(\mathbf{X}) \mathbf{A} \Phi(\mathbf{X})^T (\phi(x_i) - \phi(x_j)) \\ &= (K^i - K^j)^T \mathbf{A} (K^i - K^j) \end{aligned} \tag{3}$$

which is the expression of Mahalanobis distance in kernel space. From (3), if a kernel function is given, a kernel matrix $\mathbf{K}$ is also defined, and then it only needs learning $\mathbf{A}$ in the kernel space to solve nonlinear classification problem, the result has been proved in [26]. So the key problem is how to choose a proper kernel function. Cross-validation is a very common method, yet there are high computational costs.

Multiple kernel learning aims to learn a desirable kernel from some given basic kernels, which is an efficient method that not only can learn a proper kernel, but also can reduce the computational costs. The kernel function in MKL is usually defined as

$$k(x_i, x_j) = \sum_{s=1}^{m} \mu_s k_s(x_i, x_j), \quad \mu_s \geq 0. \tag{4}$$

where $\{k_s(x_i, x_j)\}_{i=1}^{m}$ are the basic kernels. By using this linear combination of basic kernel, we deduced (3) to a new representation in the kernel space induced by the basic kernels as follow.

$$\begin{aligned} D^2(\phi(x_i), \phi(x_j)) &= \sum_{s=1}^{m} \mu_s \left(K_s^i - K_s^j\right)^T \mathbf{A} \sum_{s=1}^{m} \mu_s \left(K_s^i - K_s^j\right) \\ &= \left(\sum_{s=1}^{m} \mu_s K_s^i - \sum_{s=1}^{m} \mu_s K_s^j\right)^T \mathbf{A} \left(\sum_{s=1}^{m} \mu_s K_s^i - \sum_{s=1}^{m} \mu_s K_s^j\right) \end{aligned} \tag{5}$$

where $K_s^j$ is the $i$-th column of kernel matrix corresponding to the $s$-th basic kernel. Then we have a kernel matrix $\mathbf{K} = \sum_{s=1}^{m} \mu_s K_s$. Now learning a Mahalanobis distance becomes learning $\mathbf{A}$ and the combination coefficients $\mu_i, i = 1, \ldots, m$ from kernel matrix $\mathbf{K}$.

### 2.2 Nonlinear Metric Learning Model

Let $\mathcal{Y} = \{y_1, \ldots, y_l\}$ be the class labels corresponding to $\mathcal{X}$. In metric learning, there are usually two constraints: pairwise constraint and triplet constraint. In pairwise constraint $(x_i, x_j)$, $x_i$ and $x_j$ are in the same class together or not, the metric is learned to satisfy keeping samples with same label closer while separating samples with different label farther. LMNN also learns metric by pairwise constraint, and the goal is that the k-nearest neighbors always belong to the same class while samples from different classes are separated by a large margin [16], so this method can efficiently increase the computation speed. In triplet constraint $(x_i, x_j, x_k)$, $(x_i, x_j)$ and $(x_i, x_k)$ are pairwise constraints, the metric is learned to make $x_i$ and $x_j$ closer than $x_i$ and $x_k$. Wang et al. propose a training goal to push $x_i$ and $x_j$ with the same label together and pull $x_i$ and $x_k$ with the different label apart so that the learned distance makes $x_j$ is closer to $x_i$ than $x_k$ to $x_i$ in [29], then a triplet constraint $\mathcal{T} = \{(x_i, x_j, x_k) : 1 + D_{ij}^2 < D_{ik}^2\}$ is given. Yet the constraint $\mathcal{T}$ is influenced very much by the distribution of data. Ying et al. [27] revise $\mathcal{T}$ to

$$\mathcal{T}' = \{(x_i, x_j, x_k) : D_{ij}^2 < \gamma D_{ik}^2, 0 < \gamma < 1\} \tag{6}$$

This triplet constraint can balance the influence between the inner-class data and inter-class data by parameter $\gamma$, and then the limitation of $\mathcal{T}$ depending on the distribution of data is overcome.

