

Research Article

Low-Rank Kernel-Based Semisupervised Discriminant Analysis

Baokai Zu,^{1,2} Kewen Xia,^{1,2} Shuidong Dai,^{1,2} and Nelofar Aslam^{1,2}

¹School of Electronic and Information Engineering, Hebei University of Technology, Tianjin 300401, China

²Key Lab of Big Data Computation of Hebei Province, Tianjin 300401, China

Correspondence should be addressed to Kewen Xia; kwxia@hebut.edu.cn

Received 16 April 2016; Accepted 14 June 2016

Academic Editor: Yu Cao

Copyright © 2016 Baokai Zu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Semisupervised Discriminant Analysis (SDA) aims at dimensionality reduction with both limited labeled data and copious unlabeled data, but it may fail to discover the intrinsic geometry structure when the data manifold is highly nonlinear. The kernel trick is widely used to map the original nonlinearly separable problem to an intrinsically larger dimensionality space where the classes are linearly separable. Inspired by low-rank representation (LRR), we proposed a novel kernel SDA method called low-rank kernel-based SDA (LRKSDA) algorithm where the LRR is used as the kernel representation. Since LRR can capture the global data structures and get the lowest rank representation in a parameter-free way, the low-rank kernel method is extremely effective and robust for kinds of data. Extensive experiments on public databases show that the proposed LRKSDA dimensionality reduction algorithm can achieve better performance than other related kernel SDA methods.

1. Introduction

For many real world data mining and pattern recognition applications, the labeled data are very expensive or difficult to obtain, while the unlabeled data are often copious and available. So how to use both labeled and unlabeled data to improve the performance becomes a significant problem [1, 2]. Recently, semisupervised dimensionality reduction has attracted considerable attention, which can be directly used in the whole database [3]. Illuminated by semisupervised learning (SSL), many methods have been put forward to relieve the so-called small sample size (SSS) problem of LDA [4, 5]. Semisupervised Discriminant Analysis (SDA) first is proposed by Cai et al. [2], which can easily resolve the out-of-sample problem [6] and is more suitable for the real world applications. In SDA algorithm, the labeled samples are used to maximize the different classes' separability and the unlabeled ones to estimate the data's intrinsic geometric information.

Semisupervised Discriminant Analysis may fail to discover the intrinsic geometry structure when the data manifold is highly nonlinear [2, 7]. The kernel trick [8] has been widely used to generalize linear dimensionality reduction

algorithms to nonlinear ones, which maps the original nonlinearly separable problem to an intrinsically larger dimensionality space where the classes are linearly separable. So the kernel SDA (KSDA) [2, 7] can discover the underlying subspace more exactly in the feature space, which brings a better subspace for the classification task by a nonlinear learning technique. Cai et al. discussed how to perform SDA in Reproducing Kernel Hilbert Space (RKHS), which gives rise to kernel SDA [2]. You et al. have presented the derivations of a first approach to optimize the parameters of a kernel. It can map the original class distributions to a space where these are optimally (with respect to Bayes) separated with a hyperplane [7]. A new kernel-based nonlinear discriminant analysis algorithm is proposed to solve the fundamental limitations in LDA [9]. A novel KFDA kernel parameters optimization criterion is presented for maximizing the uniformity of class-pair separabilities and class separability in kernel space simultaneously [10]. To overcome the nonlinear dimensionality reduction problems and adopting multiple features restrictions of LFDA, Wang and Sun proposed a new dimensionality reduction algorithm called multiple kernel local Fisher discriminant analysis (MKLFDA) based on the multiple kernel learning [11].

The kernelization of graph embedding applies the kernel trick on the linear graph embedding algorithm to handle data with nonlinear distributions [12]. Weinberger et al. described an algorithm for nonlinear dimensionality reduction based on semidefinite programming and kernel matrix factorization which learns a kernel matrix for high dimensional data that lies on or near a low-dimensional manifold [13].

Low-rank matrix decomposition and completion are recently becoming very popular since Yang et al. and Chen et al. proved that a robust estimation of an underlying subspace which can be obtained by decomposing the observations into a low-rank matrix and a sparse error matrix [14, 15]. Recently, Liu et al. propose a low-rank representation method which is robust to noise and data corruptions due to its ability to decompose noise from the data set [14]. More recently, low-rank representation [16, 17], as a promising method to capture the underlying low-dimensional structures of data, has attracted much attention in the pattern analysis and signal processing communities. LRR method [16–18] seeks the lowest rank representation of all data jointly, such that each data point can be represented as a linear combination of some bases.

The major problem of kernel methods is to find the proper kernel parameters. But all these kernel methods usually use fixed global parameters to determinate the kernel matrix, which are very sensitive to the parameters setting. In fact, the most suitable kernel parameters may vary greatly at different random distribution of the same data. Moreover, the kernel mapping of KSDA always analyze the relationship of the data using the mode one-to-others, which emphasizes local information and lacks global constraints on their solutions. These shortcomings limit the performance and efficiency of KSDA methods. To overcome the disadvantages of the traditional kernel methods, inspired by LRR, we proposed a novel kernel-based Semisupervised Discriminant Analysis called low-rank kernel-based SDA (LRKSDA) where the low-rank representation is used as the kernel method. Compared with other kernels, the low-rank kernel jointly obtains the representation of all the samples under a global low-rank constraint [19]. Thus it is better at capturing the global data structures and very robust to different random distribution of the data set. In addition, we can get the lowest rank representation in a parameter-free way, which is very convenient and robust for kinds of data. Extensive experiments on public databases show that our proposed LRKSDA dimensionality reduction algorithm can achieve better performance than other related methods.

