Pattern Recognition 42 (2009) 2615–2627

Contents lists available at ScienceDirect

Pattern Recognition

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

Semi-supervised orthogonal discriminant analysis via label propagation

Feiping Nie[a,*], Shiming Xiang[b], Yangqing Jia[a], Changshui Zhang[a]

[a]State Key Laboratory of Intelligent Technology and Systems, Tsinghua National Laboratory for Information Science and Technology, Department of Automation, Tsinghua University, Beijing 100084, China

[b]National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China

ARTICLE INFO

Article history:
Received 3 October 2008
Received in revised form 21 January 2009
Accepted 3 April 2009

Keywords:
Subspace learning
Discriminant analysis
Dimensionality reduction
Trace ratio
Semi-supervised learning

ABSTRACT

Trace ratio is a natural criterion in discriminant analysis as it directly connects to the Euclidean distances between training data points. This criterion is re-analyzed in this paper and a fast algorithm is developed to find the global optimum for the orthogonal constrained trace ratio problem. Based on this problem, we propose a novel semi-supervised orthogonal discriminant analysis via label propagation. Differing from the existing semi-supervised dimensionality reduction algorithms, our algorithm propagates the label information from the labeled data to the unlabeled data through a specially designed label propagation, and thus the distribution of the unlabeled data can be explored more effectively to learn a better subspace. Extensive experiments on toy examples and real-world applications verify the effectiveness of our algorithm, and demonstrate much improvement over the state-of-the-art algorithms.

1. Introduction

In many real-world applications in data mining, information retrieval and pattern recognition, labeled data are usually very insufficient and labeling a huge number of data points needs expensive human labor and takes much time. On the other hand, unlabeled data may be abundant and can be easily and cheaply obtained. Thus how to use the labeled and unlabeled data to improve the performance becomes an important problem. This motivation opens a hot research direction of semi-supervised learning [1–8].

Recently, semi-supervised dimensionality reduction has attracted great interest [9–14]. One of the advantages of semi-supervised dimensionality reduction is that it can be directly applied in the whole input space. Therefore, semi-supervised dimensionality reduction is an inductive method and the out-of-sample problem [15] is naturally solved, which makes it more applicable in practice.

Most dimensionality reduction methods fall into the graph embedding framework [16]. Under this framework, current semi-supervised dimensionality reduction algorithms [9–12] construct the weight matrix on graph with the labeled and unlabeled data. However, these algorithms only use the labeled and unlabeled data in a simple manner to construct the weight matrix on graph, which may not sufficiently explore the distribution of the unlabeled data.

In this paper, we propose a novel semi-supervised dimensionality reduction algorithm via a specially designed label propagation procedure. Through the label propagation, the label information in the labeled data is propagated to the unlabeled data according to the distribution of the labeled and unlabeled data, thus the distribution of the unlabeled data can be effectively explored to learn a better subspace. The scatter matrices based on soft label learned by label propagation are defined to perform the discriminant analysis, which gives us a general framework to extend many variants of supervised discriminant analysis to the semi-supervised ones. As the orthogonal projection is of desirable property and often demonstrates good performance empirically, in this paper, we focus on a trace ratio based orthogonal discriminant analysis (ODA), and propose the semi-supervised orthogonal discriminant analysis (SODA) algorithm for dimensionality reduction.

The trace ratio based orthogonal discriminant analysis is an orthogonal variant of linear discriminant analysis (LDA). The projection matrix in LDA is not orthogonal, as pointed out in [17], nonorthogonal projection matrix essentially puts different weights on different projection directions, while orthogonal matrix will not change the similarity if it is based on the Euclidean distance. Recently, orthogonality has attracted great attention in many learning problems [18–24], and there are also several algorithms to extract the orthogonal projection matrix for LDA.

A step-by-step procedure was proposed in [25] to obtain a set of orthogonal projections $\{\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_m\}$. After calculating the first $k-1$ projections $\{\boldsymbol{w}_1, \boldsymbol{w}_2, \ldots, \boldsymbol{w}_{k-1}\}$, the k-th projection $\boldsymbol{w}_k$ is

* Corresponding author. Tel.: +86 10 627 96 872; fax: +86 10 627 86 911.
E-mail addresses: feipingnie@gmail.com (F. Nie), zcs@mail.tsinghua.edu.cn (C. Zhang).

0031-3203/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.patcog.2009.04.001

calculated by solving the following optimization problem:

$$\boldsymbol{w}_k = \arg\min_{\mathbf{W}_{k-1}^T\boldsymbol{w}_k=\mathbf{0}} \frac{\boldsymbol{w}_k^T\mathbf{S}_b\boldsymbol{w}_k}{\boldsymbol{w}_k^T\mathbf{S}_w\boldsymbol{w}_k}, \tag{1}$$

where $\mathbf{W}_{k-1}=[\boldsymbol{w}_1,\boldsymbol{w}_2,\ldots,\boldsymbol{w}_{k-1}]$, $\mathbf{S}_w$ and $\mathbf{S}_b$ are the scatter matrices defined in LDA. The step-by-step procedure makes the algorithm computationally more expensive. Moreover, the optimization objective w.r.t. the whole projections $\{\boldsymbol{w}_1,\boldsymbol{w}_2,\ldots,\boldsymbol{w}_m\}$ is unclear in this procedure.

Recently, [26] proposed another orthogonal LDA (OLDA) algorithm, which is to solve the ratio trace optimization problem as follows:

$$\mathbf{W}_{\mathrm{OLDA}} = \arg\max_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} \mathrm{tr}((\mathbf{W}^T\mathbf{S}_t\mathbf{W})^+\mathbf{W}^T\mathbf{S}_b\mathbf{W}), \tag{2}$$

where $\mathbf{S}_t=\mathbf{S}_w+\mathbf{S}_b$ is the total scatter matrix, $\mathbf{I}$ denotes the identity matrix, $(\cdot)^+$ denotes the Moore–Penrose generalized inverse of a matrix [27] and $\mathrm{tr}(\cdot)$ denotes the trace of a matrix. The computation burden of the orthogonal LDA algorithm is a little alleviated compared to the above one, but the ratio trace criterion used in it is only an approximation to the trace ratio criterion [17,28].

Directly optimizing the trace ratio problem under the orthogonal constraint is a little difficult and time consuming. Recently, [17] proposed a fast algorithm to solve this problem using an iterative procedure. In this paper, we further analyze the trace ratio problem, and reveal that the solution to the trace ratio problem under the uncorrelated constraint is exactly the same as the solution to LDA when $\mathbf{S}_w$ is nonsingular. For the trace ratio problem under the orthogonal constraint, we propose a faster algorithm than the previous ones [17,28] to find the global optimum. Combining the scatter matrices based on soft label learned by label propagation and the trace ratio problem under the orthogonal constraint, we propose the semi-supervised orthogonal discriminant analysis algorithm for dimensionality reduction.

The proposed algorithm is linear, which can be easily extended to the nonlinear one by the kernel trick [29] to better fit the linear inseparable but nonlinear separable data. Experimental results on toy and real-world datasets verify the effectiveness of the proposed algorithms, and demonstrate much improvement over the state-of-the-art algorithms.

The rest of this paper is organized as follows: we analyze the trace ratio problem and introduce the orthogonal discriminant analysis with a faster algorithm in Section 2. In Section 3, we introduce the new label propagation and the soft label based scatter matrices, and propose the semi-supervised orthogonal discriminant analysis algorithm. In Section 4, we extend the proposed algorithm to the nonlinear one by the kernel trick. Extensive experiments are presented in Section 5 and conclusions are drawn in Section 6.

2. Trace ratio based orthogonal discriminant analysis

2.1. Linear discriminant analysis

The goal of LDA is to learn a linear transformation matrix $\mathbf{W}\in\mathbb{R}^{d\times m}$ $(m<d)$, and the original high-dimensional data $\boldsymbol{x}\in\mathbb{R}^d$ is transformed into a low-dimensional vector $\boldsymbol{y}\in\mathbb{R}^m$ by

$$\boldsymbol{y}=\mathbf{W}^T\boldsymbol{x}. \tag{3}$$

Given the training dataset $\mathscr{X}=\{\boldsymbol{x}_i\in\mathbb{R}^d|i=1,\ldots,n\}$, each data $\boldsymbol{x}_i$ is associated with a class label c_i from $\{1,2,\ldots,c\}$. Denote by $\mathscr{X}_i$ the dataset of class i and denote by n_i the number of data points in class i. LDA defines the within-class scatter matrix $\mathbf{S}_w$, the between-class scatter matrix $\mathbf{S}_b$ and the total-class scatter matrix $\mathbf{S}_t$ as follows:

$$\mathbf{S}_w=\sum_{i=1}^{c}\sum_{\boldsymbol{x}\in\mathscr{X}_i}(\boldsymbol{x}-\bar{\boldsymbol{x}}_i)(\boldsymbol{x}-\bar{\boldsymbol{x}}_i)^T, \tag{4}$$

$$\mathbf{S}_b=\sum_{i=1}^{c}n_i(\bar{\boldsymbol{x}}_i-\bar{\boldsymbol{x}})(\bar{\boldsymbol{x}}_i-\bar{\boldsymbol{x}})^T, \tag{5}$$

$$\mathbf{S}_t=\sum_{\boldsymbol{x}\in\mathscr{X}}(\boldsymbol{x}-\bar{\boldsymbol{x}})(\boldsymbol{x}-\bar{\boldsymbol{x}})^T, \tag{6}$$

where $\bar{\boldsymbol{x}}_i=(1/n_i)\sum_{\boldsymbol{x}_j\in\mathscr{X}_i}\boldsymbol{x}_j$ is the mean of the samples in class i and $\bar{\boldsymbol{x}}=(1/n)\sum_{i=1}^{n}\boldsymbol{x}_i$ is the mean of all the samples. It can be verified that $\mathbf{S}_t=\mathbf{S}_w+\mathbf{S}_b$.

The purpose of linear discriminant analysis is to simultaneously maximize the between-class scatter and minimize the within-class scatter. To this end, LDA solves the following ratio trace optimization problem:

$$\mathbf{W}_{\mathrm{LDA}}=\arg\max_{\mathbf{W}}\mathrm{tr}((\mathbf{W}^T\mathbf{S}_w\mathbf{W})^{-1}\mathbf{W}^T\mathbf{S}_b\mathbf{W}). \tag{7}$$

In LDA, it is assumed that the matrix $\mathbf{S}_w$ is nonsingular. It is well known that the solution to LDA is reduced to solving the following generalized eigen-decomposition problem:

$$\mathbf{S}_b\mathbf{W}=\mathbf{S}_w\mathbf{W}\mathbf{\Lambda}. \tag{8}$$

Finally, the columns in $\mathbf{W}_{\mathrm{LDA}}$ are formed by the generalized eigenvectors of $\mathbf{S}_b$ and $\mathbf{S}_w$ corresponding to the first m largest eigenvalues.

2.2. Trace ratio criterion

Another reasonable strategy to obtain great discriminative power is maximizing the term $\mathrm{tr}(\mathbf{W}^T\mathbf{S}_b\mathbf{W})$ and at the same time minimizing the term $\mathrm{tr}(\mathbf{W}^T\mathbf{S}_w\mathbf{W})$ since these two terms directly reflect the Euclidean distances between training data points. As has been pointed out in [17], a natural solution to these dual objectives is to pose a trace ratio optimization problem as follows:

$$\mathbf{W}_{\mathrm{TRDA}}=\arg\max_{\mathbf{W}}\frac{\mathrm{tr}(\mathbf{W}^T\mathbf{S}_b\mathbf{W})}{\mathrm{tr}(\mathbf{W}^T\mathbf{S}_w\mathbf{W})}. \tag{9}$$

To avoid trivial solution, the projection matrix $\mathbf{W}$ in (9) should be constrained. There are two usually used constraints, including the uncorrelated constraint [30] and the orthogonal constraint [25]. In the following, we will derive the solution under the uncorrelated constraint or the orthogonal constraint, respectively.

2.2.1. Uncorrelated constraint

Under the uncorrelated constraint, the trace ratio problem (9) becomes

$$\mathbf{W}_{\mathrm{UTRDA}}=\arg\max_{\mathbf{W}^T\mathbf{S}_t\mathbf{W}=\mathbf{\Omega}}\frac{\mathrm{tr}(\mathbf{W}^T\mathbf{S}_b\mathbf{W})}{\mathrm{tr}(\mathbf{W}^T\mathbf{S}_w\mathbf{W})}, \tag{10}$$

where $\mathbf{\Omega}$ is a given constant and diagonal matrix.