When learning Mahalanobis distance, our goal is to minimize distances between input sample and its target neighbors, while keeping the distances between input sample and its target neighbors with same label are smaller than the distances between input sample and its target neighbors with different label, i.e. input sample and its target neighbors need satisfying triplet constraint $\mathcal{T}'$. Therefor, we propose a linear supervised metric learning method as follow.

$$\min_{\mathbf{A}} \ \lambda \sum_{i,j\in\mathcal{N}(i)} D_{ij}^2 + (1-\lambda) \sum_{i,j,k\in\mathcal{N}(i),\mathcal{T}'} \left(D_{ij}^2 - \gamma D_{ik}^2\right) \tag{7}$$

$$\text{s.t.} \quad \mathbf{A} \succ 0. \tag{8}$$

where $\mathcal{N}(i)$ is the nearest neighbor list of $x_i$, $\lambda$ is a trade-off parameter between the first term and the second term, $0 \leq \lambda \leq 1$, and $\mathbf{A}$ is a positive definite matrix, denoted by $\mathbf{A} \succ 0$. Here the constrain is $\mathbf{A} \succ 0$ rather than $\mathbf{A} \succeq 0$, then we do not need to introduce a projection to ensure $\mathbf{A}$ must be a positive semi-definite matrix after each iteration. The first term of cost function ensures that the distances between input sample and its target neighbors are sufficiently small, The second term means pulling input sample and its target neighbors with same labels closer while pushing input sample and its target neighbors with different labels farther.

Similar to LMNN, we define $\eta_{ij}$ as follows. If $x_j$ is a target neighbor of $x_i$, then $\eta_{ij} = 1$, otherwise $\eta_{ij} = 0$. A character matrix is defined as $H = (\eta_{ij})_{n\times n}$. We also define $y_{ij}$ as follows. If $x_i$ and $x_j$ have the same label, then $y_{ij} = 1$, otherwise $y_{ij} = 0$. An indicator matrix is defined as $Y = (y_{ij})_{n\times n}$. Then, by using the Mahalanobis distance (6), we develop the problem (7)–(8) to a nonlinear optimization problem as follows in the kernel space induced by basic kernels.

$$\min_{\mathbf{A},\mu} \ \lambda \sum_{i=1}^{n} \sum_{j=1}^{n} \eta_{ij} D_{ij}^2 + (1-\lambda) \sum_{i,j,k=1}^{n} \eta_{ij}\,(1-y_{ik}) \left(D_{ij}^2 - \gamma D_{ik}^2\right) \tag{9}$$

$$\text{s.t.} \quad \mathbf{A} \succ 0, \mu_s \geq 0, \sum_{s=1}^{m} \mu_s = 1. \tag{10}$$

For $\mathbf{A} \succ 0$, it can be decomposed into $\mathbf{A} = QQ^T$, further we rewrite (6) to

$$\begin{aligned} D^2(\phi(x_i), \phi(x_j)) &= \left(\sum_{s=1}^{m} \mu_s K_s^i - \sum_{s=1}^{m} \mu_s K_s^j\right)^T QQ^T \left(\sum_{s=1}^{m} \mu_s K_s^i - \sum_{s=1}^{m} \mu_s K_s^j\right) \\ &= \sum_{s,t=1}^{m} \mu_s \mu_t \left({K_s^i}^T QQ^T K_t^i + {K_s^j}^T QQ^T K_t^j - 2{K_s^i}^T QQ^T K_t^j\right) \end{aligned}$$

Then, in problem (9)–(10), the first term can be reformulated to

$$\begin{aligned} \sum_{i=1}^{n} \sum_{j=1}^{n} \eta_{ij} D_{ij}^2 &= 2 \sum_{s,t=1}^{m} \mu_s \mu_t \sum_{i,j=1}^{n} \eta_{ij} \left({K_s^i}^T QQ^T K_t^i - {K_s^i}^T QQ^T K_t^j\right) \\ &= 2tr\left(Q^T \mathbf{K}(D_H - H)\mathbf{K}^T Q\right) = 2tr\left(\mathbf{K}(D_H - H)\mathbf{K}^T \mathbf{A}\right) \end{aligned}$$

where $D_H = diag(He)$, $e$ is a column vector with all elements 1, $tr$ is the trace operator. By using the same way, the second term can be reformulated to

$$\sum_{i,j,k}^{n} \eta_{ij}\,(1-y_{ik}) \left(D_{ij}^2 - \gamma D_{ik}^2\right) = 2tr\left(Q^T \mathbf{K}\,(D_c - C)\,\mathbf{K}^T Q\right) = 2tr\left(\mathbf{K}\,(D_c - C)\,\mathbf{K}^T \mathbf{A}\right)$$

where $D_c = diag(Ce)$, $C = diag((ee^T - Y)e)H - \gamma diag(He)(ee^T - Y)$.