The rest of the paper is organized as follows. We start by a brief review on an overview of SDA in Section 2. We then introduce the low-rank kernel-based SDA framework in Section 3. Then Section 4 reports the experiment results on real world database tasks. In Section 5, we conclude the paper.

2. Overview of SDA

Given a set of samples $\{\mathbf{x}_1, \dots, \mathbf{x}_m, \mathbf{x}_{m+1}, \dots, \mathbf{x}_{m+l}\}$, where $N = m + l$, the first m samples are labeled as $\{\mathbf{y}_1, \dots, \mathbf{y}_m\}$, and the remaining l are unlabeled ones. They all belong to c

classes. The SDA [2] hopes to find a rejection matrix \mathbf{a} , which motivates us to present the prior assumption of consistency by a regularizer term. The objective function is as follows:

$$\mathbf{a} = \max_{\mathbf{a}} \frac{\mathbf{a}^T \mathbf{S}_b \mathbf{a}}{\mathbf{a}^T \mathbf{S}_t \mathbf{a} + \alpha J(\mathbf{a})}, \quad (1)$$

where \mathbf{S}_b and \mathbf{S}_t are the between class scatter and total class scatter matrix. And \mathbf{S}_w is defined as the within class scatter matrix

$$\begin{aligned} \mathbf{S}_w &= \sum_{k=1}^c \left(\sum_{i=1}^{l_k} (\mathbf{x}_i^{(k)} - \mu^{(k)}) (\mathbf{x}_i^{(k)} - \mu^{(k)})^T \right), \\ \mathbf{S}_b &= \sum_{k=1}^c l_k (\mu^{(k)} - \mu) (\mu^{(k)} - \mu)^T, \\ \mathbf{S}_t &= \sum_{i=1}^l (\mathbf{x}_i - \mu) (\mathbf{x}_i - \mu)^T, \end{aligned} \quad (2)$$

where μ is the mean vector of the total sample, l_k is the number of samples in the k th class, $\mu^{(k)}$ is the average vector of the k th class, and $\mathbf{x}_i^{(k)}$ is the i th sample in the k th class.

The parameter α in (1) balances the model complexity and the empirical loss. The regularizer term supplies us with the flexibility to incorporate the prior knowledge in the applications. We aim at constructing $J(\mathbf{a})$ graph combining the manifold structure through the available unlabeled samples [2]. The key of SSL algorithm is the prior assumption of consistency. For classification, it means that the nearby samples are likely to have same label [20]. And for dimensionality reduction, it implicates that the nearby samples have similar embeddings (low-dimensional representations).

Given a set of samples $\{\mathbf{x}_i\}_{i=1}^m$, we can construct the graph \mathbf{G} to represent the relationship between nearby samples by k NN algorithm. Then put an edge between k nearest neighbors of each other. The corresponding weight matrix \mathbf{S} is defined as follows:

$$S_{ij} = \begin{cases} 1, & \text{if } \mathbf{x}_i \in N_k(\mathbf{x}_j) \text{ or } \mathbf{x}_j \in N_k(\mathbf{x}_i) \\ 0, & \text{otherwise,} \end{cases} \quad (3)$$

where $N_k(\mathbf{x}_i)$ denotes the set of k nearest neighbors of \mathbf{x}_i . Then $J(\mathbf{a})$ term can be defined as follows:

$$\begin{aligned} J(\mathbf{a}) &= \sum_{ij} (\mathbf{a}^T \mathbf{x}_i - \mathbf{a}^T \mathbf{x}_j)^2 S_{ij} \\ &= 2 \sum_i \mathbf{a}^T \mathbf{x}_i \mathbf{D}_{ii} \mathbf{x}_i^T \mathbf{a} - 2 \sum_{ij} \mathbf{a}^T \mathbf{x}_i S_{ij} \mathbf{x}_j^T \mathbf{a} \\ &= 2 \mathbf{a}^T \mathbf{X} (\mathbf{D} - \mathbf{S}) \mathbf{X}^T \mathbf{a} = 2 \mathbf{a}^T \mathbf{X} \mathbf{L} \mathbf{X}^T \mathbf{a}, \end{aligned} \quad (4)$$

where \mathbf{D} is a diagonal matrix whose entries are column (or row since \mathbf{S} is symmetric) sum of \mathbf{S} ; that is, $\mathbf{D}_{ii} = \sum_j S_{ij}$. The Laplacian matrix [21] is $\mathbf{L} = \mathbf{D} - \mathbf{S}$.

We can get the objective function of the SDA with regularizer term $J(\mathbf{a})$ [2]:

$$\mathbf{a} = \max_{\mathbf{a}} \frac{\mathbf{a}^T \mathbf{S}_b \mathbf{a}}{\mathbf{a}^T (\mathbf{S}_t + \alpha \mathbf{X} \mathbf{L} \mathbf{X}^T) \mathbf{a}}. \quad (5)$$

By maximizing the generalized eigenvalue problem, we can obtain the projective vector \mathbf{a} :

$$\mathbf{S}_b \mathbf{a} = \lambda (\mathbf{S}_t + \alpha \mathbf{X} \mathbf{L} \mathbf{X}^T) \mathbf{a}. \quad (6)$$

3. Low-Rank Kernel-Based SDA Framework

3.1. Low-Rank Representation. Yan and Wang [22] proposed sparse representation (SR) to construct l_1 -graph [23] by solving l_1 optimization problem. However, l_1 -graph lacks global constraints, which greatly reduce the performance when the data is grossly corrupted. To solve this drawback, Liu et al. proposed the low-rank representation and used it to construct the affinities of an undirected graph (here called LR-graph) [19]. It jointly obtains the representation of all the samples under a global low-rank constraint, and thus it is better at capturing the global data structures [24].