Note that $\mathbf{S}_t=\mathbf{S}_w+\mathbf{S}_b$, therefore, problem (10) can be rewritten as

$$\mathbf{W}_{\mathrm{UTRDA}}=\arg\max_{\mathbf{W}^T\mathbf{S}_t\mathbf{W}=\mathbf{\Omega}}\mathrm{tr}(\mathbf{W}^T\mathbf{S}_b\mathbf{W}). \tag{11}$$

The Lagrangian function of problem (11) is

$$\mathrm{tr}(\mathbf{W}^T\mathbf{S}_b\mathbf{W})-\mathrm{tr}(\boldsymbol{\lambda}(\mathbf{W}^T\mathbf{S}_t\mathbf{W}-\mathbf{\Omega})). \tag{12}$$

In order to obtain the optimal solution to problem (11), we should find out an appropriate $\boldsymbol{\lambda}$ and $\mathbf{W}$ such that the constraint $\mathbf{W}^T\mathbf{S}_t\mathbf{W}=\mathbf{\Omega}$ holds and the derivative of Eq. (12) w.r.t. $\mathbf{W}$ is equal to zero. Note that $\boldsymbol{\lambda}$ is a symmetric matrix, suppose the eigen-decomposition of

$\boldsymbol{\lambda}$ is $\boldsymbol{\lambda} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^T$, where $\boldsymbol{\Lambda}$ is the eigenvalue matrix of $\boldsymbol{\lambda}$, and $\mathbf{U}$ is the corresponding eigenvector matrix. By setting the derivative of Eq. (12) w.r.t. $\mathbf{W}$ to zero, we have

$$\mathbf{S}_b\mathbf{W} - \mathbf{S}_t\mathbf{W}\boldsymbol{\lambda} = \mathbf{0} \Rightarrow \mathbf{S}_b\mathbf{W} = \mathbf{S}_t\mathbf{W}\mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^T \Rightarrow \mathbf{S}_t^{-1}\mathbf{S}_b\mathbf{W}\mathbf{U} = \mathbf{W}\mathbf{U}\boldsymbol{\Lambda}. \tag{13}$$

Let $\mathbf{U}$ be an identity matrix, then the Lagrangian coefficient $\boldsymbol{\lambda} = \boldsymbol{\Lambda}$, and Eq. (13) becomes

$$\mathbf{S}_b\mathbf{W} = \mathbf{S}_t\mathbf{W}\boldsymbol{\Lambda}. \tag{14}$$

Note that $\mathbf{S}_t$ and $\mathbf{S}_b$ are symmetric, the $\mathbf{W}$ in Eq. (14) satisfies that $\mathbf{W}^T\mathbf{S}_t\mathbf{W}$ is a diagonal matrix. Therefore, when the Lagrangian coefficient $\boldsymbol{\lambda} = \boldsymbol{\Lambda}$ and $\mathbf{W}$ is formed by the generalized eigenvectors of $\mathbf{S}_b$ and $\mathbf{S}_t$ as in Eq. (14), the constraint $\mathbf{W}^T\mathbf{S}_t\mathbf{W} = \boldsymbol{\Omega}$ will hold and the derivative of Eq. (12) w.r.t. $\mathbf{W}$ is equal to zero. So the solution to problem (10) can be reduced to solving the generalized eigen-decomposition problem in Eq. (14), and the columns in $\mathbf{W}_{\mathrm{UTRDA}}$ are formed by the generalized eigenvectors of $\mathbf{S}_b$ and $\mathbf{S}_t$ corresponding to the first m largest eigenvalues.

It is interesting to see that the solution to the uncorrelated constrained trace ratio problem (10) is exactly the solution to LDA, which give us a new insight into the LDA method.

2.2.2. Orthogonal constraint

Under the orthogonal constraint, the trace ratio problem (9) becomes

$$\mathbf{W}_{\mathrm{OTRDA}} = \arg\max_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} \frac{\mathrm{tr}(\mathbf{W}^T\mathbf{S}_b\mathbf{W})}{\mathrm{tr}(\mathbf{W}^T\mathbf{S}_w\mathbf{W})}, \tag{15}$$

where $\mathbf{I}$ is an identity matrix.

In practice, to avoid overfitting, we can add a regularization term to $\mathbf{S}_w$, and the optimization problem (15) becomes

$$\mathbf{W}_{\mathrm{SODA}} = \arg\max_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} \frac{\mathrm{tr}(\mathbf{W}^T\mathbf{S}_b\mathbf{W})}{\mathrm{tr}(\mathbf{W}^T(\mathbf{S}_w + \mu\mathbf{I}_d)\mathbf{W})}, \tag{16}$$

where $\mathbf{I}_d$ is a $d \times d$ identity matrix, and $\mu > 0$ is the regularization parameter. We call the algorithm that solves the orthogonal constrained trace ratio problem (16) as orthogonal discriminant analysis.

In comparison with problem (10), problem (15) or (16) is more difficult to solve, and the closed solution is hard to derive. Fortunately, there exists efficient algorithm to solve the problem by iterative procedure.

2.3. Efficient algorithm for the orthogonal constrained trace ratio problem

2.3.1. A recently proposed fast algorithm revisited

Recently, an efficient algorithm [17] was proposed to solve the orthogonal constrained trace ratio problem (15). The algorithm is briefly stated as below:

Step 1: Initialize $\mathbf{W}$ as an arbitrary column-orthogonal matrix.

Step 2: Compute the trace ratio value $\lambda = \mathrm{tr}(\mathbf{W}^T\mathbf{S}_b\mathbf{W})/\mathrm{tr}(\mathbf{W}^T\mathbf{S}_w\mathbf{W})$.

Step 3: Construct the trace difference problem as

$$\mathbf{W}^* = \arg\max_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} \mathrm{tr}(\mathbf{W}^T(\mathbf{S}_b - \lambda\mathbf{S}_w)\mathbf{W}). \tag{17}$$

Step 4: It is well known that $\mathbf{W}^*$ is formed by the m eigenvectors of $\mathbf{S}_b - \lambda\mathbf{S}_w$ corresponding to the m largest eigenvalues. Update $\mathbf{W}$ by $\mathbf{W}^*$.

Step 5: Iteratively perform steps 2–4 until convergence.

It was rigorously proved that the algorithm converges to the global optimum [17]. To better understand it, we give a theorem to reveal the connection between the trace difference problem and the trace ratio problem. Denote $\mathbf{W}_\lambda = \arg\max_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} \mathrm{tr}(\mathbf{W}^T(\mathbf{S}_b - \lambda\mathbf{S}_w)\mathbf{W})$, we define a function as below:

$$f(\lambda) = \frac{\mathrm{tr}(\mathbf{W}_\lambda^T\mathbf{S}_b\mathbf{W}_\lambda)}{\mathrm{tr}(\mathbf{W}_\lambda^T\mathbf{S}_w\mathbf{W}_\lambda)}. \tag{18}$$

Suppose that

$$\lambda^* = \max_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} \frac{\mathrm{tr}(\mathbf{W}^T\mathbf{S}_b\mathbf{W})}{\mathrm{tr}(\mathbf{W}^T\mathbf{S}_w\mathbf{W})}, \tag{19}$$

then we have the following theorem:

Theorem 1. *The function $f(\lambda)$ is monotonically increasing when $\lambda \leqslant \lambda^*$, and is monotonically decreasing when $\lambda \geqslant \lambda^*$.*

The proof is given in Appendix A. Theorem 1 indicates that function $f(\lambda)$ has only a single peak, which gives a theoretical confirmation to the experimental observation in [31]. Theorem 1 gives us an insight into the orthogonal constrained trace ratio problem. Although the trace ratio problem is not convex, the function $f(\lambda)$ which is associated with the trace difference problem is exactly convex. Therefore, it is not hard to understand why the algorithm that converts the trace ratio problem to the trace difference problem can find the global optimum.

2.3.2. A faster algorithm

The algorithm stated in Section 2.3.1 is efficient to solve problem (15). Here, we propose a more efficient algorithm to solve it.

The Lagrangian function of problem (15) is

$$\frac{\mathrm{tr}(\mathbf{W}^T\mathbf{S}_b\mathbf{W})}{\mathrm{tr}(\mathbf{W}^T\mathbf{S}_w\mathbf{W})} - \mathrm{tr}(\boldsymbol{\lambda}(\mathbf{W}^T\mathbf{W} - \mathbf{I})). \tag{20}$$

Note that $\boldsymbol{\lambda}$ is a symmetric matrix, suppose the eigen-decomposition of $\boldsymbol{\lambda}$ is $\boldsymbol{\lambda} = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^T$, where $\boldsymbol{\Lambda}$ is the eigenvalue matrix of $\boldsymbol{\lambda}$, and $\mathbf{U}$ is the corresponding eigenvector matrix. By setting the derivative of Eq. (20) w.r.t. $\mathbf{W}$ to zero, we have

$$\begin{aligned} &\frac{\mathrm{tr}(\mathbf{W}^T\mathbf{S}_w\mathbf{W})\mathbf{S}_b\mathbf{W} - \mathrm{tr}(\mathbf{W}^T\mathbf{S}_b\mathbf{W})\mathbf{S}_w\mathbf{W}}{\mathrm{tr}(\mathbf{W}^T\mathbf{S}_w\mathbf{W})^2}\mathbf{S}_b\mathbf{W} - \mathbf{W}\boldsymbol{\lambda} = \mathbf{0} \\ &\Rightarrow \left(\mathbf{S}_b - \frac{\mathrm{tr}(\tilde{\mathbf{W}}^T\mathbf{S}_b\tilde{\mathbf{W}})}{\mathrm{tr}(\tilde{\mathbf{W}}^T\mathbf{S}_w\tilde{\mathbf{W}})}\mathbf{S}_w\right)\tilde{\mathbf{W}} = \tilde{\mathbf{W}}\tilde{\boldsymbol{\Lambda}}, \end{aligned} \tag{21}$$

where $\tilde{\mathbf{W}} = \mathbf{W}\mathbf{U}$ and $\tilde{\boldsymbol{\Lambda}} = \mathrm{tr}(\mathbf{W}^T\mathbf{S}_w\mathbf{W})\boldsymbol{\Lambda}$.

Eq. (21) indicates two facts. First, if $\mathbf{W}$ is the optimal solution to problem (15), then $\tilde{\mathbf{W}} = \mathbf{W}\mathbf{U}$ is also an optimal solution, where $\mathbf{U}$ could be an arbitrary orthogonal matrix. Second, the optimal solution $\tilde{\mathbf{W}}$ should satisfy Eq. (21). Based on these two facts, we know that finding the optimal solution to problem (15) is equivalent to finding a $\tilde{\mathbf{W}}$ such that Eq. (21) is satisfied. We propose an iterative procedure to find such a $\tilde{\mathbf{W}}$ that satisfies Eq. (21). The iterative procedure is stated as follows:

Step 1: Initialize $\mathbf{W}$ as an arbitrary column-orthogonal matrix.

Step 2: Compute the trace ratio value $\lambda = \mathrm{tr}(\mathbf{W}^T\mathbf{S}_b\mathbf{W})/\mathrm{tr}(\mathbf{W}^T\mathbf{S}_w\mathbf{W})$.

Step 3: Compute the eigen-decomposition of $\mathbf{S}_b - \lambda\mathbf{S}_w$ as

$$(\mathbf{S}_b - \lambda\mathbf{S}_w)\boldsymbol{w}_i = \tau_i\boldsymbol{w}_i, \tag{22}$$

where $\boldsymbol{w}_i$ $(i = 1, 2, \ldots d)$ is the eigenvector of $\mathbf{S}_b - \lambda\mathbf{S}_w$.

Step 4: Solve the following problem:

$$\mathbf{W}^* = \arg\max_{\mathbf{W}\in\Phi} \frac{\mathrm{tr}(\mathbf{W}^T\mathbf{S}_b\mathbf{W})}{\mathrm{tr}(\mathbf{W}^T\mathbf{S}_w\mathbf{W})}, \tag{23}$$

where Φ is the set of matrix whose columns are formed by m different eigenvectors selected from $\boldsymbol{w}_i$ $(i = 1, 2, \ldots d)$. Update $\mathbf{W}$ by $\mathbf{W}^*$.