Now we rewrite problem (9)–(10) as follows.

$$\min_{\mathbf{A},\mu} \ \lambda tr\left(\mathbf{K}\left(D_H - H\right)\mathbf{K}^T\mathbf{A}\right) + (1-\lambda)\, tr\left(\mathbf{K}\left(D_c - C\right)\mathbf{K}^T\mathbf{A}\right) \tag{11}$$

$$\text{s.t.} \quad \mathbf{A} \succ 0, \mu_s \geq 0, \sum_{s=1}^{m} \mu_s = 1. \tag{12}$$

## 3 Algorithms

The optimization problem (11)–(12) can be decomposed into two subproblems. Given $\mu$, one is problem (13)–(14).

$$\min_{\mathbf{A}} \ \lambda tr\left(\mathbf{K}\left(D_H - H\right)\mathbf{K}^T\mathbf{A}\right) + (1-\lambda)\, tr\left(\mathbf{K}\left(D_c - C\right)\mathbf{K}^T\mathbf{A}\right) \tag{13}$$

$$\text{s.t.} \ \ \mathbf{A} \succ 0 \tag{14}$$

which is an optimization problem about the positive definite matrix $\mathbf{A}$ in the kernel space. Though the cost function is linear in $\mathbf{A}$, however, the positive definiteness of the learned $\mathbf{A}$ in each iteration still can not be ensured in kernel space. Below we will treat this subproblem as an unconstrained optimization problem on positive define matrices groups, where the learned $\mathbf{A}$ in each iteration must be a positive definite matrix.

Given $\mathbf{A}$, another subproblem is (15)–(16).

$$\min_{\mu} \ \lambda tr\left(\mathbf{K}\left(D_H - H\right)\mathbf{K}^T\mathbf{A}\right) + (1-\lambda)\, tr\left(\mathbf{K}\left(D_c - C\right)\mathbf{K}^T\mathbf{A}\right) \tag{15}$$

$$\text{s.t.} \quad \mu_s \geq 0, \sum_{s=1}^{m} \mu_s = 1 \tag{16}$$

which can be rewritten as a quadratic programming problem as follows.

$$\begin{aligned} \min_{\mu} \ \lambda \sum_{s,t=1}^{m} \mu_s\mu_t tr\left(K_s\left(D_H - H\right)K_t{}^T\mathbf{A}\right) & \\ + (1-\lambda)\sum_{s,t=1}^{m} \mu_s\mu_t tr\left(K_s\left(D_c - C\right)K_t{}^T\mathbf{A}\right) & \qquad (17) \\ \text{s.t.} \quad \mu_s \geq 0, \sum_{s=1}^{m} \mu_s = 1 & \qquad (18) \end{aligned}$$

So we can learn $\mathbf{A}$ and $\mu$ by an alternating process which is similar to the two-step iterative algorithm proposed in [26], and the convergence is guaranteed in [30].

Let us denote $f(\mathbf{A}) = \lambda tr(\mathbf{K}(D_H - H)\mathbf{K}^T\mathbf{A}) + (1-\lambda)tr(\mathbf{K}(D_c - C)\mathbf{K}^T\mathbf{A})$. For $f(\mathbf{A})$ is linear in $\mathbf{A}$, whose gradient can be calculated by

$$\nabla f(\mathbf{A}) = \lambda\left((\mathbf{K}(D_H - H)\mathbf{K}^T\mathbf{A})\right)^T + (1-\lambda)\left(\mathbf{K}(D_c - C)\mathbf{K}^T\right)^T \tag{19}$$

Our works focus on the solution of problem (13)–(14). As mentioned in [27], on the positive definite matrices groups $\{P \in R^{n\times n} | P = P^T, P \succ 0\}$, a geodesic $P(t)$ can be defined by

$$P(t) = P^{\frac{1}{2}} \exp\left(tP^{-\frac{1}{2}}SP^{-\frac{1}{2}}\right)P^{\frac{1}{2}} \tag{20}$$