Let $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]$ be a set of samples; each column is a sample which can be represented by a linear combination of the dictionary \mathbf{A} [19]. Here, we select the samples themselves \mathbf{X} as the dictionary \mathbf{A} :

$$\mathbf{X} = \mathbf{A}\mathbf{Z}, \quad (7)$$

where $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_n]$ is the coefficient matrix with each \mathbf{z}_i being the representation coefficient of \mathbf{x}_i . Different from the SR which may not capture the global structure of the data, LRR seeks the lowest rank solution by solving the following optimization problem [19]:

$$\begin{aligned} & \min_{\mathbf{Z}} \text{rank}(\mathbf{Z}) \\ & \text{s.t. } \mathbf{X} = \mathbf{A}\mathbf{Z}. \end{aligned} \quad (8)$$

The above optimization problem can be relaxed to the following convex optimization [25]:

$$\begin{aligned} & \min_{\mathbf{Z}} \|\mathbf{Z}\|_* \\ & \text{s.t. } \mathbf{X} = \mathbf{A}\mathbf{Z}. \end{aligned} \quad (9)$$

Here $\|\cdot\|_*$ denotes the nuclear norm (or trace norm) [26] of a matrix, that is, the sum of the matrix's singular values. By considering the noise or corruption in our real world applications, a more reasonable objective function is

$$\begin{aligned} & \min_{\mathbf{Z}, \mathbf{E}} \|\mathbf{Z}\|_* + \lambda \|\mathbf{E}\|_l \\ & \text{s.t. } \mathbf{X} = \mathbf{A}\mathbf{Z} + \mathbf{E}, \end{aligned} \quad (10)$$

where $\|\cdot\|_l$ can be $l_{2,1}$ -norm or l_1 -norm. In this paper we choose $l_{2,1}$ -norm as the error term which is defined as $\|\mathbf{E}\|_{2,1} = \sum_{j=1}^n \sqrt{\sum_{i=1}^n ([\mathbf{E}]_{ij})^2}$. The parameter λ is used to balance the effect of low rank and the error term. The optimal solution \mathbf{Z}^* can be obtained via the inexact augmented Lagrange multipliers method [27, 28].

3.2. Kernel SDA. Semisupervised Discriminant Analysis may fail to discover the intrinsic geometry structure when the data

manifold is highly nonlinear. The kernel trick is a popular technique in machine learning which uses a kernel function to map samples to a high dimensional space [8, 29, 30]. By using the kernel trick, we can nonlinearly map the original data to the kernel feature space.

Let $Z, Z : R^m \rightarrow F$ be a nonlinear mapping from R^m into F feature space. For any two points \mathbf{x}_i and \mathbf{x}_j , we use a kernel function $K(\mathbf{x}_i, \mathbf{x}_j) = \langle \Phi(\mathbf{x}_i), \Phi(\mathbf{x}_j) \rangle$ to map the data into a kernel feature space. Some commonly used kernels are including the Gaussian radial basis function (RBF) kernel $K(\mathbf{x}, \mathbf{y}) = \exp(-\|\mathbf{x} - \mathbf{y}\|^2/\sigma^2)$, polynomial kernel $K(\mathbf{x}, \mathbf{y}) = (c + \langle \mathbf{x}, \mathbf{y} \rangle)^d$, and sigmoid kernel $K(\mathbf{x}, \mathbf{y}) = \tanh(\langle \mathbf{x}, \mathbf{y} \rangle + \alpha)$ [2, 31].

Let ϕ denote the data matrix in the kernel space: $\Phi = [\phi(\mathbf{x}_1)\phi(\mathbf{x}_2), \dots, \phi(\mathbf{x}_m)]$. The projective vectors $\alpha_1, \alpha_2, \dots, \alpha_c$ are the eigenvector problem in (6) and then we get $m \times c$ transformation matrix $\Theta = [\alpha_1, \alpha_2, \dots, \alpha_c]$. The number of the feature dimensions c can be decided by us. Then a data point can be embedded into c dimensional feature space by

$$\mathbf{x} \longrightarrow \mathbf{y} = \Theta^T K(:, \mathbf{x}), \quad (11)$$

where $K(:, \mathbf{x}) = [K(\mathbf{x}_1, \mathbf{x}), \dots, K(\mathbf{x}_m, \mathbf{x})]^T$.

Kernel SDA (KSVD) [2, 7] can discover the underlying subspace more exactly in the feature space. It results in a better subspace for the classification task by a nonlinear learning technique.

3.3. Low-Rank Kernel-Based SDA. The major problem of all these kernel methods is to find the proper kernel parameters. And they usually use fixed global parameters to determinate the kernel matrix, which is very sensitive to the parameters setting. In fact, the most proper kernel parameters may vary greatly at different random distribution even if they are for the same data. Moreover, the traditional kernel mapping always analyzes the relationship of the data using the mode one-to-others, which emphasizes local information and lacks global constraints on their solutions. These shortcomings limit the performance and efficiency of KSVD methods. To overcome these shortcomings mentioned above, inspired by low-rank representation, we propose a novel kernel-based Semisupervised Discriminant Analysis (LRKSVD) where LRR is used as the kernel representation.

Let $Z, Z : R^m \rightarrow F$ be a low-rank mapping from R^m into a low-rank kernel feature space F . For the database $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_n]$, a reasonable objective function is as follows:

$$\begin{aligned} & \min_{\mathbf{Z}, \mathbf{E}} \|\mathbf{Z}\|_* + \lambda \|\mathbf{E}\|_{2,1} \\ & \text{s.t. } \mathbf{X} = \mathbf{A}\mathbf{Z} + \mathbf{E}. \end{aligned} \quad (12)$$

The optimal solution $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_n]$ is the coefficient matrix with each \mathbf{z}_i being the low-rank representation coefficient of \mathbf{x}_i .