Step 5: Iteratively perform steps 2–4 until convergence.

Comparing this algorithm with the algorithm in [17] (described in Section 2.3.1), we can see that the main difference is in the step 4. That is, the main difference is how to update $\mathbf{W}$ in the next iteration. The algorithm in [17] uses the m eigenvectors of $\mathbf{S}_b - \lambda\mathbf{S}_w$ corresponding to the m largest eigenvalues to update $\mathbf{W}$. However, the solution $\mathbf{W}^*$ to problem (23) is not necessarily formed by these top m eigenvectors. Therefore, the trace ratio value λ in each iteration in the proposed algorithm is usually greater than that of in the algorithm in [17]. In addition, problem (23) can be efficiently solved without performing eigenvalue decomposition and the computation cost can be ignored compared with the eigenvalue decomposition [32]. Therefore, the proposed iterative algorithm is faster than the algorithm in [17]. As it has been proved that the algorithm in [17] can find the global optimum, the proposed algorithm can also find the global optimum.

Now we have introduced the trace ratio based orthogonal discriminant analysis method with a fast algorithm to find the global optimum of the orthogonal constrained trace ratio problem. In the next section, we will propose a general framework to extend the variants of supervised discriminant analysis to the semi-supervised ones. Based on the framework and the introduced orthogonal discriminant analysis, we will propose the semi-supervised orthogonal discriminant analysis algorithm and its kernel version.

3. Semi-supervised orthogonal discriminant analysis via label propagation

3.1. Calculate the soft label through label propagation

Label propagation is a key idea in many graph based semi-supervised learning algorithms [4,5]. It propagates label information from labeled data to unlabeled data according to the distribution of labeled and unlabeled data.

In the label propagation, a neighborhood weighted graph on data should be constructed first. A popular construction method is as follows: if $\boldsymbol{x}_i$ is among the k nearest neighbors of $\boldsymbol{x}_j$ or $\boldsymbol{x}_j$ is among the k nearest neighbors of $\boldsymbol{x}_i$, then $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are linked by a weight computed by

$$\mathbf{A}_{ij} = e^{-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2/\sigma^2}, \tag{24}$$

otherwise, $\mathbf{A}_{ij} = 0$. Here σ is the variance, $\|\cdot\|$ is the 2-norm of vector, i.e., $\|\boldsymbol{x}\|^2 = \boldsymbol{x}^T\boldsymbol{x}$.

In this paper, we introduce an additional class $c+1$ in order to detect outlier data. Let $\mathbf{F} = [\mathbf{F}_1^T, \ldots, \mathbf{F}_n^T]^T \in \mathbb{R}^{n\times(c+1)}$ be the predicted label matrix, where $\mathbf{F}_i \in \mathbb{R}^{c+1}$ $(1 \leqslant i \leqslant n)$ are row vectors and $0 \leqslant \mathbf{F}_{ij} \leqslant 1$. Define the initial label matrix $\mathbf{Y} = [\mathbf{Y}_1^T, \ldots, \mathbf{Y}_n^T]^T \in \mathbb{R}^{n\times(c+1)}$, where $\mathbf{Y}_i \in \mathbb{R}^{c+1}$ $(1 \leqslant i \leqslant n)$ are row vectors. For the labeled data, $\mathbf{Y}_{ij} = 1$ if $\boldsymbol{x}_i$ is labeled as j and $\mathbf{Y}_{ij} = 0$ otherwise. For the unlabeled data $\boldsymbol{x}_i$, $\mathbf{Y}_{ij} = 1$ if $j = c+1$ and $\mathbf{Y}_{ij} = 0$ otherwise.

Denote a stochastic matrix $\mathbf{P} = \mathbf{D}^{-1}\mathbf{A}$, where $\mathbf{D}$ is the diagonal matrix with the i-th diagonal element being $\mathbf{D}_{ii} = \sum_j \mathbf{A}_{ij}$. Let us consider a new iterative process for label propagation. In each iteration, the label information of each data point is partly received from its neighbors, and the rest is received from its initial label (see Fig. 1). The label information of the data at time $t+1$ is propagated based on the following equation:

$$\mathbf{F}(t+1) = \mathbf{I}_\alpha\mathbf{P}\mathbf{F}(t) + \mathbf{I}_\beta\mathbf{Y}, \tag{25}$$

where $\mathbf{I}_\alpha$ is an $n \times n$ diagonal matrix with the i-th entry being α_i, $\mathbf{I}_\beta = \mathbf{I} - \mathbf{I}_\alpha$, α_i $(0 \leqslant \alpha_i < 1)$ is a parameter for data $\boldsymbol{x}_i$ to balance the initial label information of $\boldsymbol{x}_i$ and the label information received from its neighbors during the iteration. In practice, for labeled data $\boldsymbol{x}_i$, α_i could be set to 0, while for unlabeled data $\boldsymbol{x}_i$, α_i could be set to a value near to 1.

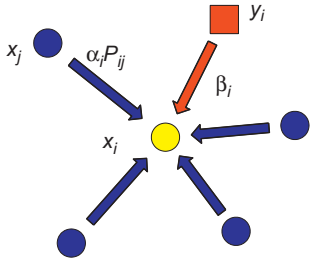

Fig. 1. A new label propagation on graphs. The blue circle data denotes the neighbors of the yellow circle data $\boldsymbol{x}_i$, and the red square denotes the initial label y_i of $\boldsymbol{x}_i$. In each iteration of the label propagation process, the label information of $\boldsymbol{x}_i$ is partly received from its neighbors' labels, and the rest is received from its initial label y_i. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

By the iteration equation (25), we have

$$\mathbf{F}(t) = (\mathbf{I}_\alpha\mathbf{P})^t\mathbf{F}(0) + \sum_{i=0}^{t-1}(\mathbf{I}_\alpha\mathbf{P})^i\mathbf{I}_\beta\mathbf{Y} \tag{26}$$

and the iteration process converges to

$$\mathbf{F} = \lim_{t\to\infty}\mathbf{F}(t) = (\mathbf{I} - \mathbf{I}_\alpha\mathbf{P})^{-1}\mathbf{I}_\beta\mathbf{Y}. \tag{27}$$

It can be easily verified that the sum of each row of $\mathbf{F}$ is equal to 1, which indicates the elements in $\mathbf{F}$ are probability values, and $\mathbf{F}_{ij}$ can be seen as an estimation of the posterior probability of $\boldsymbol{x}_i$ belonging to class j. When $j = c+1$, $\mathbf{F}_{i,c+1}$ denotes the probability of $\boldsymbol{x}_i$ belonging to outlier. As the value of $\mathbf{F}_{ij}$ is probability value, in this paper, we call $\mathbf{F}_{ij}$ $(1 \leqslant j \leqslant c)$ as the *soft label* of $\boldsymbol{x}_i$.

By this specially designed label propagation, the outlier in data can be detected and the soft label of each data is obtained. As can be seen in the next subsection, it is convenient to construct the scatter matrices for discriminant analysis using the soft label.

3.2. Soft label based scatter matrices

After the label propagation, we obtain the soft label for each data $\boldsymbol{x}_i$ $(i = 1, 2, \ldots, n)$, i.e., the probability $\mathbf{F}_{ij}$ of $\boldsymbol{x}_i$ belonging to class j $(j = 1, 2, \ldots, c)$. While in LDA, the scatter matrices $\mathbf{S}_w$, $\mathbf{S}_b$ and $\mathbf{S}_t$ are defined based on the hard label of each data $\boldsymbol{x}_i$ $(i = 1, 2, \ldots, n)$, i.e., the probability $\mathbf{F}_{ij}$ of $\boldsymbol{x}_i$ belonging to class j $(j = 1, 2, \ldots, c)$ is either 0 or 1. Here we extend the scatter matrices defined in LDA to the soft label based scatter matrices, and defined as follows:

$$\tilde{\mathbf{S}}_w = \frac{1}{\tilde{n}}\sum_{i=1}^{c}\sum_{j=1}^{n}\mathbf{F}_{ji}(\boldsymbol{x}_j - \tilde{\boldsymbol{x}}_i)(\boldsymbol{x}_j - \tilde{\boldsymbol{x}}_i)^T = \frac{1}{\tilde{n}}\mathbf{X}(\mathbf{B} - \mathbf{F}_c\mathbf{D}\mathbf{F}_c^T)\mathbf{X}^T, \tag{28}$$

$$\tilde{\mathbf{S}}_b = \sum_{i=1}^{c}\frac{\tilde{n}_i}{\tilde{n}}(\tilde{\boldsymbol{x}}_i - \tilde{\boldsymbol{x}})(\tilde{\boldsymbol{x}}_i - \tilde{\boldsymbol{x}})^T = \frac{1}{\tilde{n}}\mathbf{X}\left(\mathbf{F}_c\mathbf{D}\mathbf{F}_c^T - \frac{1}{\tilde{n}}\mathbf{B}\mathbf{1}\mathbf{1}^T\mathbf{B}\right)\mathbf{X}^T, \tag{29}$$

$$\tilde{\mathbf{S}}_t = \frac{1}{\tilde{n}}\sum_{i=1}^{n}\mathbf{B}_{ii}(\boldsymbol{x}_i - \tilde{\boldsymbol{x}})(\boldsymbol{x}_i - \tilde{\boldsymbol{x}})^T = \frac{1}{\tilde{n}}\mathbf{X}\left(\mathbf{B} - \frac{1}{\tilde{n}}\mathbf{B}\mathbf{1}\mathbf{1}^T\mathbf{B}\right)\mathbf{X}^T, \tag{30}$$

where $\tilde{n}_i = \sum_{j=1}^{n}\mathbf{F}_{ji}$, $\tilde{n} = \sum_{i=1}^{c}\tilde{n}_i$, $\tilde{\boldsymbol{x}}_i = \sum_{j=1}^{n}\mathbf{F}_{ji}\boldsymbol{x}_j/\tilde{n}_i$, $\tilde{\boldsymbol{x}} = \sum_{i=1}^{n}\sum_{j=1}^{c}\mathbf{F}_{ij}\boldsymbol{x}_i/\tilde{n}$, $\mathbf{B} \in \mathbb{R}^{n\times n}$ is a diagonal matrix, the i diagonal element of which is $\mathbf{B}_{ii} = \sum_{j=1}^{c}\mathbf{F}_{ij}$. $\mathbf{D} \in \mathbb{R}^{c\times c}$ is a diagonal matrix, the i diagonal element of which is $\mathbf{D}_{ii} = 1/\sum_{j=1}^{n}\mathbf{F}_{ji}$, $\mathbf{F}_c \in \mathbb{R}^{n\times c}$ is formed by the first c columns of $\mathbf{F}$, $\mathbf{1} = [1, 1, \ldots, 1]^T \in \mathbb{R}^{n\times 1}$, $\mathbf{X} = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n] \in \mathbb{R}^{d\times n}$.

It can be easily checked that $\tilde{\mathbf{S}}_t = \tilde{\mathbf{S}}_w + \tilde{\mathbf{S}}_b$. When the soft label becomes hard label, i.e., $\mathbf{F}_{ij}$ is either 0 or 1, the soft label based scatter matrices $\tilde{\mathbf{S}}_w$, $\tilde{\mathbf{S}}_b$ and $\tilde{\mathbf{S}}_t$ defined here become the scatter matrices defined in LDA, respectively.

Table 1
Semi-supervised orthogonal discriminant analysis algorithm.

Input:
Data matrix $\mathbf{X} \in \mathbb{R}^{d\times n}$ (each column is a data point). Projected dimension m and other related parameters.
Output:
The projection matrix $\mathbf{W} \in \mathbb{R}^{d\times m}$.
Algorithm:
1. Construct the neighborhood graph and calculate the weight matrix **A**.
2. Perform label propagation by Eq. (27) and obtain the soft label matrix **F**.
3. Calculate the soft label based scatter matrices $\tilde{\mathbf{S}}_w$ and $\tilde{\mathbf{S}}_b$ by Eqs. (28) and (29), respectively.
3. Solve the orthogonal constrained trace ratio problem (32) by the algorithm proposed in Section 2.3.2.

