Soft label based Linear Discriminant Analysis for image recognition and retrieval

Mingbo Zhao a, Zhao Zhang b, Tommy W.S. Chow a, Bing Li c,∗

a Department of Electronic Engineering, City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong Special Administrative Region
b School of Computer Science and Technology, Soochow University, PR China
c School of Economics, Wuhan University of Technology, PR China

A R T I C L E   I N F O

Article history:
Received 4 January 2013
Accepted 24 January 2014
Available online 5 February 2014

Keywords:
Linear Discriminant Analysis
Semi-supervised dimensionality reduction
Soft label
Label propagation

A B S T R A C T

Dealing with high-dimensional data has always been a major problem in the research of pattern recognition and machine learning. Among all the dimensionality reduction techniques, Linear Discriminant Analysis (LDA) is one of the most popular methods that have been widely used in many classification applications. But LDA can only utilize labeled samples while neglect the unlabeled samples, which are abundant and can be easily obtained in the real world. In this paper, we propose a new dimensionality reduction method by using unlabeled samples to enhance the performance of LDA. The new method first propagates the label information from labeled set to unlabeled set via a label propagation process, where the predicted labels of unlabeled samples, called soft labels, can be obtained. It then incorporates the soft labels into the construction of scatter matrices to find a transformed matrix for dimensionality reduction. In this way, the proposed method can preserve more discriminative information, which is preferable when solving the classification problem. Extensive simulations are conducted on several datasets and the results show the effectiveness of the proposed method.

© 2014 Elsevier Inc. All rights reserved.

1. Introduction

Dealing with high-dimensional data has always been a major problem in the research of pattern recognition and machine learning. Typical applications of these include face recognition, document categorization, and image retrieval. Finding a low-dimensional representation of high-dimensional space, namely dimensionality reduction is thus of great practical importance. The goal of dimensionality reduction is to reduce the complexity of input space and to embed high-dimensional space into a low-dimensional subspace. Given that the within-class scatter matrix is nonsingular, the optimization problem of LDA can be solved by generalized eigen-value decomposition (GEVD), i.e. to find the largest eigenvectors corresponding to the eigenvalues of $S_w^{-1}S_b$ [27]. However, for many applications where the number of dimensionality is much larger than that of the samples, the within-class scatter matrix tends to be singular. Hence the optimal projection matrix may be found incorrect. This is the so-called small sample problem and many variants of LDA have been proposed to solve it, which include Regularized LDA [23], Null-space LDA [24], Uncorrelated LDA [25], SRDA [26].

In general, LDA is supervised, which means it requires label information. Although supervised methods generally outperform unsupervised methods, a large number of labeled samples are needed in order to achieve satisfactory results [27]. But in many cases, labeling large number of samples is time-consuming and costly. On the other hand, unlabeled samples are abundant and can be easily obtained in the real world. Thus, semi-supervised learning methods [5–12,41–45], which incorporate both labeled and unlabeled samples into learning procedure, have become more effective than only relying on supervised learning. Many semi-supervised methods have been proposed in the past few years, which include Gaussian Fields and Harmonic Functions (GFHF) [5], Learning with Local and Global Consistency (LLGC) [6],

while minimizing the within-class scatter matrix $S_w$ in the low-dimensional subspace. Given that the within-class scatter matrix is nonsingular, the optimization problem of LDA can be solved by generalized eigen-value decomposition (GEVD), i.e. to find the largest eigenvectors corresponding to the eigenvalues of $S_w^{-1}S_b$ [27]. However, for many applications where the number of dimensionality is much larger than that of the samples, the within-class scatter matrix tends to be singular. Hence the optimal projection matrix may be found incorrect. This is the so-called small sample problem and many variants of LDA have been proposed to solve it, which include Regularized LDA [23], Null-space LDA [24], Uncorrelated LDA [25], SRDA [26].

In general, LDA is supervised, which means it requires label information. Although supervised methods generally outperform unsupervised methods, a large number of labeled samples are needed in order to achieve satisfactory results [27]. But in many cases, labeling large number of samples is time-consuming and costly. On the other hand, unlabeled samples are abundant and can be easily obtained in the real world. Thus, semi-supervised learning methods [5–12,41–45], which incorporate both labeled and unlabeled samples into learning procedure, have become more effective than only relying on supervised learning. Many semi-supervised methods have been proposed in the past few years, which include Gaussian Fields and Harmonic Functions (GFHF) [5], Learning with Local and Global Consistency (LLGC) [6],
Semi-supervised Discriminant Analysis (SDA) [9] and Laplacian regularized Least Square (LapRLS) [11].

The above methods can be divided into two categories: transductive method and inductive method. Two well-known transductive methods are GHF and LLGC. These methods work in a transductive way by propagating the label information from labeled set to unlabeled set. But they cannot predict the class labels of new-coming samples hence suffering out-of-sample problem. In contrast, inductive methods, such as SDA and LapRLS can solve such problem. These methods firstly construct a manifold regularized term to preserve the geometrical structure with both labeled and unlabeled set [17]. Then, they aim to find a transformed matrix to perform dimensionality reduction by incorporating the manifold regularized term into the original objective function of supervised algorithms. Hence, the new-coming samples can be projected into a low-dimensional subspace by using such transformed matrix and the out-of-sample problem can be naturally solved, which is more practical in real-world applications.

In this paper, we propose a new semi-supervised dimensionality reduction method to enhance conventional LDA performance by incorporating the soft labels into the scatter matrices. The proposed method first propagates the label information from labeled set to unlabeled set via label propagation, where the predicted class labels of unlabeled samples, called soft labels, can be obtained. Then it finds a transformed matrix to perform dimensionality reduction by incorporating the soft labels into the scatter matrices. The proposed method can be viewed as a unified framework to extend many existing LDA algorithms to their semi-supervised versions. Its basic ideas are different from those semi-supervised algorithms such as SDA and LapRLS. These methods use the labeled and unlabeled samples in a simple manner to construct a manifold regularized term and add it to the objective function of supervised algorithms. But in the proposed algorithm, by incorporating the soft labels into training, it can well preserve the probability distribution of samples both in labeled and unlabeled set hence obtaining a better subspace for dimensionality reduction. We also analyze our proposed algorithm under a least square framework. It can be concluded that given a certain class indicator, the optimization problem of the proposed method can be equivalent to a weighted least square problem.

The main contribution of this paper is summarized as follows:

(1) We present an effective label propagation procedure, which is based on a new local reconstruction graph with symmetrization and normalization. The symmetrization can add more connections of a sample with its neighborhoods and the normalization can handle with the case when the density of dataset varies dramatically.

(2) The proposed SL-LDA can preserve more discriminative information in the soft labels of unlabeled samples than other methods, which is good to the performance for classification. It can also be easily extended to its kernel version by using kernel tricks [21,22].

(3) Motivated by the equivalence between LDA and the least square problem [18–20], we extend this relationship and further analyze SL-LDA under a weighted least square problem (W-LS). Based on such relationship, we then propose a more efficient approach for calculating the optimal solution of SL-LDA, which is a linear system of equation and can be performed on large-scale dataset.

This paper is organized as follows: In Section 2, we will present an effective label propagation procedure with outlier detection and dealing with noisy labels. In Section 3, we will introduce our soft label based Linear Discriminant Analysis (SL-LDA) for semi-supervised dimensionality reduction. We will also build a close relationship between SL-LDA and W-LS in this section and propose a more efficient approach for solving SL-LDA. The simulation results based on extensive datasets are shown in Section 5 and the final conclusions are drawn in Section 6.