Let $\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_n]$ denote the data matrix in the kernel space. The projective vectors $\alpha_1, \alpha_2, \dots, \alpha_c$ are the eigenvector problem in (6) and $m \times c$ transformation matrix is $\Theta = [\alpha_1, \alpha_2, \dots, \alpha_c]$. The number of the feature dimensions

c can be decided by us. Then a data point can be embedded into c dimensional feature space by

$$\mathbf{x} \longrightarrow \mathbf{y} = \Theta^T \mathbf{z}, \quad (13)$$

where \mathbf{z} is the low-rank representation of \mathbf{x} .

Since the low-rank representation jointly obtains the representation of all the samples under a global low-rank constraint to capture the global data structures, we can get the lowest rank representation in a parameter-free way, which is very convenient and robust for kinds of data. So low-rank kernel-based SDA algorithm can improve the performance to a very large extent. The step of the LRKSDA is as follows.

Firstly, map the labeled and unlabeled data to the LRR-graph kernel space. Secondly, execute the SDA algorithm for dimensionality reduction. Finally execute the nearest neighbor method for the final classification in the derived low-dimensional feature subspace. The procedure of low-rank kernel-based SDA is described as follows.

Algorithm 1 (low-rank kernel-based SDA algorithm). *Input.* The whole data set $[\mathbf{x}_1, \dots, \mathbf{x}_m, \mathbf{x}_{m+1}, \dots, \mathbf{x}_{m+l}]$, where l samples are labeled and m are unlabeled ones.

Output. The classification results.

Step 1. Map the labeled and unlabeled data \mathbf{X} to feature space by the LRR algorithm:

$$\begin{aligned} \min_{\mathbf{Z}, \mathbf{E}} \quad & \|\mathbf{Z}\|_* + \lambda \|\mathbf{E}\|_{2,1} \\ \text{s.t.} \quad & \mathbf{X} = \mathbf{AZ} + \mathbf{E}. \end{aligned} \quad (14)$$

Step 2. Implement the SDA algorithm for dimensionality reduction.

Step 3. Execute the nearest neighbor method for final classification.

4. Experiments and Analysis

In this section, we conduct extensive experiments to examine the efficiency of low-rank kernel-based SDA algorithm. The simulation experiment is conducted in MATLAB7.11.0 (R2010b) environment on a computer with AMD Phenom(tm)II P960 1.79 GHz CPU and 2 GB RAM.

4.1. Experiment Overview

4.1.1. Databases. The proposed LRKSDA is tested on six real world databases, including three face databases and three University of California Irvine (UCI) databases. In these experiments, we normalize the sample to a unit norm.

(1) *Extended Yale Face Database B* [2]. This database has 38 individuals and around 64 near frontal images under different illuminations per individual. Each face image is resized to 32×32 pixels. And we select the first 20 persons and choose 20 samples of each subject.

(2) *ORL Database* [22]. The ORL database contains 10 different images of each for 40 distinct subjects. The images are taken at different times, varying the lighting, facial expressions, and facial details. Each face image is manually cropped and resized to 32×32 pixels, with 256 grey levels per pixel.

(3) *CMU PIE Face Database* [2]. It contains 68 subjects with 41,368 face images. The face images were captured under varying poses, illuminations, and expressions. The size of each image is resized to 32×32 pixels. We select the first 20 persons and choose 20 samples for per subject.

(4) *Musk (Version 2) Data Set 2*. This database contains 2 classes and 6598 instances with 166 features. Here, we randomly select 300 examples for the experiments.

(5) *Seeds Data Set*. It contains 210 instances for three different wheat varieties. A soft X-ray technique and GRAINS package were used to construct all seven, real-valued attributes.

(6) *SPECT Heart Data Set*. The database describes diagnosing of cardiac Single Proton Emission Computed Tomography (SPECT) images. Each of the patients is classified into two categories: normal and abnormal. The database of 267 SPECT image sets was processed to extract features that summarize the original SPECT images. The pattern was further processed to obtain 22 binary feature patterns.

4.1.2. Compared Algorithms. In order to demonstrate how the semisupervised dimensionality reduction performance can be improved by low-rank kernel-based SDA, we list out SDA, KSDA1, and KSDA2 algorithm for comparison. In all experiments, the number of the nearest neighbors in the k NN regularizer graph is set to 4.

(1) *KSDA1 Algorithm.* KSDA1 algorithm is the KSDA with Gaussian radial basis function (RBF) kernel $K(\mathbf{x}, \mathbf{y}) = \exp(-\|\mathbf{x} - \mathbf{y}\|^2 / \sigma^2)$.

(2) *KSDA2 Algorithm.* KSDA2 algorithm is the KSDA which uses polynomial kernel $K(\mathbf{x}, \mathbf{y}) = (c + \langle \mathbf{x}, \mathbf{y} \rangle)^d$. Here, $c = 1$.

The classification accuracy is influenced by the kernel parameters. So after comparing, we choose a proper kernel parameters σ and c for the KSDA1 and KSDA2 algorithm in each database in the following pairs, respectively, where (0.9, 0.9) is for Extended Yale Face Database B, (0.55, 1.5) is for ORL database, (0.9, 0.9) is for CMU PIE database, (0.65, 0.2) is for Musk database, (0.05, 0.6) is for Seeds Data Set, and (0.8, 0.3) is for SPECT Heart Data Set, respectively. Since the most suitable kernel parameters vary greatly at different random distribution even if they are for the same data, these kernel parameters are relatively suitable after comparing by many times' runs.