Algorithm 1. Intrinsic Steepest Descent Algorithm with Adaptive Step-size

```
Initialization.
Given kernel matrices $K_1, \ldots, K_m$, coefficient vector $\mu$,
precision $\epsilon > 0$ and initial matrix $\mathbf{A} = I$.
$\mathbf{K} = \sum_{s=1}^{m} \mu_s K_s$.
Repeat
    $S(t) = \frac{1}{2}((\operatorname{grad} f)(\mathbf{A}(t)) + (\operatorname{grad} f)(\mathbf{A}(t))^T)$,
    $G(t) = \mathbf{A}(t)^{-\frac{1}{2}} S(t) \mathbf{A}(t)^{-\frac{1}{2}}$,
    searching $\alpha(t)$ by $\alpha(t) = \arg\min_{\alpha \geq 0} f(\mathbf{A}(t)^{\frac{1}{2}} \exp(-\alpha G(t)) \mathbf{A}(t)^{\frac{1}{2}})$,
    $\mathbf{A}(t+1) = \mathbf{A}(t)^{\frac{1}{2}} \exp(-\alpha(t) \cdot G(t)) \mathbf{A}(t)^{\frac{1}{2}}$,
    $t := t + 1$ .
Until $\| f(\mathbf{A}(t+1)) - f(\mathbf{A}(t)) \| \leq \varepsilon$.
Output $\mathbf{A}$.
```

where exp is the exponential map, $P(0) = P \in \mathcal{P}(n)$, $\dot{P}(0) = S \in T_P\mathcal{P}(n)$ is a direction, $T_P\mathcal{P}(n)$ is the tangent space at point $P$.

Therefore $\mathbf{A}$ can be learned by

$$\mathbf{A}(t+1) = \mathbf{A}(t)^{\frac{1}{2}} \exp\left[\alpha(t) \cdot \mathbf{A}(t)^{-\frac{1}{2}} S(t) \mathbf{A}(t)^{-\frac{1}{2}}\right] \mathbf{A}(t)^{\frac{1}{2}} \tag{21}$$

where $\alpha(t)$ is the optimal step size at time $t$, and $\mathbf{A}(t)^{-\frac{1}{2}} S(t) \mathbf{A}(t)^{-\frac{1}{2}}$ is a descent direction. On positive definite matrices groups, an intrinsic steepest descent algorithm is proposed by Ying et al. [27], and this algorithm can ensure that $\mathbf{A}(t+1)$ in (21) must be a positive definite matrix. However, the step size $\alpha(t)$ is set to a constant, which will lead to the classification accuracy reducing when the constant is larger, or lead to iteration times increasing when the constant is smaller. To address these problems, we propose an intrinsic steepest descent algorithm with adaptive step-size. In our algorithm, we use (22) to search an optimal value $\alpha(t)$ as the next step size. Furthermore, we set a precision parameter $\epsilon > 0$, when it satisfied, the iteration will be stopped.

$$\alpha(t) = \arg\min_{\alpha \geq 0} f\left(\alpha(t)^{\frac{1}{2}} \exp(-tG(t)) \alpha(t)^{\frac{1}{2}}\right) \tag{22}$$

Our proposed algorithm is an improvement of the algorithm in [27], which is presented in Algorithm 1. We have proved that our algorithm converges linearly by a special Taylor's formula on positive definite matrices groups, the result is not described here for the limitation of space.

## 4 Experimental Results

To evaluate the performance of our proposed nonlinear metric learning method, we perform our method for document classification by k-NN classifier on two real document data sets: USPS handwritten digit dataset[1] and C-Cube handwriting dataset[2]. The document features of each sample are represented by a fixed-length feature vector. Each dataset is randomly

[1] https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/.

[2] http://www.idiap.ch/dataset/ccc/.

split into two parts: percent 70 for training, the rest percent 30 for testing. The basic kernels are used by ten polynomial kernels with degree from one to ten and ten Gaussian kernels with bandwidth $\sigma \in \{0.5, 1, 2, 5, 7, 10, 12, 15, 17, 20\}$, same as in [26,31]. All experimental results are presented by the average over 30 runs.

We choose k-NN with Euclidean distance as the baseline to compare our method (SIMKML, for short) with the following methods: (1) LMNN [16], (2) ITML [20], (3) the method to optimization problem (7)–(8) (SIML, for short), (4) a kernel method using best single kernel to problem (7)–(8) (BSKML, for short), where the best kernel is chosen by cross-validation, (5) a kernel method with a composite kernel to problem (9)–(10)(USMKML, for short), where the kernel is defined as the unweighted sum of all basic kernels, i.e. $\mu = \mathbf{1}$. (6) a kernel method with the concatenation kernel matrices (CMKML, for short), where the kernel matrix is the unweighted concatenation of all basic kernel matrices, $\mathbf{K} = (K_1^T, \ldots, K_m^T)^T$, which is similar to *NR-ML$_h$-MKL$_\mu$* in [26]. The dimensionality of $\mathbf{K}$ is $mn \times n$, so we just use the polynomial kernels of degree from one to two and Gaussian kernel with $\sigma = 0.5$ in the experiment.