2. Transductive learning via label propagation

Let \( X = [X_l, X_u] \in \mathbb{R}^{d \times (l+u)} \) be the labeled and unlabeled set, where each sample in \( X_l \) is associated with a class label \( c_i \), \( i \in [1, 2, \ldots, c] \). The goal of label propagation is to distribute the label information of labeled set to the unlabeled set according to the distribution associated with both labeled and unlabeled set \([5–8, 39, 46]\), through which the predicted labels of unlabeled set, called soft labels can be obtained.

2.1. Graph construction

In label propagation, a similarity matrix must be defined for evaluating the similarities between any two samples. The similarity matrix can be approximated by a neighborhood graph associated with weights on the edges. Officially, let \( G = (V, E) \) denote this graph, where \( V \) is the vertex set of \( G \) representing the training samples, \( E \) is the edge set of \( G \) associated with a weight matrix containing the local information between two nearby samples. There are many strategies to define the weight matrix \( W \), a typical one is to use Gaussian function as [5,6,39]:

\[
    w_{ij} = \exp(-||x_i - x_j||^2/\sigma) \quad x_i \in N_k(x_j) \text{ or } x_j \in N_k(x_i),
\]

where \( N_k(x_j) \) is the \( k \) neighborhood set of \( x_j \), \( \sigma \) is the Gaussian function variance. However, \( \sigma \) is hard to be determined and even a small variation of \( \sigma \) can make the results dramatically different [7]. Wang and Zhang have proposed another strategy to construct \( G \) by using the neighborhood information of samples [7]. This strategy assumes that each sample can be reconstructed by a linear combination of its neighborhoods [2], i.e. \( x_j \approx \sum_{i \in N_k(x_j)} w_{ij} x_i \). It then calculates the weight matrix by solving a standard quadratic programming (QP) problem as:

\[
    \min \left\| x_l - \sum_{j \in N_k(x_l)} w_{lj} x_j \right\|_2^2 \quad \text{s.t. } w_{lj} \geq 0, \quad \sum_{j \in N_k(x_l)} w_{lj} = 1. \quad (2)
\]

The above strategy is empirically better than the Gaussian function, as the weight matrix can be automatically calculated in a closed form once the neighborhood size is fixed. In addition, in order to make a connection to the normalized graph, we symmetrize \( W \) as \( W \rightarrow (W + W^T)/2 \) or \( W \rightarrow (W_l + W_u)/2 \). The advantage of this step is that it considers the node degree of each sample and a sample with large node degree can connect more neighborhoods. Then similar to [39], we normalize \( W \) as \( \tilde{W} = D^{-1/2}WD^{-1/2} \) or \( \tilde{W} = \tilde{W}_l = \sqrt{d_l} \tilde{d}_l \tilde{d}_l \tilde{d}_l \). The normalization can strengthen the weights in the low-density region and weaken the weights in the high density region, which is good for handling the case that the density of dataset varies dramatically. Finally, to satisfy the sum-to-one constraint as Eq. (2), the weight matrix \( W \) is set as \( W \rightarrow \tilde{W}D^{-1} \) or \( \tilde{W} \rightarrow \tilde{W} / \sum_{j \in N_k(x)} \tilde{w}_{lj} \), where \( D \) is the diagonal matrix satisfying \( d_j = \sum_{i \in N_k(x_j)} \tilde{w}_{ij} \). The basic steps for graph construction can be seen in Table 1.

2.2. Label propagation process

We will then predict the labels of unlabeled samples based on a label propagation process, through which the soft labels of unlabeled set can be obtained. Let \( Y = [y_1, y_2, \ldots, y_u] \in \mathbb{R}^{(c+1) \times (l+u)} \) be
Graph construction.

| Input: Data matrix $X \in \mathbb{R}^{n \times (m+1)}$ |
| Output: The constructed Graph $\hat{W}$ |

Algorithm:
1. Construct the neighborhood graph and calculate the weight matrix $\hat{W}$ as in Eq. (2).
2. Symmetrize $W$ as $W = \frac{(W + W^T)}{2}$.
3. Normalize $W$ as $\hat{W} = D^{-1/2}WD^{-1/2}$, where $D$ is the diagonal matrix satisfying $d_{ii} = \sum_{j=1}^{n} w_{ij}$.
4. Output $\hat{W}$.

Based on the properties of matrix, i.e. $\lim_{t \to \infty} (P_{l})^{t} = 0$ and $\lim_{t \to \infty} \sum_{k=0}^{t} (P_{l})^{k} = (I - P_{l})^{-1}$, the iterative process of Eq. (4) converges to:

$$F = \lim_{t \to \infty} F(t) = YI_{t}(I - P_{l})^{-1}.$$  

It can be easily proved that the sum of each column of $F$ is equal to 1 [39] (Corollary 1, we also include the proof in Appendix 6.1 to make our paper self-contained). This indicates that the elements in $F$ are the probability values and $f_{ij}$ can be seen as the posterior probability of $x_{i}$ belonging to the $j$th class. When $i = c + 1$, $f_{ij}$ represents the probability of $x_{i}$ belonging to the outliers. The basic steps for the proposed label propagation process can be seen in Table 1. A simple example of the label propagation process with outlier detection can be seen in Fig. 1. However, in many real-world applications, there may exist noisy labeled samples due to the careless of the annotator, our method can also automatically erase such noisy labeled samples by setting $\zeta > 0$. A case in point is given in Fig. 2, where we generalize a two-moon dataset with wrong labeled samples.

In order to analyze the sensitivities in terms of how to bootstrap the label propagation, we compare the performance of the proposed label propagation with those of GGL and LNP. Specifically, we in Fig. 3 first show the parameter sensitivities of the proposed label propagation with comparison to that of GGL. In Fig. 3, we generalize a two-cycle dataset with two samples labeled per class. The marker size of each unlabeled data point is proportional to its predicted label. From Fig. 3, we can see that the performance of GGL is sensitive to the Gaussian variance and even small

![Fig. 1. The label propagation process with outlier detection: two-moon dataset (two labels per class) (a) initial data with outliers, (b) 200 iterations, (c) 400 iterations, and (d) 800 iterations. At first, all unlabeled set are viewed as outliers (black points). After several hundred iterations (about 800 iterations), the label information is successfully propagated to the unlabeled set (blue and red points). However, the points in the left-lower corner are still outliers as the label information cannot be propagated to them. Here, the label assigned to an unlabeled sample $x_i$ in $t$ iteration is determined by $\max_{i \in \mathbb{U}} f_{ij}(t)$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)](image1)

![Fig. 2. The label propagation process with noise labels: two-moon dataset (a) initial data with noisy labeled samples (each per class), (b) $\zeta_t = 0$, (c) $\zeta_t = 0.5$, and (d) $\zeta_t = 0.99$. From the simulation results we can see that when setting $\zeta_t = 0$, the noisy labeled sample cannot be erased, as setting $\zeta_t = 0$ means the resulted labels of labeled samples will not change from its initial labels; On the other hand, if we cannot guarantee that the label information are all accurate, the corresponding $\zeta_t$ should be set large than zero in case the label of noisy labeled sample can be updated by those of its neighborhood.)](image2)
variation can make the results dramatically different. However, the proposed label propagation has no such parameter and can be more robust than GGSSL. We can also see that the proposed label propagation can perform well in a wide range of $a$ that is close to $1$. In Fig. 4, we generalize a one-Swiss-roll dataset with one sample labeled in the core. The density of dataset is dense in the core of Swiss-roll while is sparse in the around. Hence the one-Swiss-roll dataset is a varied-density dataset. Similar to Fig. 3, the marker size of each unlabeled data point is proportional to its predicted label. From simulation results we can see that GGSSL and the proposed method can better handle the varied-density dataset than LNP due to the normalization process. The proposed method is slightly superior to GGSSL.