Note that $\tilde{\mathbf{S}}_w$ and $\tilde{\mathbf{S}}_b$ are constructed based on the soft labels which are learned from the label propagation between labeled and unlabeled data, thus by using these scatter matrices in a discriminant analysis algorithm, the distribution of the unlabeled data can be effectively explored to learn a better subspace.

3.3. Semi-supervised orthogonal discriminant analysis

Now, we have the scatter matrices based on soft labels learned by label propagation within labeled data and unlabeled data. Applying them to a supervised discriminant analysis will derive a semi-supervised counterpart, which give us a general framework to extend many variants of LDA to the semi-supervised ones. In this paper, we focus on the trace ratio based orthogonal discriminant analysis, and propose the semi-supervised orthogonal discriminant analysis algorithm, which is to solve the following optimization problem:

$$\mathbf{W}_{\mathrm{SODA}} = \arg\max_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} \frac{\mathrm{tr}(\mathbf{W}^T\tilde{\mathbf{S}}_b\mathbf{W})}{\mathrm{tr}(\mathbf{W}^T\tilde{\mathbf{S}}_w\mathbf{W})}, \tag{31}$$

where $\tilde{\mathbf{S}}_w$ and $\tilde{\mathbf{S}}_b$ are the soft label based scatter matrices defined in (28) and (29), respectively.

Similarly to the orthogonal discriminant analysis, to avoid overfitting in practice, we add a regularization term to $\tilde{\mathbf{S}}_w$, and the optimization problem (31) becomes

$$\mathbf{W}_{\mathrm{SODA}} = \arg\max_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} \frac{\mathrm{tr}(\mathbf{W}^T\tilde{\mathbf{S}}_b\mathbf{W})}{\mathrm{tr}(\mathbf{W}^T(\tilde{\mathbf{S}}_w+\mu\mathbf{I}_d)\mathbf{W})}, \tag{32}$$

where $\mathbf{I}_d$ is a $d\times d$ identity matrix and $\mu>0$ is the regularization parameter.

We summarize the proposed semi-supervised orthogonal discriminant analysis algorithm in Table 1. It is worth noting that the step of label propagation in the algorithm only needs to solve sparse linear equations, which has been intensively studied and there exist efficient algorithms whose computational time is nearly linear [33]. Therefore, the computational burden of this step can be ignored in the algorithm.

4. Kernel semi-supervised orthogonal discriminant analysis

The proposed semi-supervised orthogonal discriminant analysis is a linear algorithm. By the kernel trick, it is easy to extend the linear algorithm to a nonlinear one.

Suppose the data are mapped from the original input space to a higher dimensional Hilbert space $\mathscr{F}$ with a nonlinear mapping $\phi: x\to\mathscr{F}$, and the map is implicitly implemented via kernel function $\mathscr{K}(\boldsymbol{x}_i,\boldsymbol{x}_j)=\phi(\boldsymbol{x}_i)\cdot\phi(\boldsymbol{x}_j)$. The kernel function $\mathscr{K}: \mathbb{R}^d\times\mathbb{R}^d\to\mathbb{R}$ may be any positive kernel satisfying Mercer's condition [34,35]. For instance, a frequently used one is the radial basis function (RBF) kernel defined by

$$\mathscr{K}(\boldsymbol{x}_i,\boldsymbol{x}_j)=e^{-\|\boldsymbol{x}_i-\boldsymbol{x}_j\|^2/\sigma^2}. \tag{33}$$

Performing the semi-supervised orthogonal discriminant analysis algorithm in the nonlinearly mapped high-dimensional space $\mathscr{F}$ will derive the kernel semi-supervised orthogonal discriminant analysis (KSODA).

Denote $\bar{\phi}=(1/n)\sum_i\phi(\boldsymbol{x}_i)$, $\phi(\mathbf{X})=[\phi(\boldsymbol{x}_1),\phi(\boldsymbol{x}_2),\ldots,\phi(\boldsymbol{x}_n)]$. Denote a centralization matrix by $\mathbf{L}_c=\mathbf{I}-(1/n)\mathbf{1}\mathbf{1}^T$, where $\mathbf{I}$ is an $n\times n$ identity matrix, and $\mathbf{1}\in\mathbb{R}^n$ is a column vector in which all the elements are equal to one. Note that the solution $\mathbf{W}$ lies in the subspace spanned by the centralized training data $\{\phi(\boldsymbol{x}_1)-\bar{\phi},\phi(\boldsymbol{x}_2)-\bar{\phi},\ldots,\phi(\boldsymbol{x}_n)-\bar{\phi}\}$, we can express $\mathbf{W}$ as $\mathbf{W}=\phi(\mathbf{X})\mathbf{L}_c\boldsymbol{\alpha}$, where $\boldsymbol{\alpha}\in\mathbb{R}^{n\times m}$. Denote the centralized kernel matrix by

$$\mathbf{K}=\mathbf{L}_c\phi(\mathbf{X})^T\phi(\mathbf{X})\mathbf{L}_c, \tag{34}$$

then we have the following optimization problem from (32):

$$\boldsymbol{\alpha}_{\mathrm{KSODA}} = \arg\max_{\boldsymbol{\alpha}^T\mathbf{K}\boldsymbol{\alpha}=\mathbf{I}} \frac{\mathrm{tr}(\boldsymbol{\alpha}^T\mathbf{K}\tilde{\mathbf{L}}_b\mathbf{K}\boldsymbol{\alpha})}{\mathrm{tr}(\boldsymbol{\alpha}^T(\mathbf{K}\tilde{\mathbf{L}}_w\mathbf{K}+\mu\mathbf{K}\boldsymbol{\alpha})}, \tag{35}$$

where $\tilde{\mathbf{L}}_w=\mathbf{B}-\mathbf{F}_c\mathbf{D}\mathbf{F}_c^T$ and $\tilde{\mathbf{L}}_b=\mathbf{F}_c\mathbf{D}\mathbf{F}_c^T-(1/\tilde{n})\mathbf{B}\mathbf{1}\mathbf{1}^T\mathbf{B}$, both of which are Laplacian matrices.

Suppose the eigen-decomposition of $\mathbf{K}$ is

$$\mathbf{K}=\mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^T, \tag{36}$$

where $\boldsymbol{\Lambda}$ is the positive eigenvalue matrix and $\mathbf{U}$ is the corresponding eigenvector matrix. Let $\boldsymbol{\alpha}=\mathbf{U}\boldsymbol{\Lambda}^{-1/2}\boldsymbol{\beta}$, then problem (35) can be rewritten as an orthogonal constrained trace ratio problem:

$$\boldsymbol{\beta}_{\mathrm{KSODA}} = \arg\max_{\boldsymbol{\beta}^T\boldsymbol{\beta}=\mathbf{I}} \frac{\mathrm{tr}(\boldsymbol{\beta}^T\mathbf{H}_b\boldsymbol{\beta})}{\mathrm{tr}(\boldsymbol{\beta}^T(\mathbf{H}_w+\mu\mathbf{I}_r)\boldsymbol{\beta})}, \tag{37}$$

where

$$\mathbf{H}_w=\boldsymbol{\Lambda}^{-1/2}\mathbf{U}^T\mathbf{K}\tilde{\mathbf{L}}_w\mathbf{K}\mathbf{U}\boldsymbol{\Lambda}^{-1/2} \tag{38}$$

and

$$\mathbf{H}_b=\boldsymbol{\Lambda}^{-1/2}\mathbf{U}^T\mathbf{K}\tilde{\mathbf{L}}_b\mathbf{K}\mathbf{U}\boldsymbol{\Lambda}^{-1/2}. \tag{39}$$

Problem (37) is exactly an orthogonal constrained trace ratio problem, and can be efficiently solved by the algorithm proposed in Section 2.3.2. After computing $\boldsymbol{\beta}_{\mathrm{KSODA}}$, we have $\mathbf{W}_{\mathrm{KSODA}}=\phi(\mathbf{X})\mathbf{L}_c\mathbf{U}\boldsymbol{\Lambda}^{-1/2}\boldsymbol{\beta}_{\mathrm{KSODA}}$. For any data point $\boldsymbol{x}\in\mathbb{R}^d$, the projected data point is

$$\boldsymbol{y}=\mathbf{W}_{\mathrm{KSODA}}^T\phi(\boldsymbol{x})=\boldsymbol{\beta}_{\mathrm{KSODA}}^T\boldsymbol{\Lambda}^{-1/2}\mathbf{U}^T\mathbf{L}_c\phi(\mathbf{X})^T\phi(\boldsymbol{x}). \tag{40}$$

5. Experiments

In this section, we evaluate the proposed algorithms with toy examples and several real-world applications, and compared them with several representative algorithms. The algorithms performed in the experiments are as follows: linear discriminant analysis, orthogonal linear discriminant analysis [26], orthogonal discriminant analysis (to solve problem (16)), semi-supervised discriminant

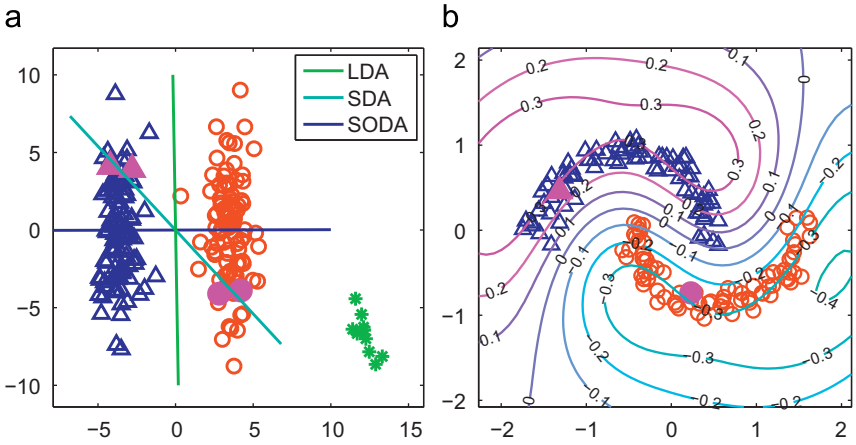

Fig. 2. (a) The projection direction learned by LDA, SDA and SODA; Gaussian. The blue star points are the outlier data detected by SODA. (b) The contour lines learned by KSODA; two moon. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

analysis (SDA) [10], semi-supervised orthogonal discriminant analysis (to solve problem (32)) and kernel semi-supervised orthogonal discriminant analysis (to solve problem (35)).

5.1. Toy examples

We present two toy examples to demonstrate the effectiveness of our algorithms.

In the first toy example, we generate a dataset with two classes, each of which is distributed with a Gaussian. We also add some outlier data in the dataset. Two data points of each class are labeled (shown as the purple points). Fig. 2(a) shows the learned projection directions by LDA, SDA and SODA, respectively. The experimental result indicates that SDA can improve LDA by use of unlabeled data. However, SDA only use the labeled and unlabeled data in a simple manner to construct the weight matrix on graph, and the distribution in data might not be explored thoroughly. In contrast, SODA use label propagation to learn the soft labels for the unlabeled data, thus the labeled information and the distribution of the unlabeled data can be effectively explored to learn a better subspace. Moreover, by virtue of the new defined label propagation procedure, the outlier in the data (shown as the green star points) can be effectively detected. In fact, for the green star samples, the values of $\mathbf{F}_{ij}$ $(j=1,2)$ calculated by Eq. (27) are very small and approach zero, while the values of $\mathbf{F}_{ij}$ $(j=3)$ approach 1.

σ) of each algorithm are reported in Table 2

In the second toy example, we generate another dataset with two classes, each of which is distributed with a half moon. In this dataset, only one data point of each class is labeled (shown as the purple points). Since the distribution of the data is non-Gaussian, here only the kernelized SODA is evaluated. Fig. 2(b) shows the contour lines learned by KSODA. The projected dimension is one in this experiment. The value in each line is the distance difference of the data point in the line to the two labeled data points after projection by KSODA. Therefore, the line with value 0 can act as the classification boundary. From Fig. 2(b) we observe that KSODA fits the nonlinear data well and obtains a desired nonlinear classification, which indicates that KSODA is a more suitable algorithm when the distribution of data is nonlinear. Although other semi-supervised dimensionality reduction methods (such as kernel SDA) can also cope with the two-moon problem, our method can learn a smoother and more accurate decision boundary. In fact, the decision boundary and the contour lines learned by our method with only two labeled data is almost the same as those learned by using all the samples as the labeled data, which indicates that our method could effectively explore the labeled and unlabeled data in the learning procedure.