4.2. Experiment 1: Different Algorithms Performances. To examine the effectiveness of the proposed LRKSDA algorithm, we conduct experiments on the six public databases. In our experiments, we randomly select 30% samples from

TABLE 1: Classification accuracy of different SDA algorithms on six databases.

	Yale B	ORL	PIE	Musk	Seeds	SPECT Heart
LRKSDA	0.825769	0.815	0.578243	0.836667	0.90625	0.778378
KSDA1	0.691392	0.693025	0.541478	0.756849	0.825	0.683333
KSDA2	0.723549	0.681576	0.534542	0.755128	0.709814	0.694154
SDA	0.668397	0.687692	0.527715	0.757407	0.819122	0.69857

TABLE 2: Classification accuracy of different graphs on ORL, Yale, and USPS databases.

Database	Algorithm	The percentage of labeled samples				
		10%	20%	30%	40%	50%
Yale B	LRKSDA	0.711471	0.792667	0.825769	0.848636	0.877778
Yale B	KSDA1	0.316113	0.536868	0.691392	0.807183	0.856101
Yale B	KSDA2	0.33994	0.569484	0.723549	0.819317	0.877637
Yale B	SDA	0.325919	0.560259	0.668397	0.815348	0.855167
ORL	LRKSDA	0.615556	0.728125	0.815	0.873333	0.9
ORL	KSDA1	0.172412	0.448653	0.693167	0.868578	0.937414
ORL	KSDA2	0.172454	0.454899	0.681576	0.851755	0.930487
ORL	SDA	0.173478	0.442731	0.687692	0.877294	0.948035
PIE	LRKSDA	0.29	0.46	0.578243	0.701044	0.82734
PIE	KSDA1	0.195252	0.371027	0.541478	0.711166	0.827522
PIE	KSDA2	0.195345	0.37646	0.543015	0.712033	0.825519
PIE	SDA	0.195707	0.387806	0.527715	0.725264	0.82658

each class as the labeled samples to evaluate the performance with different numbers of selected features. The evaluations are conducted with 20 independent runs for each algorithm. We average them as the final results. First we utilize different kernel methods to get the kernel mapping, and then we implement the SDA algorithm for dimensionality reduction. Finally, the nearest neighbor approach is employed for the final classification in the derived low-dimensional feature subspace. For each database, the classification accuracy for different algorithms is shown in Figure 1. Table 1 shows the performance comparison of different algorithms. Note that the results are the best results of all these different selected features mentioned above. From these results, we can observe the following.

In most cases, our proposed low-rank kernel-based SDA algorithm consistently achieves the highest classification accuracy compared to the other algorithms. LRKSDA achieves the best performance when the dimensionality is larger than a certain low dimension. And the classification accuracy is much higher than the other kernel SDA algorithms. So it improves the classification performance to a large extent, which suggests that low-rank kernel is more informative and suitable for SDA algorithm.

Since the proper kernel parameters are the most important thing of these traditional algorithms and since the kernel parameters of KSDA1 and KSDA2 algorithm are fixed global parameters, the two algorithms are very sensitive to different data or different random distribution of the same data. The performance improvement of these KSDA methods is not obvious. More seriously, as a result of randomly select labeled samples, the random distribution in each run may not adapt

the so-called proper kernel parameters of KSDA1 and KSDA2 algorithm. Moreover, the traditional kernel mapping always analyzes the relationship of the data using the mode one-to-others, which emphasizes local information and lacks global constraints on their solutions. This situation may result in not good performance in some case, while the low-rank representation is better at capturing the global data structures. And we can get the lowest rank representation in a parameter-free way, which is very convenient and robust for kinds of data. So low-rank kernel-based SDA separates the different classes very well compared to other kernel SDA. And it can improve the performance to a very large extent, which means that our proposed low-rank kernel method is extremely effective.

4.3. Experiment 2: Influence of the Label Number. We evaluate the influence of the label number in this part. The experiments are conducted with 20 independent runs for each algorithm. We average them as the final results. The procedure is the same with experiment 1. For each database, we vary the percentage of labeled samples from 10% to 50% and the recognition accuracy is shown in Tables 2 and 3, from which we observe the following.

In most cases, our proposed low-rank kernel-based SDA algorithm consistently achieves the best results, which is robust to the label percentage variations. While some other compared algorithms are not as robust as our LRKSDA algorithm, we can see that the classification accuracy is very awful when the label rate is low. Thus, our proposed method has much superiority than the traditional KSDA and SDA algorithms. Sometimes these traditional methods