By the above presentation, we design an effective label propagation procedure. But label propagation can only predict the labels of unlabeled samples and cannot predict the labels of new-coming samples. The reason is that label propagation is a transductive learning procedure and cannot learn a model. To solving this problem, a typical method is to regression the soft labels with a linear model as

$$F = V^T X + b^T,$$

where $W$ is the projection matrix and $b$ is a bias term. Then, the label of a new-coming sample $x$ can be handled by $f = V^T x + b^T$.

Next, we will present a new effective method to calculate the projection matrix, which is based on the LDA criterion and arms to preserve more discriminative information.

3. Soft label based Linear Discriminant Analysis

In general, LDA is a supervised method, which means the label of each sample is fixed, i.e. the probability $f_i^j$ of $x_j$ belonging to the $i$th class is either 0 or 1. The total-class, within-class and between-class scatter matrices $S_t$, $S_w$ and $S_b$ are then constructed based on such fixed label (we call it hard label):

$$S_t = \sum_{i=1}^{C} \sum_{x \in x_i} (x - \mu_i)(x - \mu_i)^T,$$

$$S_w = \sum_{i=1}^{C} \sum_{x \in x_i} (x - \mu_i)(x - \mu_i)^T,$$

$$S_b = \sum_{i=1}^{C} (\mu_i - \mu)(\mu_i - \mu)^T,$$

(a) (b) (c) (d)

Fig. 3. The label propagation process with sensitivity analysis: two-cycle dataset (two labels per class) (a) the initial dataset, (b–d) the performances of GGSSL with different Gaussian variance, $\sigma = \tilde{\sigma}/1$, $\tilde{\sigma}/10$, $\tilde{\sigma}/20$, where $\tilde{\sigma}$ is the half of the average distance of all pair-wise neighborhood samples, (e–h) the performances of the proposed method with different $a_u$, (e) $a_u = 0.95$, (f) $a_u = 0.999$, (g) $a_u = 0.999999$, and (h) $a_u = 1$. From simulation results we can see that GGSSL is sensitive to the Gaussian variance $\sigma$ and even small variation can make the results dramatically different. However, the proposed method does not have this parameter, and the proposed method can perform well in a wide range of $a_u$ that is close to $1$.

(a) (b) (c) (d)

Fig. 4. The label propagation process with varied-density dataset: one-Swiss-roll dataset (a) initial dataset with varied-density distribution (one label), (b) GGSSL, (c) LNP, and (d) the proposed method. The density of dataset is dense in the core of Swiss-roll while is sparse in the around. The marker size of each unlabeled data point is proportional to its predicted label. From simulation results we can see that GGSSL and the proposed method can better handle the varied-density dataset than LNP due to the normalization process. The proposed method is slightly superior to GGSSL.
where \( l_i \) and \( \mu_i \) are the number and mean of labeled set in the \( i \)th class, \( \mu \) is the mean of all classes. The goal of LDA is thus to maximize the between-class scatter matrix while minimize the within-class scatter matrix in the reduced subspace as:

\[
J(V) = \max \text{Tr}((V^T S_u V)^{-1} V S_w V)
\]  

(8)

However, since LDA is a supervised method, it needs a large number of labeled samples in order to achieve satisfying classification accuracies [4]. However, labeling large number of samples is time-consuming and impractical. Hence, to enhance the performance of conventional LDA, we consider increasing the labeled samples for training by estimating the labels of unlabeled samples. Fig. 5 illustrates our expectation for the improvement of LDA, in which we generalize a two-line dataset and show the boundary in the 1d reduced space (the boundary is orthogonal to the projection direction).

Note that via label propagation process, we obtain the soft labels of unlabeled samples \( f_{ij} \) (0 ≤ \( f_{ij} \) ≤ 1). These soft labels have key probability information for discriminative learning and can be divided into three categories: (1) for \( f_{ij} \) that is close to 1, \( \forall i \in \{1, 2, \ldots, c\} \), it can be taken into account as this means that \( x_i \) is a newly added labeled sample in the \( i \)th class; (2) for \( f_{ij} \) that is close to 0 when \( i = c + 1 \), it can be neglected as this means that \( x_i \) may belong to the outliers; and (3) for other \( f_{ij} \) (0 < \( f_{ij} \) < 1), it can also be taken into account as it may have some probability information of \( x_i \). Hence, in order to use the estimated labels of unlabeled samples to enhance the performance of LDA, we propose a soft label based LDA, called SL-LDA, by incorporating the soft label into the construction of scatter matrices.

### 3.1. Definition of SL-LDA

In this subsection, we first define the soft label based total-class, within-class and between-class scatter matrices. Specifically, let \( \mu_i, \bar{\mu} \) are the soft means of dataset in the \( i \)th class and in all classes as:

\[
\begin{align*}
\bar{\mu}_i &= \frac{\sum_{j=1}^{l_i} f_{ij} x_j}{\sum_{j=1}^{l_i} f_{ij}} \\
\bar{\mu} &= \frac{\sum_{i=1}^c \sum_{j=1}^{l_i} f_{ij} x_j}{\sum_{i=1}^c \sum_{j=1}^{l_i} f_{ij}}
\end{align*}
\]  

(9)

Then following Eq. (10), we define the soft label based total-class, within-class and between-class scatter matrices \( \bar{S}_i, \bar{S}_u \) and \( \bar{S}_b \) as:

\[
\begin{align*}
\bar{S}_i &= \sum_{i=1}^c \sum_{j=1}^{l_i} f_{ij} (x_j - \bar{\mu} ) (x_j - \bar{\mu})^T = \sum_{i=1}^c \sum_{j=1}^{l_i} f_{ij} (x_j - \bar{\mu}_i ) (x_j - \bar{\mu}_i)^T \\
&= \sum_{i=1}^c \sum_{j=1}^{l_i} f_{ij} E_{x} x_j x_j^T - \sum_{i=1}^c \sum_{j=1}^{l_i} f_{ij} \bar{\mu}_i \bar{\mu}_i^T \\
&= \sum_{i=1}^c \sum_{j=1}^{l_i} \left( \sum_{k=1}^c f_{ik} G_{ik} \right) x_j x_j^T - \sum_{i=1}^c \sum_{k=1}^c \sum_{j=1}^{l_i} f_{ik} E_{x} x_j x_j^T
\end{align*}
\]  

(11)

\[
\begin{align*}
\bar{S}_u &= \sum_{i=1}^c \sum_{j=1}^{l_i} f_{ij} (x_j - \bar{\mu}_i ) (x_j - \bar{\mu}_i)^T \\
&= \sum_{i=1}^c \sum_{j=1}^{l_i} f_{ij} E_{x} x_j x_j^T - \sum_{i=1}^c \sum_{j=1}^{l_i} f_{ij} \bar{\mu}_i \bar{\mu}_i^T \\
&= \sum_{i=1}^c \sum_{j=1}^{l_i} \left( \sum_{k=1}^c f_{ik} G_{ik} \right) x_j x_j^T - \sum_{i=1}^c \sum_{k=1}^c \sum_{j=1}^{l_i} f_{ik} E_{x} x_j x_j^T
\end{align*}
\]  