5.2. Real-world applications

In this subsection, we evaluate our algorithms in four real-world problems, including face recognition, object recognition, digit recognition and text categorization.

We use PCA as the preprocessing step to eliminate the null space of data covariance matrix. For LDA, we further reduce the dimension of data such that the within-class scatter matrix $\mathbf{S}_w$ is nonsingular.

In the experiments, we select a part of data as the transductive set and the remaining data as the unseen set. We randomly split the transductive set into labeled set and unlabeled set, and the experimental results over 20 random splits are recorded. For LDA, OLDA and ODA, only the labeled set is used to learn the subspace, while for SDA, SODA and KSODA, the whole transductive set is used to learn the subspace. The 1-nearest neighbor classifier is then performed in the subspace. When there is only one labeled sample in each class, the supervised methods LDA, OLDA and ODA cannot be performed.

For ODA, SDA, SODA and KSODA, the regularization parameter μ is simply set to be $0.1\mu_0$, where μ_0 is the largest value of the diagonal elements of the matrix calculated by Eqs. (4), (28) and (38) for ODA, SODA and KSODA, respectively. For semi-supervised algorithms SDA, SODA and KSODA, the number k of neighbors to construct the graph is simply set to 8. For simplicity, we use a polynomial kernel function instead of a Gaussian kernel in KSODA to avoid tuning the kernel parameter, which is defined by

$$\mathcal{K}(\boldsymbol{x},\boldsymbol{x}') = (\boldsymbol{x}^T\boldsymbol{x}' + 1)^3. \tag{41}$$

5.2.1. Face recognition

The UMIST repository is a multiview face database, consisting of 575 images of 20 people, each covering a wide range of poses from profile to frontal views. The size of each cropped image is 112×92 with 256 gray-levels per pixel [36]. We down-sample the size of each image to 28×23 and no other preprocessing is preformed.

In this dataset, 80 percent of the data are selected as the transductive set and the remaining data are as the unseen set. In the transductive set, 1, 4 or 7 samples per class are randomly selected as the labeled set and the others are as the unlabeled set.

For semi-supervised algorithms SDA, SODA and KSODA, the weights in the neighborhood graph are computed by Eq. (24), and

Table 2
Experimental results on the unlabeled dataset and the unseen dataset over 20 random splits in each dataset (mean ± std-dev%).

Dataset	Method	1 labeled			4 labeled			7 labeled		
		Unlabel (%)	Unseen (%)	dim	Unlabel (%)	Unseen (%)	dim	Unlabel (%)	Unseen (%)	dim
UMIST	LDA	–	–	–	85.4 ± 3.1	81.9 ± 5.1	19	93.3 ± 2.5	93.1 ± 1.9	19
	OLDA	–	–	–	91.1 ± 2.4	88.8 ± 4.9	10	96.6 ± 1.5	96.1 ± 1.7	10
	ODA	–	–	–	89.0 ± 3.3	86.1 ± 4.9	10	96.3 ± 1.6	95.2 ± 1.9	10
	SDA	46.6 ± 4.9	44.9 ± 4.5	19	89.1 ± 3.7	86.9 ± 4.9	19	95.5 ± 2.1	94.5 ± 2.2	16
	SODA	**73.1 ± 5.0**	**69.6 ± 6.9**	7	**94.9 ± 2.7**	**92.6 ± 4.3**	16	**97.8 ± 0.8**	**96.5 ± 1.3**	10
	KSODA	69.7 ± 5.1	68.5 ± 8.0	7	93.7 ± 2.9	90.5 ± 4.1	16	96.7 ± 1.1	95.1 ± 1.7	16
COIL20	LDA	–	–	–	78.4 ± 1.8	77.8 ± 1.4	19	84.1 ± 2.1	83.7 ± 1.9	19
	OLDA	–	–	–	84.0 ± 2.2	84.1 ± 2.2	7	88.8 ± 2.0	89.2 ± 2.0	10
	ODA	–	–	–	85.4 ± 2.3	85.1 ± 2.7	7	91.7 ± 1.5	91.8 ± 1.9	10
	SDA	62.0 ± 2.8	61.9 ± 3.0	19	85.5 ± 1.8	85.0 ± 2.1	10	92.8 ± 2.0	91.9 ± 1.8	10
	SODA	**77.3 ± 2.4**	75.5 ± 3.0	40	89.1 ± 1.1	**89.5 ± 1.2**	13	**93.9 ± 1.0**	**93.3 ± 1.3**	16
	KSODA	76.9 ± 3.2	**75.9 ± 3.3**	16	**89.8 ± 1.6**	88.5 ± 2.3	16	93.4 ± 1.3	93.0 ± 1.3	16
Dataset	Method	5 labeled			20 labeled			50 labeled		
		Unlabel (%)	Unseen (%)	dim	Unlabel (%)	Unseen (%)	dim	Unlabel (%)	Unseen (%)	dim
USPS	LDA	70.0 ± 3.0	70.2 ± 2.9	9	51.8 ± 2.9	51.2 ± 3.1	9	78.6 ± 1.3	78.3 ± 1.1	9
	OLDA	68.3 ± 2.8	68.8 ± 3.1	9	50.0 ± 3.2	50.2 ± 2.8	9	74.7 ± 2.3	74.2 ± 2.0	9
	ODA	79.0 ± 2.2	79.5 ± 2.0	21	85.9 ± 1.1	86.6 ± 1.1	69	89.0 ± 0.7	89.1 ± 0.5	99
	SDA	79.3 ± 1.7	80.0 ± 1.6	9	86.7 ± 1.1	87.3 ± 1.0	9	90.1 ± 0.6	89.9 ± 0.4	9
	SODA	76.7 ± 4.5	77.5 ± 3.9	105	85.8 ± 1.3	86.2 ± 1.0	105	89.4 ± 0.7	89.1 ± 0.5	105
	KSODA	**80.7 ± 2.6**	**81.0 ± 2.3**	105	**89.6 ± 0.9**	**89.7 ± 0.8**	21	**93.0 ± 0.6**	**92.7 ± 0.5**	9
20NEWS	LDA	32.3 ± 7.8	33.3 ± 7.8	3	44.2 ± 8.6	45.9 ± 8.4	3	56.6 ± 9.3	57.7 ± 9.1	3
	OLDA	33.1 ± 4.1	33.6 ± 3.8	3	49.6 ± 7.8	51.9 ± 6.7	3	70.8 ± 6.3	72.2 ± 4.8	3
	ODA	37.0 ± 6.8	37.6 ± 6.0	3	69.8 ± 4.3	71.0 ± 3.4	7	86.5 ± 2.1	87.1 ± 1.8	3
	SDA	40.8 ± 8.2	41.9 ± 7.4	3	71.0 ± 4.9	72.9 ± 4.3	3	86.8 ± 2.0	87.1 ± 1.6	3
	SODA	**68.5 ± 8.5**	**67.2 ± 8.8**	23	**86.3 ± 2.1**	**86.5 ± 1.9**	11	**90.2 ± 1.3**	**90.9 ± 1.1**	3
	KSODA	48.6 ± 8.7	42.1 ± 8.0	2	67.7 ± 4.2	64.3 ± 5.8	3	79.8 ± 1.9	79.1 ± 1.9	3

The 'dim' is the corresponding dimensionality of the best result.

the variance σ is determined by

$$\sigma = \sqrt{-\frac{\bar{d}}{\ln(s)}}, \tag{42}$$

where $\bar{d}$ is the average of squared Euclidean distances for all the edged pairs on the graph, and s is searched from: $s \in \{10^{-9}/k, 10^{-7}/k, 10^{-5}/k, 10^{-3}/k, 10^{-1}/k\}$ ($k = 8$ is the neighbor number to construct the neighborhood graph).

The average results over 20 random splits with the best parameters (dimension m and variance . Fig. 3 shows the average accuracy of the unlabeled and unseen set over 20 random splits with the best parameter σ for each algorithm under different dimensions.

From the results we observe that, the two orthogonal supervised methods outperform the LDA, the semi-supervised methods outperform the supervised counterparts, and the proposed semi-supervised method (SODA) further outperforms the semi-supervised discriminant analysis. In this experiment, the result of the trace ratio based orthogonal method (ODA) is not as good as that of the ratio trace based orthogonal method (OLDA). However, if the regularization parameter μ is well tuned, as can be seen in Section 5.3, the performance of ODA in this dataset can be improved and outperforms that of OLDA.

In this dataset, the linear algorithm (SODA) outperforms the nonlinear one (KSODA). The phenomenon can also be observed in the datasets COIL20 and 20NEWS, which indicates that linear algorithm may perform very well in some applications with high-dimensional and under-sampled data, where the data are more likely to be linearly separable. The possible reason for the poor performance of KSODA is that we use a polynomial kernel function in the experiment. If the Gaussian kernel function with a well-tuned parameter is used in KSODA, the performance could be expected to be comparable to that of SODA. Nevertheless, the linear SODA is preferable in this case as it performs well without having to heavily tune the parameter.

5.2.2. Object recognition

The COIL-20 dataset [37] consists of images of 20 objects viewed from varying angles at the interval of 5°, resulting in 72 images per object. Each image is down-sampled to the size of 32 × 32.

In this dataset, 60 percent of the data are selected as the transductive set and the remaining data are as the unseen set. In the transductive set, 1, 4 or 7 samples per class are randomly selected as the labeled set and the others are as the unlabeled set.

For semi-supervised algorithms SDA, SODA and KSODA, the weights in the neighborhood graph are computed as the same as that in the experiment of face recognition.

The average results over 20 random splits with the best parameters (dimension m and variance σ) of each algorithm are reported in Table 2. Fig. 4 shows the average accuracy of the unlabeled and unseen set over 20 random splits with the best parameter σ for each algorithm under different dimensions.

In this dataset, we observe similar results. The semi-supervised methods demonstrate significant improvement over the supervised counterparts, and the proposed semi-supervised method (SODA) see a further improvement over the semi-supervised discriminant analysis. Meanwhile, the two orthogonal supervised methods also outperform the LDA, and the trace ratio based orthogonal method (ODA) outperforms the ratio trace based orthogonal method (OLDA) in this experiment.

5.2.3. Digit recognition

In this experiment, we focus on the digit recognition task using the USPS handwritten 16 × 16 digits dataset.[1] The dataset consists of 9298 images of 10 classes.

[1] Available at http://www.kernel-machines.org/data.

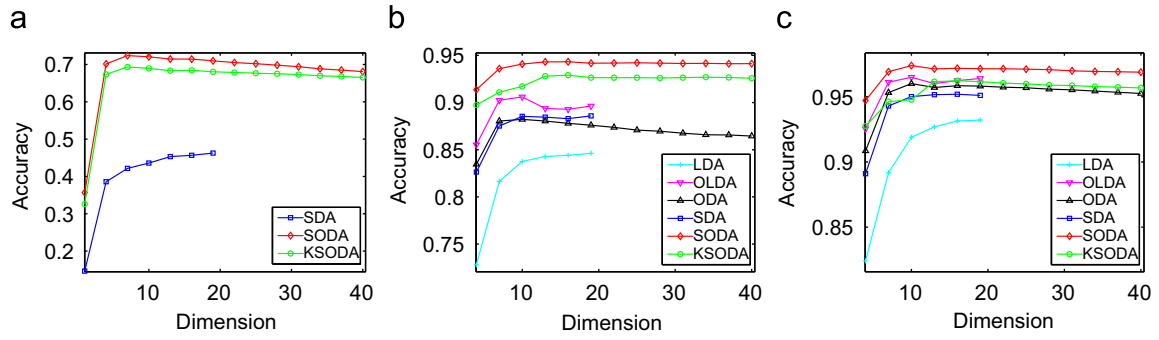

Fig. 3. Accuracy vs. dimension on the UMIST dataset. The transductive set (including the labeled set and the unlabeled set) consists 80 percent of the total data in the dataset. The number of labeled data per class is (a) 1, (b) 4, (c) 7, respectively.