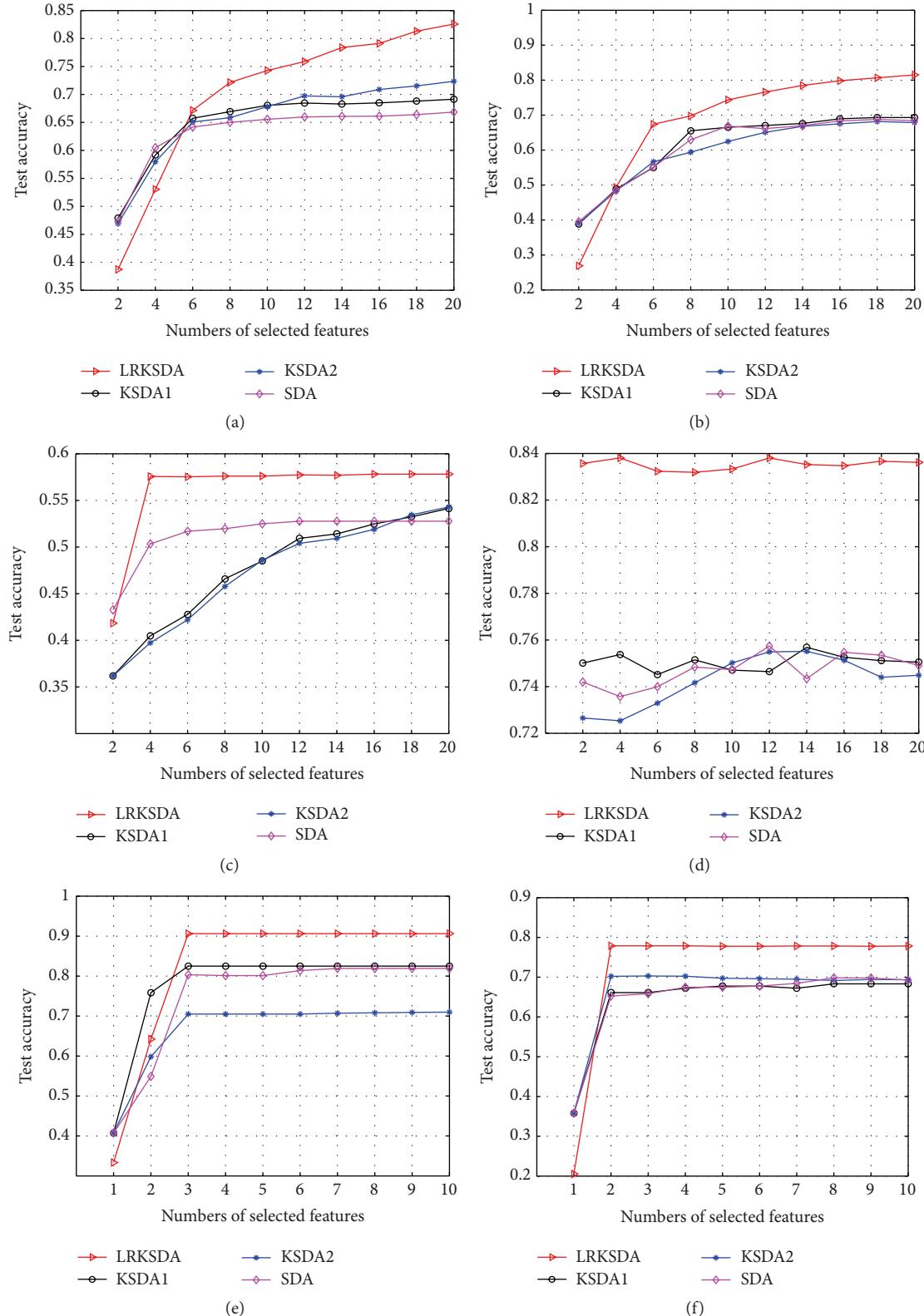


FIGURE 1: Classification accuracy of different SDA algorithms on the six databases of (a) Extended Yale Face Database B, (b) ORL database, (c) CMU PIE face database, (d) Musk (Version 2) Data Set 2, (e) Seeds Data Set, and (f) SPECT Heart Data Set.

TABLE 3: Classification accuracy of different graphs on Musk, Seeds, and SPECT Heart databases.

Database	Algorithm	The percentage of labeled samples				
		10%	20%	30%	40%	50%
Musk	LRKSDA	0.767778	0.827083	0.838095	0.838889	0.895125
Musk	KSDA1	0.356299	0.592578	0.756849	0.83253	0.883174
Musk	KSDA2	0.418741	0.607271	0.755128	0.817146	0.888439
Musk	SDA	0.352444	0.611676	0.757407	0.840006	0.894315
Seeds	LRKSDA	0.890323	0.893333	0.90625	0.90813	0.929608
Seeds	KSDA1	0.46676	0.654862	0.825	0.874946	0.914757
Seeds	KSDA2	0.410025	0.609322	0.709814	0.845034	0.890559
Seeds	SDA	0.503879	0.725435	0.819122	0.872932	0.929595
SPECT Heart	LRKSDA	0.778992	0.778378	0.776216	0.78038	0.826076
SPECT Heart	KSDA1	0.404513	0.622916	0.683333	0.786381	0.85786
SPECT Heart	KSDA2	0.398401	0.608538	0.702989	0.77983	0.869786
SPECT Heart	SDA	0.364647	0.556669	0.69857	0.759696	0.818995

TABLE 4: Classification accuracy of different graphs with varying noise on Yale B database.

Noise types	Algorithm	Variance or density of the three noises					
		0	0.02	0.04	0.06	0.08	0.1
Gaussian	LRKSDA	0.825769	0.816429	0.814286	0.807857	0.808214	0.807143
Gaussian	KSDA1	0.691392	0.555408	0.565422	0.562556	0.574249	0.579816
Gaussian	KSDA2	0.723549	0.585366	0.597015	0.602456	0.590576	0.59866
Gaussian	SDA	0.668397	0.543879	0.540266	0.542199	0.541947	0.543264
“Salt and pepper”	LRKSDA	0.825769	0.794643	0.7675	0.711786	0.643929	0.599286
“Salt and pepper”	KSDA1	0.691392	0.56888	0.509246	0.474557	0.450803	0.436003
“Salt and pepper”	KSDA2	0.723549	0.59498	0.522505	0.478096	0.446308	0.43581
“Salt and pepper”	SDA	0.668397	0.553305	0.498777	0.468533	0.452681	0.429647
Multiplicative	LRKSDA	0.825769	0.825357	0.821429	0.82	0.814286	0.793929
Multiplicative	KSDA1	0.691392	0.631297	0.619849	0.594597	0.584588	0.576168
Multiplicative	KSDA2	0.723549	0.641188	0.622062	0.616446	0.594529	0.594516
Multiplicative	SDA	0.668397	0.594035	0.588897	0.58513	0.582225	0.556328

may achieve good performances in some databases with high enough label rate. But they are not as stable as our proposed algorithm. Since the labeled data is very expensive and difficult, our proposed algorithm is much robust and suitable to the real word data.