(12)

\[
\begin{align*}
\bar{S}_b &= \sum_{i=1}^c \sum_{j=1}^{l_i} f_{ij} (\bar{\mu}_i - \bar{\mu} ) (\bar{\mu}_i - \bar{\mu})^T \\
&= \sum_{i=1}^c \sum_{j=1}^{l_i} f_{ij} E_{\bar{\mu}} \bar{\mu}_i \bar{\mu}_i^T - \sum_{i=1}^c \sum_{j=1}^{l_i} f_{ij} \bar{\mu}_i \bar{\mu}_i^T \\
&= \sum_{i=1}^c \sum_{j=1}^{l_i} \left( \sum_{k=1}^c f_{ik} G_{ik} \right) x_j x_j^T - \sum_{i=1}^c \sum_{k=1}^c \sum_{j=1}^{l_i} f_{ik} E_{\bar{\mu}} \bar{\mu}_i \bar{\mu}_i^T
\end{align*}
\]  

(13)

where \( E \in \mathbb{R}^{(c+1)\times(l+u)} \) and \( G \in \mathbb{R}^{c\times c} \) are two diagonal matrices with each element satisfying \( E_{ij} = \sum_{k=1}^c f_{ik} \) and \( G_{ii} = \sum_{j=1}^{l_i} f_{ij} \). \( F_c \in \mathbb{R}^{c\times(l+u)} \) is a matrix formed by the first \( c \) rows of \( F \). \( e \in [1, 1, \ldots, 1] \in \mathbb{R}^{(c+1)\times(u)} \) is a unit vector. The above scatter matrices can be the extensions to the ones in LDA: (1) Following Eqs. (11)–(13), it can be easily noted...
that the equality of scatter matrices still holds, i.e. \( \hat{S}_i = \hat{S}_w + \hat{S}_b \). (2) When the soft label becomes hard label, i.e. \( \eta_j \in \{0, 1\} \), the soft label based scatter matrices will be the conventional ones as in Eq. (7).

(3) By incorporating the soft labels of unlabeled samples into the construction, the soft labeled based scatter matrices can preserve more discriminative information and alleviate the effects of outliers. Hence by maximizing the between-class scatter matrix while minimizing the within-class scatter matrix in the reduced subspace, i.e.

\[
J(V) = \max_V \text{Tr}\left( (V^T \hat{S}_w V)^{-1} V S_b V \right).
\]  

(14)

The optimal solution of SL-LDA can then be obtained by the generalized eigen-value decomposition (GEVD) as:

\[
(\hat{S}_w + \alpha I) - \hat{S}_b V' = V' A \quad \text{or} \quad (\hat{S}_i + \alpha I) - \hat{S}_b V = V' A,
\]  

where \( A \) is the eigenvalue diagonal matrix, \( \alpha \) is a multiply of identity matrix for avoiding the singularity of \( \hat{S}_w \) or \( \hat{S}_i \).

3.2. Relation between SL-LDA and weighted least square

Least square is another popular technique which has been widely used for regression and classification [28]. Let \( T \in \mathbb{R}^{n \times m} \) be a class indicator matrix and \( b \in \mathbb{R}^{1 \times m} \) be the bias term, the goal of least square is then to find a linear model \( t = V^T X + b^T \) by regresssing \( X \) on \( T \). It has been rigorously proved that the conventional LDA can be equivalent to least square given a certain class indicator [18–20]. In this subsection, we further extend this relationship and analyze SL-LDA under a weighted least square framework (W-LS). Based on such relationship, we then, in the next subsection, propose an efficient approach for solving SL-LDA. Let we consider a weighted and regularized least square problem:

\[
J(V, b) = \min_{V, b} \sum_{i=1}^{n} \| V^T x_i + b - t_i \|^2 + \lambda \| V \|^2.
\]  

(16)

For convenience, we write Eq. (16) in matrix form as

\[
J(V, b) = \text{Tr}\left( (V^T X + b^T) E (V^T X + b^T)^T \right)

- 2 \text{Tr}\left( TE(V^T X + b^T)^T \right) + \alpha \text{Tr}(V^T V)
\]

(17)

By setting the derivative w.r.t. \( V \) and \( b \) to zero, we have:

\[
\begin{align*}
XEX^TV + XEe(e^T b - T^T) + \alpha V = 0 \\
Ee^Te b + e (X^T V - T^T) = 0
\end{align*}
\]  

(18)

Following Eq. (18), we can obtain the optimal solution of W-LS as:

\[
V^* = (XLX^T + \alpha I)^{-1} XL^T
\]  

(19)

where \( L^T = E - EXe E^T e \) and following Eq. (11), we have \( XLX^T = \hat{S}_i \). In addition, to establish the connection to SL-LDA, we choose a class indicator as \( T = G^{-1/2} FE^{-1} \), that is for each \( t_j \), it satisfies:

\[
t_j = f_j \left( E_j / F_i \right).
\]  

(20)

It can be easily verified that \( XLT^T TLX^T = \hat{S}_b \) (Corollary 2, see in Appendix 6.2). Hence if we let \( \hat{H}_b = XL^T L \in \mathbb{R}^{p \times p} \), it then follows \( \hat{H}_b \hat{H}_b^T = \hat{S}_b \). The optimal solution of W-LS in Eq. (20) can thus be rewritten as \( V^* = (\hat{S}_i + \alpha I)^{-1} \hat{H}_b \).

We next build the relationship between SL-LDA and W-LS. This relationship is based on the following theorem:

**Theorem 1.** Given two matrix \( A \) and \( B \), then \( AB \) and \( BA \) have the same non-zero eigenvalues. For each non-zero eigenvalue of \( AB \), if the corresponding eigenvector of \( AB \) is \( v \), then the corresponding eigenvector of \( BA \) is \( u = BV \).

The proof of Theorem 1 can be seen in [28]. But in order to make our paper self-contained, we also include the proof in Appendix 6.3. Recall that the optimal solution of SL-LDA is formed by the eigenvectors of \((\hat{S}_i + \alpha I)^{-1} \hat{H}_b \). According to Theorem 1, it has the same non-zero eigenvalues to auxiliary matrix as:

\[
M = \hat{H}_b^T (\hat{S}_i + \alpha I)^{-1} \hat{H}_b \in \mathbb{R}^{c \times c}.
\]  

(21)

According to Theorem 1 again, if \( U \) is the eigenvectors of \( M \) corresponding to non-zero eigenvalues, then \( (\hat{S}_i + \alpha I)^{-1} \hat{H}_b U \) is the eigenvectors of \((\hat{S}_i + \alpha I)^{-1} \hat{H}_b \hat{H}_b^T \). Hence we have the following theorem:

**Theorem 2.** Let \( V_{SL-LDA} \) be the optimal solution of SL-LDA as in Eq. (13), \( V_{W-LS} \) be the optimal solution of weighted least square as in Eq. (18), they then follows: \( V_{SL-LDA} = V_{W-LS} U \), where \( U \) are eigenvectors corresponding to the non-zero eigenvalues of \( M \) defined in Eq. (21).