All the experimental results with 1-NN are presented in Table 1. The classification error rates with different methods over 30 times are displayed by box-plots in Fig. 1. The changing of classification error rates with different "k" in k-NN classifier on USPS and C-Cube are shown in Fig. 2.

*USPS dataset* The United State Postal Service (USPS) database is a kind of handwritten digit data set. There are 7319 training samples, 2007 testing samples and ten classes from 0 to 9. Each image is normalized to grayscale image of size 16 × 16. In the experiment, 1000 data are randomly chosen from USPS, then 70% of it are randomly chosen again for training, the rest 30% are for testing. From Table 1 and Fig. 1a, we see that classification error rate of SIMKML is higher than LMNN, the result indicates that USPS data set may be more suitable to the linear method LMNN. We also observe that our method has the second lowest classification error rate, which shows that it is a reasonable nonlinear metric learning. The results of Fig. 2a show that the proposed method is sensitive to the number "k" of k-NN classifiers on USPS dataset as other methods.

*C-Cube dataset* Cursive Character Challenge (C-Cube) is a new benchmark for machine learning and pattern recognition algorithms, which is formed by 57,293 cursive characters extracted from handwritten words, including both upper and lower case versions of each letter [32]. C-Cube has been randomly split into a training set with 38,160 characters and a test set with 19,133 characters. In the experiments, we choose the lower case versions from a to z as experiment data, which contains 11,162 training binary images and 22,273 testing binary images in 26 clusters. So the distribution of C-Cube Data is more nonlinear than USPS data which is just composed of numbers from 0 to 9. For the different sizes of images, by

**Table 1** Classification average error rates on document datasets (mean ± SD, in %)

| Dataset | Euclidean | LMNN | ITML | SIML |
|---|---|---|---|---|
| USPS | 7.58 ± 1.17 | 6.29 ± 1.00 | 7.38 ± 1.21 | 7.00 ± 1.25 |
| C-Cube | 57.19 ± 1.88 | 49.51 ± 1.90 | 53.33 ± 1.82 | 84.32 ± 4.90 |
| Dataset | BSKML | USMKML | CMKML | SIMKML |
| USPS | 6.84 ± 1.30 | 6.87 ± 1.27 | 7.50 ± 1.03 | 6.83 ± 1.07 |
| C-Cube | 38.71 ± 2.01 | 42.47 ± 1.93 | 39.70 ± 1.70 | 38.60 ± 1.84 |

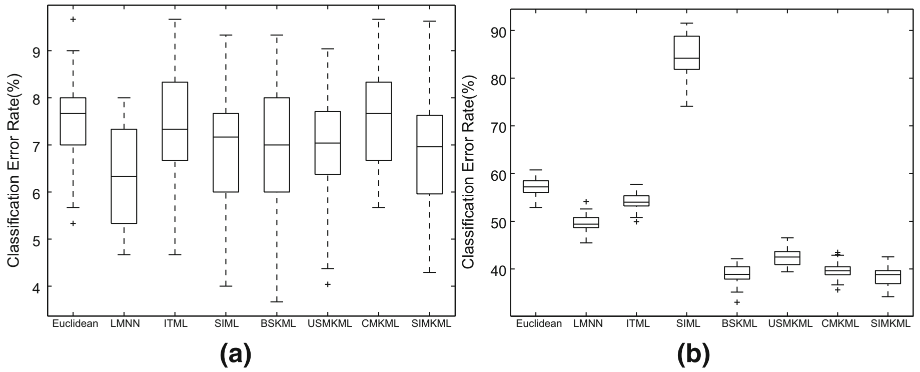

**Fig. 1** Classification error rates of different metrics on **a** USPS and **b** C-Cube. The eight methods are (1) Euclidean, (2) LMNN, (3) ITML, (4) SIML, (5) BSKML, (6) USMKML, (7) CMKML, (8) SIMKML