As we mentioned in the previous part, since the low-rank kernel method gets the kernel matrix in a parameter-free way, it is robust for different kinds of data, while for the traditional kernel like Gaussian radial basis function kernel and polynomial kernel, if the data’s structure does not fit the stable kernel parameters they used, they cannot obtain the good representation of the original data set. Therefore, the low-rank kernel method is much more stable for all the data sets we use. And the low-rank representation jointly obtains the representation of all the samples under a global low-rank constraint, which can capture the global data structures. So it is robust to the label percentage variations even though the label rate is low.

4.4. Experiment 3: Robustness to Different Types Noises. In this test we compare the performance of different algorithms in the noisy environment. Extended Yale Face Database B

and Musk database are randomly selected in this experiment. The Gaussian white noise, “salt and pepper” noise, and multiplicative noise are added to the data, respectively. The Gaussian white noise is with mean 0 and different variances from 0 to 0.1. The “salt and pepper” noise is added to the image with different noise densities from 0 to 0.1. And multiplicative noise is added to the data I , using the equation $J = I + n * I$, where I and J are the original and noised data and n is uniformly distributed random noise with mean 0 and varying variance from 0 to 0.1. The number of labeled samples in each class is 30%. The experiments are conducted with 20 runs for each algorithm. We average them as the final results. The procedure is the same with experiment 1. For each graph, we vary the parameter of different noise. The results are shown in Tables 4 and 5.

As we can see, our proposed low-rank kernel-based SDA algorithm always achieves the best results, which means that our method is stable for Gaussian noise, “salt and pepper” noise, and multiplicative noise. And because of the robustness of the low-rank representation to noise, our method LRKSDA is much more robust than other algorithms. With the different kinds of gradually increasing noise, the

TABLE 5: Classification accuracy of different graphs with varying noise on Musk database.

Noise types	Algorithm	Variance or density of the three noises					
		0	0.02	0.04	0.06	0.08	0.1
Gaussian	LRKSDA	0.838095	0.783333	0.810476	0.795238	0.789524	0.777619
Gaussian	KSDA1	0.756849	0.689112	0.705138	0.702206	0.699312	0.710083
Gaussian	KSDA2	0.755128	0.705054	0.695936	0.697523	0.695125	0.70306
Gaussian	SDA	0.757407	0.713289	0.699202	0.714286	0.676558	0.681785
“Salt and pepper”	LRKSDA	0.838095	0.785238	0.771429	0.772143	0.766429	0.761905
“Salt and pepper”	KSDA1	0.756849	0.683009	0.667079	0.656237	0.66388	0.653854
“Salt and pepper”	KSDA2	0.755128	0.705003	0.664427	0.658723	0.656934	0.652174
“Salt and pepper”	SDA	0.757407	0.70503	0.697131	0.681818	0.678207	0.666734
Multiplicative	LRKSDA	0.838095	0.832381	0.827143	0.809524	0.793333	0.784286
Multiplicative	KSDA1	0.756849	0.733777	0.723228	0.71144	0.716774	0.71115
Multiplicative	KSDA2	0.755128	0.737889	0.716812	0.710216	0.701506	0.68323
Multiplicative	SDA	0.757407	0.749432	0.738486	0.726044	0.703764	0.68799

traditional KSDA and SDA algorithms' performance falls a lot, while our method's performance is robust to these three noises and drops a few.

Notice that the noise is from a different model other than the original data's subspaces. LRR can well solve the low-rank representation problem. When the data corrupted by arbitrary errors, LRR can also approximately recover the original data with theoretical guarantees. In other words, LRR is robust in an efficient way. Therefore, our method is much more robust than other algorithms with the three noises mentioned above.

5. Conclusions

In this paper, we propose a novel low-rank kernel-based SDA (LRKSDA) algorithm, which largely improves the performance of KSDA and SDA. Since low-rank representation is better at capturing the global data structures, LRKSDA algorithm separates the different classes very well compared to other kernel SDA. Therefore, our proposed low-rank kernel method is extremely effective. Empirical studies on six real world databases show that our proposed low-rank kernel-based SDA is much robust and suitable to the real word applications.

Disclosure

Current affiliation for Baokai Zu is Computer Science Department, Worcester Polytechnic Institute, Worcester, MA 01609, USA.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (no. 51208168), Tianjin Natural Science

Foundation (no. 13JCYBJC37700), Hebei Province Natural Science Foundation (no. E2016202341), Hebei Province Natural Science Foundation (no. F2013202254 and no. F2013202102), and Hebei Province Foundation for Returned Scholars (no. C2012003038).