3.3. An efficient approach for solving SL-LDA

Theorem 2 has given the relationship between SL-LDA and W-LS, which leads to the following theorem:

**Theorem 3.** Given \( V_{SL-LDA} = V_{W-LS} U \) as in Theorem 2, we have

\[
V_{SL-LDA} V_{SL-LDA}^T = V_{W-LS} V_{W-LS}^T.
\]  

Accordingly, for any two samples \( x_i \) and \( x_j \), it follows:

\[
\left\| V_{SL-LDA}^T (x_i - x_j) \right\|^2_F = \left\| V_{W-LS}^T (x_i - x_j) \right\|^2_F,
\]  

(22)

where \( \left\| V^T (x_i - x_j) \right\|_F^2 = (x_i - x_j)^T V^T V (x_i - x_j) \) is Mahalanobis distance between \( x_i \) and \( x_j \).

The proof of Theorem 3 is given in Appendix 6.4. Following Theorem 3, it indicates that when applying a distance-based classifier such as KNN, SL-LDA and W-LS can achieve the same classification results, which leads us to propose a more efficient approach for solving SL-LDA. Specifically, instead of directly solving the GEVD problem as in Eq. (14), we can solve a W-LS problem as in Eq. (16) to obtain the projection matrix of SL-LDA. We next show the efficiency of the approach. Since \( (XLX^T + \alpha I)X = (XLX^T + \alpha I) \), we can rewrite the optimal solution of W-LS in Eq. (19) as:

\[
V^* = (XLX^T + \alpha I)^{-1} XL^T

= (XLX^T + \alpha I)^{-1} (XLX^T + \alpha I)(XLX^T + \alpha I)^{-1} L^T

= (XLX^T + \alpha I)^{-1} (XLX^T + \alpha I)(XLX^T + \alpha I)^{-1} L^T

= (XLX^T + \alpha I)^{-1} L^T.
\]  

(22)

Note that for calculating the inverse of \((XLX^T + \alpha I)\), the computational complex is \(O(1 + \alpha^2)\), while for calculating the inverse of \((XLX^T + \alpha I)\), the computational complex is \(O(D^2)\). Another efficient approach for solving the problem in Eq. (16) is ISQR [40] and the computational complex is \(O(1 + \alpha \delta d)\), where \( \delta \) is the non-zero entries in \( X \). Hence given \( X \) is high-dimensional and sparse, solving the W-LS problem in Eq. (19) can be more efficient than the GEVD method. The basic steps of the efficient approach are given in Table 2.

After we have obtained the projection matrix \( V_{SL-LDA} \), given new coming sample \( x \), its output in the reduced space can be obtained by \( z = V_{SL-LDA}^T x \). Note that the proposed SL-LDA is a linear method. It can be easily extended to the kernel version by using KPCA plus LDA trick [21,22].
Table 2
The algorithm.

<table>
<thead>
<tr>
<th>Input: Data matrix $X \in \mathbb{R}^{d \times \text{D}}$, reduced matrix $d$ and other related parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Output: The projection matrix $V_{\text{SL-LDA}} \in \mathbb{R}^{d \times d}$</td>
</tr>
<tr>
<td>Algorithm:</td>
</tr>
<tr>
<td>1. Construct the neighborhood graph as in Table 1</td>
</tr>
<tr>
<td>2. Perform the label propagation and obtain the soft labels $F$ as in Eq. (5)</td>
</tr>
<tr>
<td>3. Calculate the class indicator using soft labels as in Eq. (20)</td>
</tr>
<tr>
<td>4. Solve the weighted least square regression problem in Eq. (16) and obtain the optimal solution $V_{\text{SL-LDA}}$ in Eq. (22)</td>
</tr>
<tr>
<td>5. Output $V_{\text{SL-LDA}} = V_{\text{WLS}}$</td>
</tr>
</tbody>
</table>

4. Simulations

In this section, we evaluate our algorithms with two synthetic dataset and several UCI datasets and real-world datasets. For the synthetic datasets, we evaluate our algorithm using two-moon and two-Swiss-roll datasets. For other datasets, we focus on solving the classification problems based on 8 UCI datasets and 5 real-world datasets which are all benchmark datasets. Furthermore, we compare our algorithm with other state-of-the-art supervised and semi-supervised algorithms. For the comparative study, we randomly split each dataset into training set and test set. We also randomly select samples from the training set to form labeled and unlabeled set. All the training sets are preliminarily processed with a PCA operator to eliminate the null space before performing dimensionality reduction [3]. All algorithms used the training set in the output reduced space to train a nearest neighborhood classifier for evaluating the accuracy of test set.

4.1. Examples for synthetic datasets

In this toy example, we generate a dataset with two classes, each follows a half-moon distribution. In each class, two samples are selected as labeled set and the remaining are as unlabeled set. Since the distribution of two-moon dataset is nonlinear, to handle this problem, we first perform KPCA to the two-moon dataset; we then use the output in the full-rank KPCA to train the linear methods [21,22]. Fig. 6 shows the gray images of decision surfaces and boundaries obtained by LDA, SDA and SL-LDA. The gray value of each pixel and boundary represents the same means as in two-moon dataset. All algorithms used the training set in the output reduced space to train a nearest neighborhood classifier for evaluating the accuracy of test set.

4.2. Classification

For classification problem, we use 8 UCI datasets and 5 real-world datasets to evaluate the performance of methods. The detailed information of dataset can be shown in Table 3. For each dataset, we randomly select $l$ samples from each class as labeled set and $u$ samples as unlabeled set. The test set is then formed by the selected or all remaining samples. The data partitioning for each dataset is also given in Table 3. We also show some sampled images of real-world datasets as in Fig. 8.

Next, we compare our method with other supervised and semi-supervised dimensionality reduction methods. These methods include RLDA [23], RLS [28], CCA [15], SDA [9], LapLDA [10],
LapRLS [11], SSCCA [12] and the proposed SL-LDA. Note that the original CCA [15] only deals with binary classification, therefore, we choose a variant version of CCA and SSCCA by using the \(c/C_0\) label coding for solving multi-class classification problem [16]. We also compare our method with three label propagation methods, i.e. SSKDE [41], First-order LNP [7] and Second-order LNP [8], and we denote them as LNP1 and LNP2, respectively. The simulation settings are as follows: For all the methods that have Tikhonov regularized term, we set the same value of parameter \(\alpha\) in each dataset. For all the semi-supervised methods that have manifold regularized term, we set the same value of parameter \(\alpha_m\) and the same neighborhood size for constructing the neighborhood graph in each dataset. For the proposed SL-LDA, there are three parameters, i.e. the balancing parameter \(\gamma\) in Eqs. (3) and (4), the neighborhood size and the Tikhonov regularized parameter \(\alpha_t\). For the first parameter, we use 5-fold cross validation to determine the value of \(\gamma\) and the candidate set for \(\gamma\) is \(\{10^{-2}, 10^{-4}, 10^{-6}, 10^{-8}, 10^{-10}, 10^{-12}\}\); for other two parameters, we choose the same values as in other methods in each dataset.