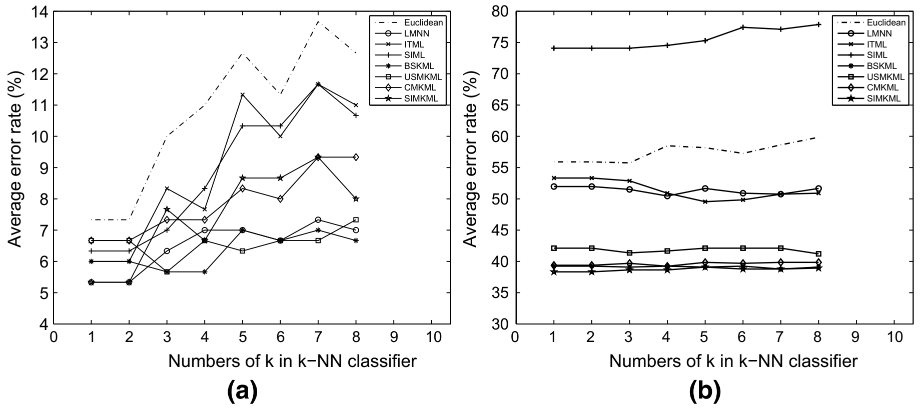

**Fig. 2** The changing of classification error rates with different “k” in k-NN classifier on **a** USPS and **b** C-Cube

calibrating the centers of images and filling the missing parts by zeros, all the images are normalized into the same size 209 × 123. Furthermore, 1000 data are randomly chosen from C-Cube, 70% of it are randomly chosen as training set and the rest 30% as testing set. From Table 1 and Fig. 1b, we observe that our proposed linear method is not a proper method to C-Cube data set for the higher classification error rate, but the proposed nonlinear method SIMKML outperforms all the other methods, which shows that our nonlinear method is an efficiently method for nonlinear classification problem by using the revised triplet constraint, multiple kernel and intrinsic steepest descent algorithm with adaptive iteration step size. From Fig. 2b, we see that our method is stable and not sensitive to the number of classifiers, which shows that our proposed nonlinear method is more suitable to nonlinear data C-Cube.

## 5 Conclusion

In this paper, we propose a nonlinear metric learning method based on multiple kernel learning for document classification. By defining a kernel with a weighted linear combination of basic kernels, a new expression of distance is presented. Then by using k-nearest neighbor and a revised triplet constraint, an optimization model for classification is proposed, which avoids to introduce slack variables and reduces the computational costs. Further, an intrinsic steepest descent algorithm on positive definite matrices groups is provided, the searching step is adopted with optimal value, which ensures classification accuracy will not be reduced. The experiment results show that, our method not only have a good effectiveness for the classification of documents whose features are represented by a fixed-length feature vector, but also can provide an alternative method to choose a proper kernel function for nonlinear classification problem.

For the future work, one is to design parallel algorithm based on our proposed algorithm to improve the computational performance, the other is the research for document classification based on semi-supervised metric learning.

**Acknowledgements** The research is supported by the National Natural Science Foundation of China (Grant Nos. 11371242, 11471208, 61573274, 11601315), Shanghai Sailing Program (Grant No. 16YF1404000).