References

- [1] D. Zhang, Z. H. Zhou, and S. Chen, *Semi-Supervised Dimensionality Reduction*, SDM, Minneapolis, Minn, USA, 2007.
- [2] D. Cai, X. He, and J. Han, “Semi-supervised discriminant analysis,” in *Proceedings of the IEEE 11th International Conference on Computer Vision (ICCV '07)*, pp. 1–7, Rio de Janeiro, Brazil, October 2007.
- [3] Y. Zhang and D.-Y. Yeung, “Semi-supervised discriminant analysis using robust path-based similarity,” in *Proceedings of the 26th IEEE Conference on Computer Vision and Pattern Recognition (CVPR '08)*, pp. 1–8, Anchorage, Alaska, USA, June 2008.
- [4] M. Sugiyama, T. Idé, S. Nakajima, and J. Sese, “Semi-supervised local Fisher discriminant analysis for dimensionality reduction,” *Machine Learning*, vol. 78, no. 1-2, pp. 35–61, 2010.
- [5] Y. Song, F. Nie, C. Zhang, and S. Xiang, “A unified framework for semi-supervised dimensionality reduction,” *Pattern Recognition*, vol. 41, no. 9, pp. 2789–2799, 2008.
- [6] Y. Bengio, J. F. Paiement, P. Vincent et al., “Out-of-sample extensions for LLE, isomap, MDS, eigenmaps, and spectral clustering,” in *Advances in Neural Information Processing Systems 16*, pp. 177–184, MIT Press, 2004.
- [7] D. You, O. C. Hamsici, and A. M. Martinez, “Kernel optimization in discriminant analysis,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 3, pp. 631–638, 2011.
- [8] K.-R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf, “An introduction to kernel-based learning algorithms,” *IEEE Transactions on Neural Networks*, vol. 12, no. 2, pp. 181–201, 2001.
- [9] W.-J. Zeng, X.-L. Li, X.-D. Zhang, and E. Cheng, “Kernel-based nonlinear discriminant analysis using minimum squared errors criterion for multiclass and undersampled problems,” *Signal Processing*, vol. 90, no. 8, pp. 2333–2343, 2010.

- [10] J. Liu, F. Zhao, and Y. Liu, "Learning kernel parameters for kernel Fisher discriminant analysis," *Pattern Recognition Letters*, vol. 34, no. 9, pp. 1026–1031, 2013.
- [11] Z. Wang and X. Sun, "Multiple kernel local Fisher discriminant analysis for face recognition," *Signal Processing*, vol. 93, no. 6, pp. 1496–1509, 2013.
- [12] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: a general framework for dimensionality reduction," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 1, pp. 40–51, 2007.
- [13] K. Q. Weinberger, B. D. Packer, and L. K. Saul, "Nonlinear dimensionality reduction by semidefinite programming and kernel matrix factorization," in *Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics (AISTATS '05)*, pp. 381–388, January 2005.
- [14] S. Yang, X. Wang, M. Wang, Y. Han, and L. Jiao, "Semi-supervised low-rank representation graph for pattern recognition," *IET Image Processing*, vol. 7, no. 2, pp. 131–136, 2013.
- [15] C.-F. Chen, C.-P. Wei, and Y.-C. F. Wang, "Low-rank matrix recovery with structural incoherence for robust face recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*, pp. 2618–2625, IEEE, Providence, RI, USA, June 2012.
- [16] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 171–184, 2013.
- [17] G. Liu and S. Yan, "Latent low-rank representation for subspace segmentation and feature extraction," in *Proceedings of the IEEE International Conference on Computer Vision (IC-CV '11)*, pp. 1615–1622, IEEE, Barcelona, Spain, November 2011.
- [18] X. Lu, Y. Wang, and Y. Yuan, "Graph-regularized low-rank representation for destriping of hyperspectral images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 51, no. 7, pp. 4009–4018, 2013.
- [19] G. Liu, Z. Lin, and Y. Yu, "Robust subspace segmentation by low-rank representation," in *Proceedings of the 27th International Conference on Machine Learning (ICML '10)*, pp. 663–670, Haifa, Israel, June 2010.
- [20] D. Zhou, O. Bousquet, N. T. La et al., "Learning with local and global consistency," *Advances in Neural Information Processing Systems*, vol. 16, no. 16, pp. 321–328, 2004.
- [21] D. M. Cvetkovic and P. Rowlinson, "Spectral graph theory," in *Topics in Algebraic Graph Theory*, pp. 88–112, Cambridge University Press, 2004.
- [22] S. Yan and H. Wang, "Semi-supervised learning by sparse representation," in *Proceedings of the 2009 SIAM International Conference on Data Mining*, pp. 792–801, SDM, 2009.
- [23] J. Wright, A. Y. Yang, A. Ganesh, S. S. Sastry, and Y. Ma, "Robust face recognition via sparse representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 2, pp. 210–227, 2009.
- [24] C. Cortes and M. Mohri, "On transductive regression," in *Advances in Neural Information Processing Systems 19*, pp. 305–312, 2007.
- [25] E. J. Candès, X. Li, Y. Ma, and J. Wright, "Robust principal component analysis?" *Journal of the ACM*, vol. 58, no. 3, article 11, 2011.
- [26] J.-F. Cai, E. J. Candès, and Z. Shen, "A singular value thresholding algorithm for matrix completion," *SIAM Journal on Optimization*, vol. 20, no. 4, pp. 1956–1982, 2010.
- [27] Z. Lin, M. Chen, and Y. Ma, "The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices," <https://arxiv.org/abs/1009.5055>.
- [28] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 1, pp. 171–184, 2013.
- [29] S. Yang, Z. Feng, Y. Ren, H. Liu, and L. Jiao, "Semi-supervised classification via kernel low-rank representation graph," *Knowledge-Based Systems*, vol. 69, no. 1, pp. 150–158, 2014.
- [30] V. N. Vapnik and V. Vapnik, *Statistical Learning Theory*, John Wiley & Sons, New York, NY, USA, 1998.
- [31] H. Nguyen, W. Yang, F. Shen, and C. Sun, "Kernel Low-Rank Representation for face recognition," *Neurocomputing*, vol. 155, pp. 32–42, 2015.