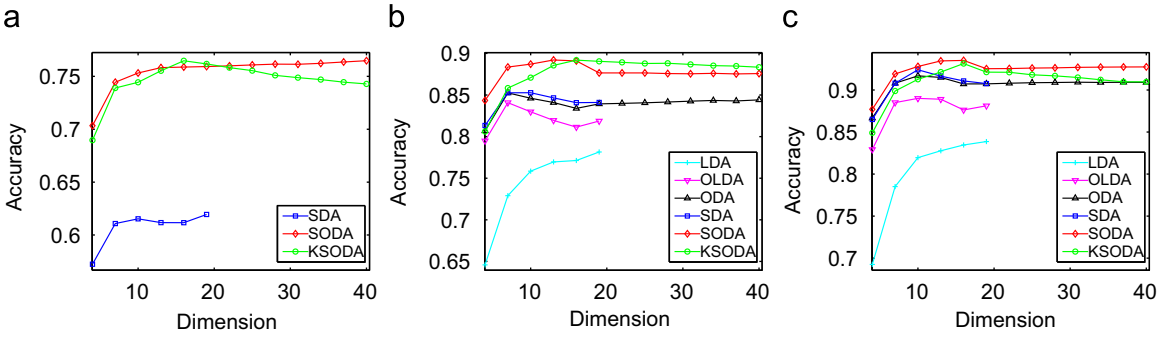

Fig. 4. Accuracy vs. dimension on the COIL20 dataset. The transductive set (including the labeled set and the unlabeled set) consists 60 percent of the total data in the dataset. The number of labeled data per class is (a) 1, (b) 4, (c) 7, respectively.

In this dataset, 20 percent of the data are selected as the transductive set and the remaining data are as the unseen set. In the transductive set, 5, 20 or 50 samples per class are randomly selected as the labeled set and the others are as the unlabeled set.

For semi-supervised algorithms SDA, SODA and KSODA, the weights in the neighborhood graph are computed as the same as that in the experiment of face recognition.

The average results over 20 random splits with the best parameters (dimension m and variance σ) of each algorithm are reported in Table 2. Fig. 5 shows the average accuracy of the unlabeled and unseen set over 20 random splits with the best parameter σ for each algorithm under different dimensions.

In this dataset, the LDA outperforms the ratio trace based orthogonal method (OLDA), and the trace ratio based orthogonal method (ODA) significantly outperforms both of the two methods. We observe that the proposed semi-supervised method SODA does not perform very well. The reason may be that the unlabeled data do not play important role on the linear algorithm. Although the semi-supervised method SDA show improvement over its supervised counterpart (LDA), it is the regularization μ but not the unlabeled data takes effect, which can be seen in Section 5.3. We can also see in Section 5.4 that the increasing number of unlabeled data does not improve the performance of the semi-supervised methods SDA and SODA, which further confirms the analysis.

We also observe that the nonlinear algorithm (KSODA) outperforms its linear one (SODA) in this dataset, which indicates that the kernel algorithm may make improvement over its linear one and perform very well in some applications with low-dimensional and over-sampled data, where the data are more likely to be linearly nonseparable but nonlinearly separable.

5.2.4. Text categorization

In this experiment, we investigated the task of text categorization using the 20-newsgroups dataset.[2] The topic *rec* which contains *autos*, *motorcycles*, *baseball*, and *hockey* was chosen from the version 20-news-18828. The articles were preprocessed with the same procedure as in [5]. This results in 3970 document vectors in a 8014-dimensional space. Finally, the documents were normalized into TFIDF representation.

In this dataset, 20 percent of the data are selected as the transductive set and the remaining data are as the unseen set. In the transductive set, 5, 20 or 50 samples per class are randomly selected as the labeled set and the others are as the unlabeled set.

In the application of text categorization, the cosine similarity is usually applied to measure the similarity between two documents [38]. Thus for the semi-supervised algorithms SDA, SODA and KSODA, the weight of edge between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ in the neighborhood graph are computed by

$$\mathbf{A}_{ij} = \frac{\boldsymbol{x}_i^T \boldsymbol{x}_j}{\|\boldsymbol{x}_i\| \|\boldsymbol{x}_j\|}. \tag{43}$$

The average results over 20 random splits with the best parameters (dimension m) of each algorithm are reported in Table 2. Fig. 6 show the average accuracy of the unlabeled and unseen set over 20 random splits under different dimensions.

In this dataset, the semi-supervised methods also demonstrate significant improvement over the supervised counterparts, and the proposed semi-supervised method (SODA) further outperforms the

[2] Available at http://people.csail.mit.edu/jrennie/20Newsgroups/.

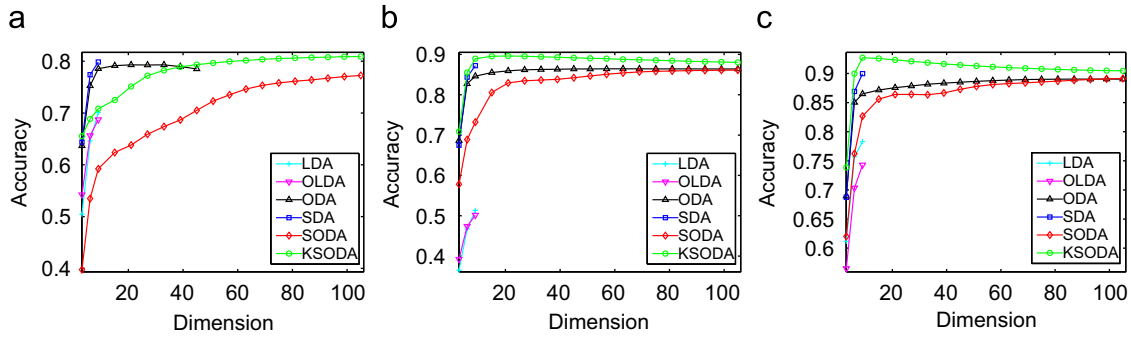

Fig. 5. Accuracy vs. dimension on the USPS dataset. The transductive set (including the labeled set and the unlabeled set) consists 20 percent of the total data in the dataset. The number of labeled data per class is (a) 5, (b) 20, (c) 50, respectively.

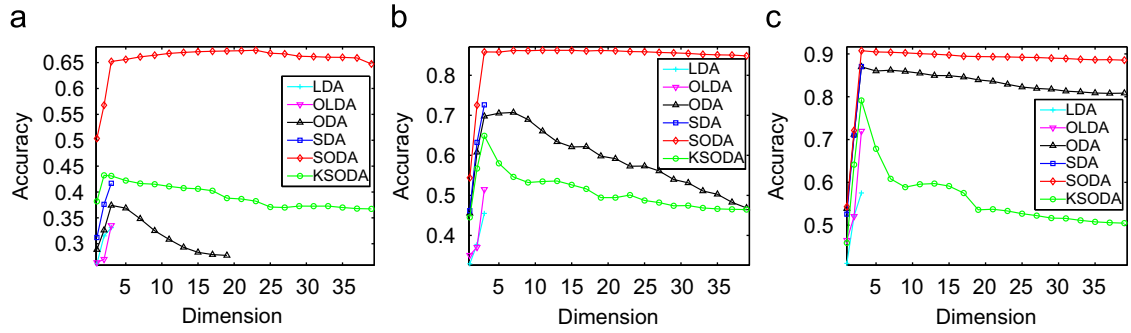

Fig. 6. Accuracy vs. dimension on the 20NEWS dataset. The transductive set (including the labeled set and the unlabeled set) consists 20 percent of the total data in the dataset. The number of labeled data per class is (a) 5, (b) 20, (c) 50, respectively.

semi-supervised discriminant analysis significantly. Similarly, the two orthogonal supervised methods also outperform LDA, and the trace ratio based orthogonal method (ODA) further outperforms the ratio trace based orthogonal method (OLDA).

5.3. Performance analysis on the parameters

For ODA and SODA, we introduced a regularization parameter μ in problems (16) and problem (32). For semi-supervised algorithms SDA and SODA, the neighborhood number k and the variance σ are introduced to construct graph. In this subsection, we present experiments to look into the influence of these parameters. The experimental setting is the same as the previous experiments.

5.3.1. Performance analysis on parameter μ

In this experiment, we analyze the performance of ODA, SDA and SODA on the regularization parameter μ. The projected dimension is set to $c-1$, where c is the number of classes. The parameter s in Eq. (42) is set to $10^{-3}/8$ (applied on UMIST, COIL20 and USPS), and the other parameters are set to the same as the experiments in Section 5.2. The parameter μ is changed from 0 to infinite. When μ is infinite, SDA becomes a supervised method. In this case, both ODA and SDA are in fact to solve the following problem:

$$\mathbf{W}_{\mathrm{ODA}} = \arg\max_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} \mathrm{tr}(\mathbf{W}^T\mathbf{S}_b\mathbf{W}), \tag{44}$$

which is equivalent to the orthogonal centroid method proposed in [39]. When μ is infinite, SODA is in fact to solve the following problem:

$$\mathbf{W}_{\mathrm{SODA}} = \arg\max_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} \mathrm{tr}(\mathbf{W}^T\tilde{\mathbf{S}}_b\mathbf{W}), \tag{45}$$

which can be seen as a semi-supervised extension to the orthogonal centroid method [39].

In the datasets UMIST, COIL20, USPS and 20NEWS, the labeled data number per class is 4, 4, 20, 20, respectively. In the 20 runs, the average accuracies of the unlabeled and unseen set under different μ are shown in Fig. 7.

From the results we can see that the value of μ have significant influence on classification performance. Therefore, how to effectively and efficiently select the value of μ is a very important issue. An interesting observation is that in the text dataset 20NEWS, all of the three methods ODA, SDA and SODA perform near to the optimum when $\mu \to \infty$. The observation indicates that the orthogonal centroid method and the semi-supervised orthogonal centroid method are suitable to the application on text data, which is consistent with the experiments in [18,39].

Another observation is that the performance is almost monotonic with respect to the parameter μ, which provides us with a clue to select a suitable parameter μ. As the performance of SODA is significantly deteriorated on the USPS dataset when μ is set to $0.1\mu_0$ (μ_0 is calculated as in Section 5.2), we set μ to be μ_0 for SODA on the USPS in the subsequent experiments.

5.3.2. Performance analysis on the neighbor number k

In this experiment, we analyze the performance of SDA and SODA on the neighbor number k. The projected dimension is set to $c-1$, the parameter s in Eq. (42) is set to $10^{-3}/8$ (applied on UMIST, COIL20 and USPS), and the other parameters are set to the same as the experiments in Section 5.2.

In the datasets UMIST, COIL20, USPS and 20NEWS, the labeled data number per class is 4, 4, 20, 20, respectively. In the 20 runs, the average accuracies of the unlabeled and unseen set under different parameter k are shown in Fig. 8. The experimental results show that the neighbor number k is not very sensitive to performance. We can set k to be a relatively small value in application.

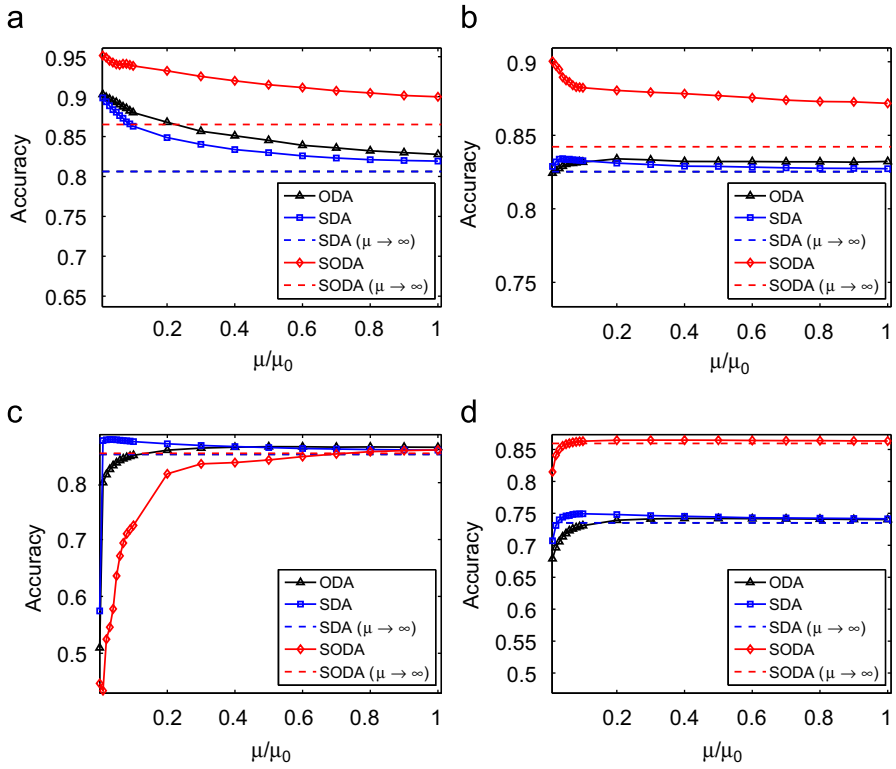

Fig. 7. Accuracy vs. the regularization parameter μ, where μ_0 is the largest value of the diagonal elements of the matrix calculated by Eqs. (4), (28) and (38) for ODA, SODA and KSODA, respectively. (a) UMIST. (b) COIL20. (c) USPS. (d) 20NEWS.