The average accuracies over 20 random splits with the above parameters for each dataset are shown in Table 4. From the simulation results, we can obtain the following observation: (1) the semi-supervised dimensionality reduction methods are better than

<table>
<thead>
<tr>
<th>Dataset</th>
<th>#Samples (N)</th>
<th>#Dim (D)</th>
<th>#Class (c)</th>
<th>Balance</th>
<th>#Labeled (l)</th>
<th>#Unlabeled (u)</th>
<th>#Test (t)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heart-statlog</td>
<td>270</td>
<td>13</td>
<td>2</td>
<td>0.8</td>
<td>10</td>
<td>30</td>
<td>40</td>
</tr>
<tr>
<td>Ionosphere</td>
<td>351</td>
<td>34</td>
<td>2</td>
<td>0.56</td>
<td>10</td>
<td>30</td>
<td>40</td>
</tr>
<tr>
<td>Iris</td>
<td>150</td>
<td>4</td>
<td>3</td>
<td>1</td>
<td>3</td>
<td>30</td>
<td>Remains</td>
</tr>
<tr>
<td>Wine</td>
<td>178</td>
<td>13</td>
<td>3</td>
<td>0.6761</td>
<td>5</td>
<td>30</td>
<td>Remains</td>
</tr>
<tr>
<td>Waveform</td>
<td>5000</td>
<td>40</td>
<td>3</td>
<td>0.9770</td>
<td>50</td>
<td>200</td>
<td>400</td>
</tr>
<tr>
<td>Balance</td>
<td>625</td>
<td>1</td>
<td>2</td>
<td>0.56</td>
<td>10</td>
<td>30</td>
<td>Remains</td>
</tr>
<tr>
<td>SCCTS</td>
<td>600</td>
<td>60</td>
<td>6</td>
<td>0.9208</td>
<td>20</td>
<td>100</td>
<td>300</td>
</tr>
<tr>
<td>Letters</td>
<td>20000</td>
<td>16</td>
<td>26</td>
<td>0.9028</td>
<td>20</td>
<td>100</td>
<td>300</td>
</tr>
<tr>
<td>UMNIST [29]</td>
<td>1012</td>
<td>1024</td>
<td>20</td>
<td>0.2857</td>
<td>5</td>
<td>10</td>
<td>Remains</td>
</tr>
<tr>
<td>CMU-PIE [30]</td>
<td>11,554</td>
<td>1024</td>
<td>68</td>
<td>0.9647</td>
<td>10</td>
<td>20</td>
<td>30</td>
</tr>
<tr>
<td>COIL100 [31]</td>
<td>7200</td>
<td>1024</td>
<td>100</td>
<td>1</td>
<td>10</td>
<td>20</td>
<td>30</td>
</tr>
<tr>
<td>USPS [32]</td>
<td>9298</td>
<td>256</td>
<td>10</td>
<td>0.4559</td>
<td>50</td>
<td>100</td>
<td>100</td>
</tr>
<tr>
<td>MNIST [33]</td>
<td>60,000</td>
<td>784</td>
<td>10</td>
<td>1</td>
<td>50</td>
<td>200</td>
<td>200</td>
</tr>
</tbody>
</table>

**Fig. 7.** Gray images and decision boundaries learned by LDA, SDA and SL-LDA: two-swiss-roll dataset (a) LDA, (b) SDA, (c) SL-LDA.

**Fig. 8.** Sample images of real-world datasets: (a) UMNIST dataset, (b) COIL100 dataset, (c) CMU-PIE dataset, and (d) USPS hand-written dataset (3) MNIST hand-written dataset.
Average classification accuracy over 20 random splits on test set of different datasets (values in the brackets show the standard derivations), the bold values show the best classification accuracy.

Table 4

<table>
<thead>
<tr>
<th>Dataset</th>
<th>INN</th>
<th>RLDA</th>
<th>RLS</th>
<th>CCA</th>
<th>SDA</th>
<th>Lap-LDA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heart-statlog</td>
<td>77.03(6.4009)</td>
<td>77.03(6.4009)</td>
<td>79.26(5.8195)</td>
<td>77.43(5.362)</td>
<td>77.03(7.4629)</td>
<td>78.36(5.3534)</td>
</tr>
<tr>
<td>Ionosphere</td>
<td>82.74(2.4746)</td>
<td>82.74(2.4746)</td>
<td>83.91(7.0933)</td>
<td>83.21(5.4567)</td>
<td>83.01(5.7119)</td>
<td>83.48(5.9061)</td>
</tr>
<tr>
<td>Iris</td>
<td>92.52(4.1487)</td>
<td>92.47(3.3031)</td>
<td>93.19(4.4413)</td>
<td>93.19(4.4413)</td>
<td>92.61(7.4305)</td>
<td>92.95(4.7472)</td>
</tr>
<tr>
<td>Wine</td>
<td>84.70(7.1389)</td>
<td>85.53(5.9412)</td>
<td>86.27(9.7381)</td>
<td>83.48(9.2860)</td>
<td>85.01(7.6858)</td>
<td>87.52(9.7381)</td>
</tr>
<tr>
<td>Waveform</td>
<td>80.20(1.4196)</td>
<td>80.20(1.5244)</td>
<td>83.03(1.2290)</td>
<td>81.08(1.3661)</td>
<td>80.52(1.7963)</td>
<td>81.06(1.1760)</td>
</tr>
<tr>
<td>Balance</td>
<td>78.68(4.7760)</td>
<td>78.60(4.7963)</td>
<td>80.84(3.6405)</td>
<td>78.37(7.5101)</td>
<td>78.91(9.6630)</td>
<td>79.29(5.9573)</td>
</tr>
<tr>
<td>SCCTS</td>
<td>81.82(5.1475)</td>
<td>81.54(5.672)</td>
<td>84.87(4.5657)</td>
<td>83.12(4.6326)</td>
<td>82.87(4.5657)</td>
<td>84.68(4.7007)</td>
</tr>
<tr>
<td>Letters</td>
<td>76.77(3.8048)</td>
<td>76.32(1.5494)</td>
<td>80.56(2.0911)</td>
<td>79.66(1.6543)</td>
<td>75.04(1.6860)</td>
<td>76.93(1.6772)</td>
</tr>
<tr>
<td>UMNIST</td>
<td>92.55(1.7377)</td>
<td>91.74(3.9560)</td>
<td>94.53(3.1298)</td>
<td>93.99(3.4524)</td>
<td>92.27(3.6835)</td>
<td>94.15(3.0718)</td>
</tr>
<tr>
<td>CMU-PIE</td>
<td>73.80(5.1688)</td>
<td>73.83(1.8069)</td>
<td>76.99(1.922)</td>
<td>73.29(1.4002)</td>
<td>72.09(2.2878)</td>
<td>74.19(1.6265)</td>
</tr>
<tr>
<td>COIL100</td>
<td>91.63(3.6100)</td>
<td>91.05(1.4217)</td>
<td>94.98(5.1933)</td>
<td>93.87(1.4544)</td>
<td>92.80(1.4633)</td>
<td>93.46(1.4656)</td>
</tr>
<tr>
<td>USPS</td>
<td>93.16(1.4704)</td>
<td>93.05(4.6982)</td>
<td>95.05(7.9530)</td>
<td>93.92(7.3221)</td>
<td>93.88(9.6360)</td>
<td>94.36(5.7851)</td>
</tr>
<tr>
<td>MNIST</td>
<td>91.98(0.9945)</td>
<td>91.67(1.1207)</td>
<td>92.91(1.2554)</td>
<td>91.03(6.2124)</td>
<td>90.52(6.2561)</td>
<td>91.16(6.9008)</td>
</tr>
</tbody>
</table>