## References

1. Schneider KM (2005) Techniques for improving the performance of naive bayes for text classification. In: International conference on intelligent text processing and computational linguistics. Springer, Berlin, pp 682–693
2. Klopotek MA, Woch M (2003) Very large Bayesian networks in text classification. In: International conference on computational science. Springer, Berlin, pp 397–406
3. Siolas G, Dalchebuc F (2000) Support vector machines based on a semantic kernel for text categorization. In: Proceedings of the IEEE-INNS-ENNS international joint conference on neural networks. IEEE, pp 205–209
4. Shanahan JG, Roma N (2003) Improving SVM text classification performance through threshold adjustment. In: European conference on machine learning. Springer, Berlin, pp 361–372
5. Johnson DE, Oles FJ, Zhang T, Goetz T (2002) A decision-tree-based symbolic rule induction system for text categorization. IBM Syst J 41(3):428–437
6. Nigam K, McCallum AK, Thrun S, Mitchell T (2000) Text classification from labeled and unlabeled documents using EM. Mach Learn 39:103–134
7. Lim HS (2004) Improving kNN based text classification with well estimated parameters. In: International conference on neural information processing. Springer, Berlin, pp 516–523
8. Vapnik VN, Vapnik V (1998) Statistical learning theory. Wiley, New York
9. Tam V, Santoso A, Setiono R (2002) A comparative study of centroid-based, neighborhood-based and statistical approaches for effective document categorization. In: International conference on pattern recognition. IEEE, pp 235–238
10. Tong S, Koller D (2001) Support vector machine active learning with applications to text classification. J Mach Learn Res 2:45–66
11. Bijalwan V, Kumar V, Kumari P, Pascual J (2014) KNN based machine learning approach for text and document mining. Int J Database Theory Appl 7(1):61–70
12. Gao Y, Wang M, Zha ZJ, Shen J, Li X, Wu X (2013) Visual-textual joint relevance learning for tag-based social image search. IEEE Trans Image Process 22(1):363–376
13. Gao Y, Ji R, Cui P, Dai Q, Hua G (2014) Hyperspectral image classification through bilayer graph-based learning. IEEE Trans Image Process 23(7):2769–2778
14. Mahalanobis P (1936) On the generalized distance in statistics. In: Proceedings of the National Institute of Sciences (Calcutta), vol 2. pp 49–55
15. Xing EP, Ng AY, Jordan MI, Russell S (2002) Distance metric learning with application to clustering with side-information. In: Advances in neural information processing systems. The MIT Press Publication, pp 505–512
16. Weinberger KQ, Saul LK (2009) Distance metric learning for large margin nearest neighbor classification. J Mach Learn Res 10:207–244
17. Goldberger J, Roweis S, Hinton GE, Salakhutdinov R (2004) Neighbourhood components analysis. In: Advances in neural information processing systems. The MIT Press Publication, pp 513–520

18. Hoi SC, Liu W, Lyu MR, Ma WY (2006) Learning distance metrics with contextual constraints for image retrieval. In: IEEE computer society conference on computer vision and pattern recognition, vol 2. IEEE, pp 2072–2078
19. Torresani L, Lee KC (2007) Large margin component analysis. In: Advances in neural information processing systems. The MIT Press Publication, pp 1385–1392
20. Davis JV, Kulis B, Jain P, Sra S, Dhillon IS (2007) Information-theoretic metric learning. In: Proceedings of the 24th international conference on machine learning. ACM, pp 209–216
21. Zhang D, Chen X, Lee WS (2005) Text classification with kernels on the multinomial manifold. In: Proceedings of the 28th annual international ACM SIGIR conference on research and development in information retrieval. ACM, pp 266–273
22. Jain P, Kulis B, Davis JV, Dhillon IS (2012) Metric and kernel learning using a linear transformation. J Mach Learn Res 13(1):519–547
23. Lanckriet GR, Cristianini N, Bartlett P, Ghaoui LE, Jordan MI (2004) Learning the kernel matrix with semidefinite programming. J Mach Learn Res 5:27–72
24. Rakotomamonjy A, Bach FR, Canu S, Grandvalet Y (2008) SimpleMKL. J Mach Learn Res 9:2491–2521
25. Gonen M, Alpaydin E (2011) Multiple kernel learning algorithms. J Mach Learn Res 12:2211–2268
26. Wang J, Do HT, Woznica A, Kalousis A (2011) Metric learning with multiple kernels. In: Advances in neural information processing systems. The MIT Press Publication, pp 1170–1178
27. Ying SH, Wen ZJ, Shi J, Peng YX, Peng JG (2017) Manifold preserving: an intrinsic approach for semi-supervised distance metric learning. IEEE Trans Neural Netw Learn Syst. doi:10.1109/TNNLS.2017.2691005
28. Jain P, Kulis B, Dhillon IS (2010) Inductive regularized learning of kernel functions. In: Advances in neural information processing systems. The MIT Press Publication, pp 946–954
29. Wang Q, Yuen PC, Feng G (2013) Semi-supervised metric learning via topology preserving multiple semi-supervised assumptions. Pattern Recognit 46(9):2576–2587
30. Grippo L, Sciandrone M (2000) On the convergence of the block nonlinear GaussCSeidel method under convex constraints. Oper Res Lett 26(3):127–136
31. Gai K, Chen G, Zhang CS (2010) Learning kernels with radiuses of minimum enclosing balls. In: Advances in neural information processing systems. The MIT Press Publication, pp 649–657
32. Camastra F, Spinetti M, Vinciarelli A (2006) Offline cursive character challenge: a new benchmark for machine learning and pattern recognition algorithms. In: International conference on pattern recognition. IEEE, pp 913–916