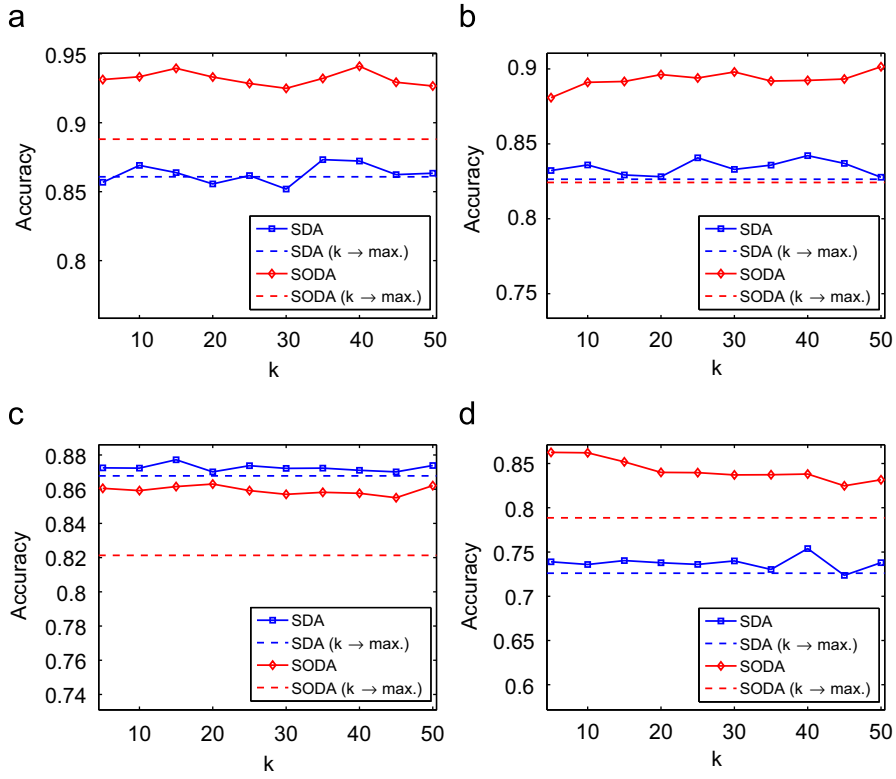

Fig. 8. Accuracy vs. the neighbor number k. The dashed line ($k \rightarrow$ max) indicates the accuracy when k reaches the maximum of the available neighbors, i.e., $n-1$, where n is the number of the training data. (a) UMIST. (b) COIL20. (c) USPS (d). 20NEWS.

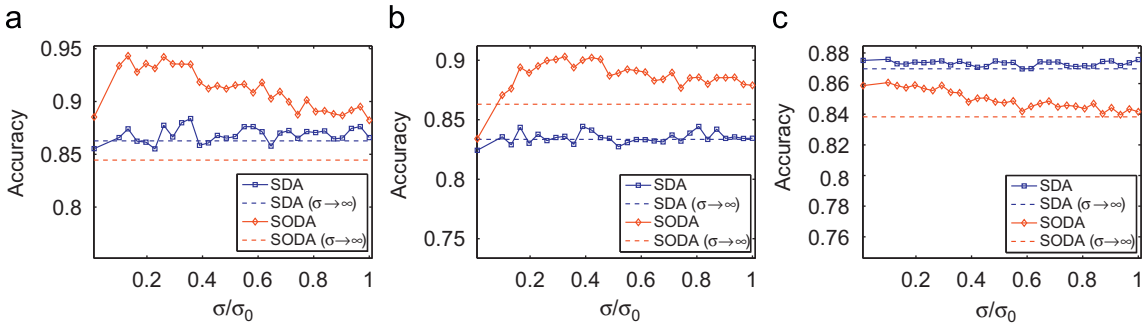

Fig. 9. Accuracy vs. variance σ. σ_0 is the value when $s=\frac{1}{8}$ in Eq. (42). (a) UMIST. (b) COIL20. (c) USPS.

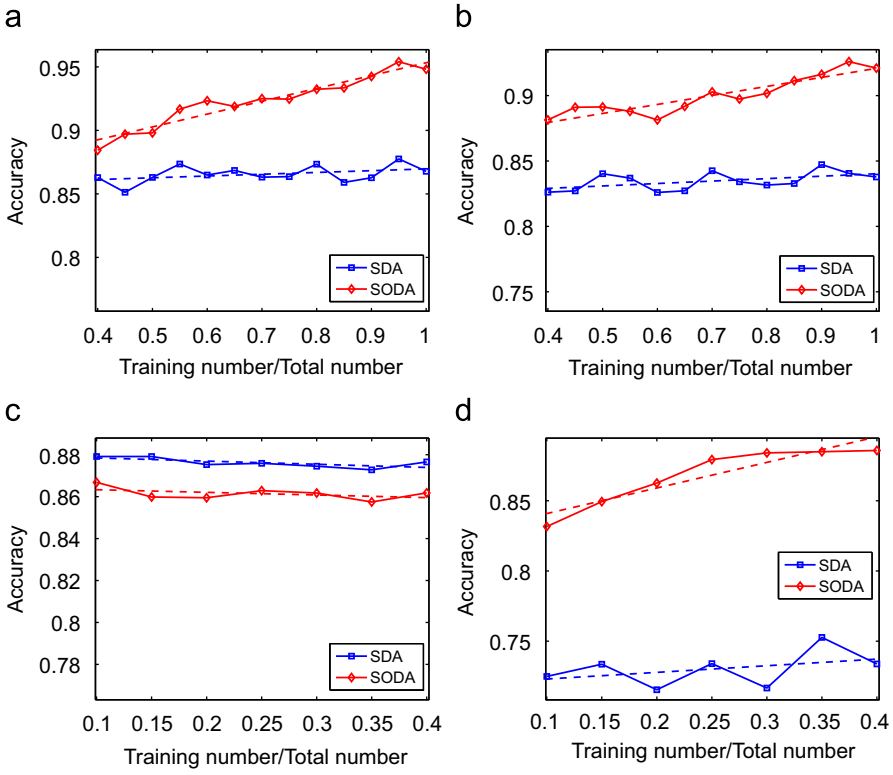

Fig. 10. Accuracy vs. the number of training data (including the labeled data and the unlabeled data). The dashed line indicates the fitting line for the points. (a) UMIST. (b) COIL20. (c) USPS. (d) 20NEWS.

5.3.3. Performance analysis on variance σ

In this experiment, we analyze the performance of SDA and SODA on the variance σ in Eq. (24). The projected dimension is set to $c-1$, and the other parameters are set to the same as the experiments in Section 5.2. The variance σ is changed from 0 to infinite.

The datasets UMIST, COIL20, and USPS are used in this experiment, and the labeled data number per class is 4, 4 and 20, respectively. In the 20 runs, the average accuracies of the unlabeled and unseen set under different variance σ are shown in Fig. 9.

The experimental results show that the value of σ has significant influence on classification performance. Therefore, how to effectively and efficiently select the value of σ is also an important issue.

An interesting observation is that the performance of SODA tends to monotonically decrease with respect to the parameter σ when the value of σ is not too small, which gives us a guideline to search a suitable value of σ. Empirically, we can search the σ in the range between $0.2\sigma_0$ and $0.6\sigma_0$ in application, where σ_0 is the value when $s=1/k$ in Eq. (42), k is the number of neighbors to construct the neighborhood graph.

5.4. Performance analysis on the number of unlabeled data

Based on the motivation that unlabeled data may be useful to improve the performance, an effective semi-supervised may improve the performance when the number of the available unlabeled data increases. In this subsection, we present experiments to verify the performances of SDA and SODA on increasing number of unlabeled data.

In this experiments, the projected dimension is set to $c-1$, the parameter s in Eq. (42) is set to $10^{-3}/8$ (applied on UMIST, COIL20 and USPS), and the other parameters are set to the same as the experiments in Section 5.2. For UMIST, COIL20, USPS and 20NEWS, the labeled data number per class is 4, 4, 20, 20, respectively. The

number of training data (including the labeled data and the unlabeled data) is changed from $0.4n$ to n for UMIST and COIL20, and from $0.1n$ to $0.4n$ for USPS and 20NEWS, where n is the total number of data in the dataset.

In the 20 runs, the average accuracies of the unlabeled and unseen set under different number of unlabeled data are shown in Fig. 10. We also show the fitting line for the points in the figure. In the four datasets UMIST, COIL20, USPS and 20NEWS, the slope of the fitting line for SODA is 0.1011, 0.0690, −0.0125, 0.1823, respectively, while the slope of the fitting line for SDA is 0.0141, 0.0187, −0.0152, 0.0471, respectively.

On the datasets UMIST, COIL20 and 20NEWS, the performance of SODA is significantly improved when the number of the available unlabeled data increases, and the improvement speed is much faster than that of SDA. The experimental result indicates that the proposed SODA can explore the unlabeled data more effective than SDA to learn a better subspace, which confirms the analysis in the previous sections.

On the dataset USPS, both SODA and SDA do not improve the performance when the number of the available unlabeled data increases. The experimental result indicates that the unlabeled data in USPS may not be very helpful to improve the performance when a linear algorithm is applied.

6. Conclusion

In this paper, we re-analyzed the trace ratio problem, and developed a faster algorithm than a recently proposed one to solve the orthogonal constrained trace ratio problem. Based on this problem, we proposed a novel semi-supervised orthogonal discriminant analysis via label propagation. The algorithm propagate the label information from the labeled data to the unlabeled data through a specially designed label propagation, and thus the distribution of the unlabeled data are effectively explored to learn a better subspace. Extensive experiments are presented to verify the effectiveness of our algorithms, and the experimental results demonstrate much improvement over the state-of-the-art algorithms.

Acknowledgements

This work is supported by NSFC (Grant No. 60721003 and 60675009).

Appendix A. Proof of Theorem 1

Before proving the theorem, we first prove the following lemma:

Lemma 1. *If* $\lambda_1 \leqslant \lambda_2$, *then* $\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_w \mathbf{W}_{\lambda_1}) \geqslant \mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_w \mathbf{W}_{\lambda_2})$.

Proof. Note that $\mathbf{W}_\lambda = \arg\max_{\mathbf{W}^T\mathbf{W}=\mathbf{I}} \mathrm{tr}(\mathbf{W}^T(\mathbf{S}_b - \lambda \mathbf{S}_w)\mathbf{W})$, so we have the following two inequalities:

$$\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_b \mathbf{W}_{\lambda_1}) - \lambda_1 \mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_w \mathbf{W}_{\lambda_1}) \geqslant \mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_b \mathbf{W}_{\lambda_2}) - \lambda_1 \mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_w \mathbf{W}_{\lambda_2}), \tag{A.1}$$

$$\mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_b \mathbf{W}_{\lambda_2}) - \lambda_2 \mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_w \mathbf{W}_{\lambda_2}) \geqslant \mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_b \mathbf{W}_{\lambda_1}) - \lambda_2 \mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_w \mathbf{W}_{\lambda_1}). \tag{A.2}$$

Summing the above two inequalities on the two sides, we have

$$(\lambda_2 - \lambda_1)\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_w \mathbf{W}_{\lambda_1}) \geqslant (\lambda_2 - \lambda_1)\mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_w \mathbf{W}_{\lambda_2}). \tag{A.3}$$

Then if $\lambda_1 \leqslant \lambda_2$, we have $\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_w \mathbf{W}_{\lambda_1}) \geqslant \mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_w \mathbf{W}_{\lambda_2})$. □

Now we begin to prove the Theorem 1.