The improvement can even achieve almost 5% in Letters and SCCTS datasets. This enhancement is believed to be true due to the reason that by taking into account the soft labels of unlabeled samples, which are obtained from label propagation process, the SL-LDA can preserve more discriminative information. While in SDA and Lap-RLS, these algorithms can only use unlabeled samples in a simple manner to preserve the manifold structure but cannot preserve more discriminative structure. (3) We also compare the proposed SL-LDA with two popular label propagation methods LNP1 and LNP2. Note that in these two methods, the predicted label of a test sample is obtained by reconstructing from the labels of its neighborhoods, where the reconstruction weights are calculated following Eq. [2] [9,10]. From Table 4, we can observe the proposed SL-LDA can achieve better performance than LNP1 and LNP2 by about 2% in the dataset. The reason for it is that SL-LDA is an inductive learning model and can learn a linear model. As the projection matrix can be obtained by incorporating the soft labels into the linear model, the new sample can be processed with the projection matrix and preserves more discriminative information. While in LNP, the label of a new sample can only be predicted using the label information of its neighborhoods. (4) In order to further show further show the superiority of the proposed method, we conduct more simulations for classification by varying the number of labeled samples. Table 5 shows the average...
classification accuracy over 20 random splits on test set of COIL100 and CMU-PIE datasets with 2, 4, 6, 8, 10 samples per class selected as labeled set. Other parameters are set the same as above. From the simulation results we can observe that the proposed method can deliver better classification accuracy than those obtained by other supervised and semi-supervised methods by about 2–3% in most cases of fixed labeled number. The superiority can even achieve 5% when only 2 samples per class are selected as labeled set both in COIL100 and CMU-PIE datasets. This indicates that the proposed method is more suitable for the case when few samples are labeled. Since labeling large number of samples is time-consuming and costly, our proposed method is more practical than other semi-supervised methods in real-world application.

In addition, in order to further verify the above analysis, we demonstrate the sub-manifold visualization of the proposed method and compare it with other state-of-the-art methods such as PCA, LPP, LDA, SDA and Lap-RLS. In this study, we randomly choose the first five classes of COIL100 dataset for simulation. In this dataset, we randomly select 2 samples per class as labeled set and 20 samples per class as unlabeled set. The remaining 50 samples per class are selected as test set. Hence the number of labeled samples is quite few compared with that unlabeled samples. Our goal is to visualize the test set in the 2d subspace by projecting the test set on the 2d projection matrix learned by different methods. The simulation results are shown in Fig. 9. From the simulation results, we can see that the proposed method can achieve the best performance than other methods, in a way that the sub-manifold of each object is closely conglomerated, while those belonging to different objects are clearly separated. The main reason is that the proposed method can provide more discriminative information compared with other state-of-art methods.

Fig. 9. Visualization performance of different methods: COIL100 (first 5 classes, each color represents a object). (a) PCA. (b) LPP. (c) LDA. (d) SDA. (e) Lap-RLS. (f) SL-LDA. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)
4.3. Application for content-based image retrieval

The goal of content-based image retrieval (CBIR) is to retrieve relevant images from the dataset by using semantic similarity between imagery contents [47]. In this subsection, we evaluate the CBIR performance of the proposed SL-LDA and compare it with those of RLDA, RLS, SDA and Lap-RLS based on a large-scale image dataset, called Corel dataset [34]. In this dataset, we choose a subset from the dataset for simulation, which contains 50 concepts with each having 100 images [38]. Some samples of Corel dataset is shown in Fig. 10.

In CBIR, low-level image representation is of great importance to characterize the feature information for each image. Current visual features include color, texture, shape and different combinations of the aforesaid methods. In this work, we combine a 48-dimensionality color histogram in HSV space [35] and a 48-dimensionality color texture moment (CTM) [36,37] to represent each image. For color histogram, in order to avoid the computation burden, we only consider 16 bins to calculate the histograms for each of the three HSV channels. Specifically, the histogram of the hue channel is calculated as follows:

\[ h_{H_i} = n_{H_i} / n_T, \quad i = 1, 2, \ldots, 16, \]  

where \( H_i \) is the \( i \)th quantized level of the hue channel, \( n_{H_i} \) is the total number of pixels with their hue values in that level and \( n_T \) is the total number of pixels in the images. The histograms of the Saturation (\( h_{S_i} \)) and Value (\( h_{V_i} \)) channels can also be calculated as in Eq. (23). For color texture moment (CTM), local Fourier transform (LFT) is adopted as a texture representation strategy and eight characteristic maps are derived for describing different aspects of co-occurrence relations of images pixels in the three channels of the color space (\( SV \cos H, SV \sin H \) and \( V \)). Then, CTM calculates the first and second moments of these maps to represent the natural color image pixel distribution.

We next design an automatic feedback strategy to model the retrieval process. For each query image submitted by the user, the system retrieves and ranks the images in the database. The top 10 images are then selected as the feedback images and their feedback information can be used for re-ranking. It is worth noting that for the sake of feedbacks, users can either mark them as relevant or irrelevant ones or provide their class labels. Since our method is more proper to the class labels, we focus on the latter ones. The above automatic feedback strategy is performed for four iterations for each query image.

But in a real image retrieval system, the query image is usually not in the image database. To simulate this environment, we divide the dataset into two non-overlapping subsets, i.e. the first one includes 4500 images used for training the system and the second one includes 500 images used as query images. For the image retrieval results, we use accuracy-scope curve [37] to evaluate the performance of different methods under a fixed iteration. The scope is the number of top-ranked images presented to the user and the accuracy is the ratio between the numbers of relevant images to the given scope. The accuracy-scope curve describes the accuracies with varied scopes and thus gives an overall performance evaluation of the algorithms. We then fix the scope by 10, 20, 50 and evaluate the performance of different methods with varied iterations.

Fig. 11 shows the average accuracy-scope curves of different methods at Iteration 2 and Iteration 4. Fig. 12 shows the average accuracies with varied iterations under the fixed scope 10, 20 and 50. From Figs. 11 and 12, we observe: (1) All of the five
methods i.e. RLDA, RLS, SDA, Lap-RLS and SL-LDA can achieve better performance when iteration increases from 2 to 4, which indicates that the user-provided feedbacks are helpful to enhance the performance of image retrieval. (2) The semi-supervised dimensionality reduction methods are better than their corresponding supervised methods due to the utilization of the unlabeled set, i.e. SDA and Lap-RLS can achieve approximately 4–5% improvements to RLDA and RLS in most cases. (3) By iteratively adding the user’s feedbacks, the proposed SL-LDA can achieve much better retrieval results than other methods over the entire scope and iterations. The improvement can be the largest by approximately 5% compared to other methods when the scope is fixed to 10, 20 in the last round of feedback, shown in Fig. 12. This is demonstrated by incorporating the soft labels of unlabeled samples into learning, SL-LDA can preserve more discriminative information than other methods hence enhance the performance of image retrieval.

5. Conclusions

In this paper, we propose a new semi-supervised dimensionality reduction method by incorporating the soft label of unlabeled samples into learning. The soft labels can be obtained by a specially designed label propagation, which is based on a new local reconstruction graph with symmetrization and normalization. The symmetrization can add more connections of a sample with its neighborhoods and the normalization can handle with the case when the density of dataset varies dramatically. As an extension to the conventional LDA, the proposed SL-LDA can preserve more discriminative information embedded in the soft labels than other state-of-the-art methods, which can be beneficial to the performance of classification. In addition, motivated by the equivalence between the conventional LDA and least square problem [18–20], we further extend this relationship and analyze SL-LDA under a weighted least square problem (W-LS). We then propose a more efficient approach for solving SL-LDA. Theoretical analysis and extensive simulations show the effectiveness of our algorithm. The results in simulations demonstrate that our proposed algorithm can achieve great superiority compared with other existing methods.