Proof. Suppose $\lambda_1 \leqslant \lambda_2 \leqslant \lambda^*$, according to inequality (A.2), we have

$$\begin{aligned}
&\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_b \mathbf{W}_{\lambda_1}) \\
&\leqslant \mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_b \mathbf{W}_{\lambda_2}) - \lambda_2 \mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_w \mathbf{W}_{\lambda_2}) + \lambda_2 \mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_w \mathbf{W}_{\lambda_1}) \\
&\Rightarrow \frac{\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_b \mathbf{W}_{\lambda_1})}{\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_w \mathbf{W}_{\lambda_1})} \leqslant \frac{\mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_b \mathbf{W}_{\lambda_2}) - \lambda_2 \mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_w \mathbf{W}_{\lambda_2})}{\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_w \mathbf{W}_{\lambda_1})} + \lambda_2 \\
&\Rightarrow \frac{\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_b \mathbf{W}_{\lambda_1})}{\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_w \mathbf{W}_{\lambda_1})} \leqslant \frac{\mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_b \mathbf{W}_{\lambda_2}) - \lambda_2 \mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_w \mathbf{W}_{\lambda_2})}{\mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_w \mathbf{W}_{\lambda_2})} + \lambda_2 \\
&\Rightarrow \frac{\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_b \mathbf{W}_{\lambda_1})}{\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_w \mathbf{W}_{\lambda_1})} \leqslant \frac{\mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_b \mathbf{W}_{\lambda_2})}{\mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_w \mathbf{W}_{\lambda_2})},
\end{aligned}$$

where the last but one inequality follows according to Lemma 1 and a fact that $\mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_b \mathbf{W}_{\lambda_2}) - \lambda_2 \mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_w \mathbf{W}_{\lambda_2}) \geqslant 0$ when $\lambda_2 \leqslant \lambda^*$ [40]. Therefore, if $\lambda_1 \leqslant \lambda_2 \leqslant \lambda^*$, then $f(\lambda_1) \leqslant f(\lambda_2)$. Thus function $f(\lambda)$ is monotonically increasing when $\lambda \leqslant \lambda^*$.

On the other hand, suppose $\lambda^* \leqslant \lambda_1 \leqslant \lambda_2$, according to inequality (A.1) and Lemma 1, we have

$$\begin{aligned}
&\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_b \mathbf{W}_{\lambda_1}) - \mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_b \mathbf{W}_{\lambda_2}) \\
&\geqslant \lambda_1(\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_w \mathbf{W}_{\lambda_1}) - \mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_w \mathbf{W}_{\lambda_2})) \\
&\Rightarrow \mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_b \mathbf{W}_{\lambda_1}) - \mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_b \mathbf{W}_{\lambda_2}) \\
&\quad \geqslant \lambda^*(\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_w \mathbf{W}_{\lambda_1}) - \mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_w \mathbf{W}_{\lambda_2})) \\
&\Rightarrow \mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_b \mathbf{W}_{\lambda_1}) - \mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_b \mathbf{W}_{\lambda_2}) \\
&\quad \geqslant \frac{\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_b \mathbf{W}_{\lambda_1})}{\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_w \mathbf{W}_{\lambda_1})}(\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_w \mathbf{W}_{\lambda_1}) - \mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_w \mathbf{W}_{\lambda_2})) \\
&\Rightarrow \frac{\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_b \mathbf{W}_{\lambda_1})}{\mathrm{tr}(\mathbf{W}_{\lambda_1}^T \mathbf{S}_w \mathbf{W}_{\lambda_1})} \geqslant \frac{\mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_b \mathbf{W}_{\lambda_2})}{\mathrm{tr}(\mathbf{W}_{\lambda_2}^T \mathbf{S}_w \mathbf{W}_{\lambda_2})}.
\end{aligned}$$

Therefore, if $\lambda^* \leqslant \lambda_1 \leqslant \lambda_2$, then $f(\lambda_1) \geqslant f(\lambda_2)$. Thus function $f(\lambda)$ is monotonically decreasing when $\lambda \geqslant \lambda^*$. □

References

[1] M. Seeger, Learning with labeled and unlabeled data, Technical Report, The University of Edinburgh, 2000.
[2] A. Blum, S. Chawla, Learning from labeled and unlabeled data using graph mincuts, in: ICML, 2001, pp. 19–26.
[3] M. Szummer, T. Jaakkola, Partially labeled classification with Markov random walks, in: NIPS, 2001, pp. 945–952.
[4] X. Zhu, Z. Ghahramani, J.D. Lafferty, Semi-supervised learning using Gaussian fields and harmonic functions, in: ICML, 2003, pp. 912–919.
[5] D. Zhou, O. Bousquet, T.N. Lal, J. Weston, B. Schölkopf, Learning with local and global consistency, in: NIPS, 2004.
[6] O. Chapelle, A. Zien, Semi-supervised classification by low density separation, in: The Tenth International Workshop on Artificial Intelligence and Statistics, 2005.
[7] M. Belkin, P. Niyogi, V. Sindhwani, Manifold regularization: a geometric framework for learning from labeled and unlabeled examples, Journal of Machine Learning Research 7 (2006) 2399–2434.
[8] O. Chapelle, B. Schölkopf, A. Zien, Semi-Supervised Learning, MIT Press, Cambridge, MA, 2006.
[9] D. Zhang, Z.-H. Zhou, S. Chen, Semi-supervised dimensionality reduction, in: SDM, 2007.
[10] D. Cai, X. He, J. Han, Semi-supervised discriminant analysis, in: ICCV, 2007.
[11] Y. Song, F. Nie, C. Zhang, S. Xiang, A unified framework for semi-supervised dimensionality reduction, Pattern Recognition 41 (9) (2008) 2789–2799.
[12] M. Sugiyama, T. Idé, S. Nakajima, J. Sese, Semi-supervised local Fisher discriminant analysis for dimensionality reduction, in: PAKDD, 2008, pp. 333–344.
[13] Y. Zhang, D.-Y. Yeung, Semi-supervised discriminant analysis via CCCP, in: ECML/PKDD, 2008, pp. 644–659.
[14] Y. Zhang, D.-Y. Yeung, Semi-supervised discriminant analysis using robust path-based similarity, in: CVPR, 2008.
[15] Y. Bengio, J.-F. Paiement, P. Vincent, O. Delalleau, N.L. Roux, M. Ouimet, Out-of-sample extensions for LLE, ISOMAP, MDS, eigenmaps, and spectral clustering, in: NIPS, 2003.
[16] S. Yan, D. Xu, B. Zhang, H. Zhang, Q. Yang, S. Lin, Graph embedding and extensions: a general framework for dimensionality reduction, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (1) (2007) 40–51.

[17] H. Wang, S. Yan, D. Xu, X. Tang, T.S. Huang, Trace ratio vs. ratio trace for dimensionality reduction, in: CVPR, 2007.
[18] J. Yan, N. Liu, B. Zhang, S. Yan, Z. Chen, Q. Cheng, W. Fan, W.-Y. Ma, OCFS: optimal orthogonal centroid feature selection for text categorization, in: SIGIR, 2005, pp. 122–129.
[19] D. Cai, X. He, Orthogonal locality preserving indexing, in: SIGIR, 2005, pp. 3–10.
[20] C.H.Q. Ding, T. Li, W. Peng, H. Park, Orthogonal nonnegative matrix tri-factorizations for clustering, in: KDD, 2006, pp. 126–135.
[21] D. Cai, X. He, J. Han, H.-J. Zhang, Orthogonal Laplacian faces for face recognition, IEEE Transactions on Image Processing 15 (11) (2006) 3608–3614.
[22] E. Kokiopoulou, Y. Saad, Orthogonal neighborhood preserving projections: a projection-based dimensionality reduction technique, IEEE Transactions on Pattern Analysis and Machine Intelligence 29 (12) (2007) 2143–2156.
[23] X. Liu, J. Yin, Z. Feng, J. Dong, L. Wang, Orthogonal neighborhood preserving embedding for face recognition, in: ICIP, 2007, pp. I-133–I-136.
[24] H. Wang, S. Chen, Z. Hu, W. Zheng, Locality-preserved maximum information projection, IEEE Transactions on Neural Networks 19 (4) (2008) 571–585.
[25] L. Duchene, S. Leclerq, An optimal transformation for discriminant and principal component analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 10 (6) (1988) 978–983.
[26] J. Ye, Characterization of a family of algorithms for generalized discriminant analysis on undersampled problems, Journal of Machine Learning Research 6 (2005) 483–502.
[27] R.A. Horn, C.R. Johnson, Matrix Analysis, Cambridge University Press, Cambridge, 1990.
[28] C. Shen, H. Li, M.J. Brooks, Supervised dimensionality reduction via sequential semidefinite programming, Pattern Recognition 41 (12) (2008) 3644–3652.
[29] B. Scholkopf, A.J. Smola, Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond, MIT Press, Cambridge, MA, USA, 2001.
[30] Z. Jin, J.-Y. Yang, Z.-S. Hu, Z. Lou, Face recognition based on the uncorrelated discriminant transformation, Pattern Recognition 34 (7) (2001) 1405–1416.
[31] S. Yan, X. Tang, Trace quotient problems revisited, in: ECCV, 2006, pp. 232–244.
[32] F. Nie, S. Xiang, Y. Jia, C. Zhang, S. Yan, Trace ratio criterion for feature selection, in: AAAI, 2008.
[33] D.A. Spielman, S. H. Teng, Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems, in: Annual ACM Symposium on Theory of Computing, 2004.
[34] R. Courant, D. Hilbert, Methods of Mathematical Physics, Interscience Publishers, New York, 1953.
[35] V.N. Vapnik, The Nature of Statistical Learning Theory, Springer, New York, NY, USA, 1995.
[36] D.B. Graham, N.M. Allinson, Characterizing Virtual Eigensignatures for General Purpose Face Recognition in Face Recognition: From Theory to Applications, in: NATO ASI Series F, Computer and Systems Sciences, vol. 163, 1998, pp. 446–456.
[37] S.A. Nene, S.K. Nayar, H. Murase, Columbia object image library (COIL-20), Technical Report CUCS-005-96, Columbia University, 1996.
[38] F. Sebastiani, C.N.D. Ricerche, Machine learning in automated text categorization, ACM Computing Surveys 34 (2002) 1–47.
[39] H. Park, M. Jeon, J. Rosen, Lower dimensional representation of text data based on centroids and least squares, BIT Numerical Mathematics 43 (2) (2003) 427–448.
[40] F. Nie, S. Xiang, C. Zhang, Neighborhood minmax projections, in: IJCAI, 2007, pp. 993–998.

About the Author—FEIPING NIE received his B.S. degree in Computer Science from North China University of Water Conservancy and Electric Power, China, in 2000, and received his M.S. degree in Computer Science from Lanzhou University, China, in 2003. He is currently a Ph.D. Candidate in the Department of Automation, Tsinghua University, China. His research interests include machine learning, pattern recognition, data mining and image processing.

About the Author—SHIMING XIANG received his B.S. degree from Department of Mathematics of Chongqing Normal University, China, in 1993 and M.S. degree from Department of Mechanics and Mathematics of Chongqing University, China, in 1996, and Ph.D. degree from Institute of Computing Technology, Chinese Academy of Sciences, China, in 2004. He was a post Doctor in Department of Automation, Tsinghua University until 2006. He is currently an associate researcher in Institute of Automation Chinese Academy of Science. His interests include computer vision, pattern recognition, machine learning, etc.

About the Author—YANGQING JIA received his B.S. degree from Department of Automation, Tsinghua University, China, in 2006. He is currently a Master candidate in Department of Automation, Tsinghua University. His research interests focus on machine learning and its applications.

About the Author—CHANGSHUI ZHANG received his B.S. degree in Mathematics from Peking University, China, in 1986, and Ph.D. degree from Department of Automation, Tsinghua University in 1992. He is currently a professor of Department of Automation, Tsinghua University. He is an Associate Editor of the journal Pattern Recognition. His interests include artificial intelligence, image processing, pattern recognition, machine learning, evolutionary computation and complex system analysis, etc.