Appendix A

A.1. Proof of Corollary 1

Denote \( e_n = [1, 1, \ldots, 1] \in \mathbb{R}^{1 \times n} \) as the unit vector, to prove Corollary 1, we need to prove \( e_{c+1}F = e_{\nu u} \). Actually, we first have \( e_{\nu u}P = e_{\nu u} \) and \( e_{c+1}Y = e_{\nu u} \), then:

\[
\begin{align*}
& e_{c+1}Y = e_{\nu u} \\
& e_{\nu u}P = e_{\nu u} \\
& => e_{c+1}Y_{\beta} = e_{\nu u}(I - PL) \\
& => e_{c+1}Y_{\beta}(I - PL)^{-1} = e_{\nu u} \\
& => e_{c+1}F = e_{\nu u}
\end{align*}
\]

Fig. 12. Average accuracies at scope 10, 20, 50 of different methods: corel dataset (a) scope 10, (b) scope 20, and (c) scope 50.
To prove Corollary 2, we first rewrite $S_b$ as:

$$S_b = \sum_{i=1}^{n} \sum_{j=1}^{n} f_{ij} (\tilde{\mu}_i - \tilde{\mu}) (\tilde{\mu}_j - \tilde{\mu})^T$$

$$= (X^T G^{-1} \frac{X X^T e_i}{e_i e_i} G (X^T G^{-1} \frac{X X^T e_i}{e_i e_i}))^T$$

$$= (X^T G^{-1} \frac{X X^T e_i}{e_i e_i} G (X^T G^{-1} \frac{X X^T e_i}{e_i e_i}))^T$$

$$= (X^T G^{-1} \frac{X X^T e_i}{e_i e_i} G (X^T G^{-1} \frac{X X^T e_i}{e_i e_i}))^T$$

The second equation holds as $[\mu_1, \mu_2, \ldots, \mu_c] = X^T G^{-1}$ and $\tilde{\mu} = X^T G^{-1} e_i e_i$. The third equation holds as $e_i G = e_i e_i$. Since $L_s = E - E e_i e_i e_i^T$ and $T = G^{-1/2} F^{-1}$, following Eq. (25), we have $X_{Ls}^T T L X_{Ls}^T = S_b$.

A.3. Proof of Theorem 1

Let $\tau$ be a nonzero eigenvalue of $AB$ and $v$ be its corresponding eigenvector, it follows $ABv = \tau v$. Hence, we have that $AB$ is the eigenvector of $BA$ corresponding to the same nonzero eigenvalue $\tau$. On the other hand, if $\tau$ is a nonzero eigenvalue of $BA$ and $u$ be its corresponding eigenvector, similarly, we can also verify that $Au$ is an eigenvector of $AB$ corresponding to the same nonzero eigenvalue $\tau$. As analyzed above, $AB$ and $BA$ have the same nonzero eigenvalues, and for each nonzero eigenvalue, if the corresponding eigenvector of $AB$ is $v$, then the corresponding eigenvector of $BA$ is $u = Bv$.

A.4. Proof of Theorem 3

Let the rank of auxiliary matrix $M$ be $q$, then, $U \in C^{n \times q}$ is an orthonormal matrix formed by the eigenvectors corresponding to the $q$ nonzero eigenvalues of $M$. Since $q \leq c$, there exists an orthogonal matrix $U_i \in C^{n \times q}$ formed by the eigenvectors corresponding to the $c-q$ zero eigenvalues of $M$, which satisfies $U_i U_i^T = O$ and follows:

$$U_i^T M U_i = O = U_i^T H_i (S_i + \lambda I)^{-1} H_i U_i = O = H_i U_i = O$$

The third equation holds as $(S_i + \lambda I)$ is positive definite. Recall that the optimal solution of $W-LS$ is $(S_i + \lambda I)^{-1} H_i$, it follows $V_{W-LS} U_i = (S_i + \lambda I)^{-1} H_i U_i = O$. In addition, let $\tilde{U} \in C^{n \times c}$ be the orthogonal matrix formed by $\tilde{U} = [U, U_i^T]$, it follows $\tilde{U}^T \tilde{U} = \tilde{U} \tilde{U}^T = I$. Hence we have

$$V_{W-LS} \tilde{U} = V_{W-LS} U = V_{W-LS} U_i U_i^T V_{W-LS}$$

$$= V_{W-LS} U_i U_i^T + V_{W-LS} U_i U_i^T V_{W-LS}$$

$$= V_{W-LS} U_i U_i^T$$

Since $V_{W-LS} \tilde{U} = V_{W-LS} U_i$, following Eq. (27), we thus prove $V_{W-LS} \tilde{U} = V_{W-LS} U_i$, which is the desired result.


Zhao Zhang received the Ph.D. degree in computer engineering from the Department of Electronic Engineering, City University of Hong Kong in July, 2013; He received his B.Sc. and M.Sc. degrees from Nanjing Forestry University, China, both in computer science and technology, in 2008 and 2010, respectively. He is currently an associate professor with the School of Computer Science and Technology, Soochow University, China. He was a visiting research engineer at the Learning & Vision Research Group at National University of Singapore, mentored by Prof. Shuicheng Yan, from Feb. to May 2012. He then visited the National Laboratory of Pattern Recognition (NLPR) at Chinese Academy of Sciences (CAS), mentored by Prof. Cheng-Lin Liu, from Sep. to Dec. 2012. His current interests are pattern recognition, machine learning & data mining, and computer vision.

Zhao Zhang received the Ph.D. degree in computer engineering from the Department of Electronic Engineering, City University of Hong Kong in July, 2013; He received his B.Sc. and M.Sc. degrees from Nanjing Forestry University, China, both in computer science and technology, in 2008 and 2010, respectively. He is currently an associate professor with the School of Computer Science and Technology, Soochow University, China. He was a visiting research engineer at the Learning & Vision Research Group at National University of Singapore, mentored by Prof. Shuicheng Yan, from Feb. to May 2012. He then visited the National Laboratory of Pattern Recognition (NLPR) at Chinese Academy of Sciences (CAS), mentored by Prof. Cheng-Lin Liu, from Sep. to Dec. 2012. His current interests are pattern recognition, machine learning & data mining, and computer vision.

Mingbo Zhao received the Ph.D. degree in computer engineering from the Department of Electronic Engineering, City University of Hong Kong in Jan. 2013. He also received his B.Sc. and M.Sc. degrees from Shanxi University, Shanxi, P.R. China in 2005 and 2008, respectively. He is currently a Senior Research Assistant at the Department of Electronic Engineering, City University of Hong Kong. Dr. Zhao’s current interests include data mining, machine learning, pattern recognition, and their applications.

Zhao Zhang received the Ph.D. degree in computer engineering from the Department of Electronic Engineering, City University of Hong Kong in July, 2013; He

Bing Li received the Ph.D. degree in computer engineering from the Department of Electronic Engineering, City University of Hong Kong in December 2013. She received her B.Sc. degree and M.Sc. degrees from the College of Computer Science, Northeast Normal University, Jilin, China in 2006 and 2009, respectively. Her current interests include data mining, machine learning, pattern recognition, and their applications.