Supervised Discriminative Sparse PCA with Adaptive Neighbors for Dimensionality Reduction

Zhenhua Shi, Dongrui Wu
School of Artificial Intelligence and Automation
Huazhong University of Science and Technology
Wuhan, China
Email: {zhenhuashi, drwu}@hust.edu.cn

Yu-Kai Wang, Chin-Teng Lin
Faculty of Engineering and Information Technology
University of Technology
Sydney, Australia
Email: {YuKai.Wang, Chin-Teng.Lin}@uts.edu.au

***Abstract*—Dimensionality reduction is an important operation in information visualization, feature extraction, clustering, regression, and classification, especially for processing noisy high dimensional data. However, most existing approaches preserve either the global or the local structure of the data, but not both. Approaches that preserve only the global data structure, such as principal component analysis (PCA), are usually sensitive to outliers. Approaches that preserve only the local data structure, such as locality preserving projections, are usually unsupervised (and hence cannot use label information) and uses a fixed similarity graph. We propose a novel linear dimensionality reduction approach, supervised discriminative sparse PCA with adaptive neighbors (SDSPCAAN), to integrate neighborhood-free supervised discriminative sparse PCA and projected clustering with adaptive neighbors. As a result, both global and local data structures, as well as the label information, are used for better dimensionality reduction. Classification experiments on nine high-dimensional datasets validated the effectiveness and robustness of our proposed SDSPCAAN.**

***Index Terms*—Principal component analysis, adaptive neighbors, linear dimensionality reduction**

I. Introduction

Defined as the process of projecting high-dimensional data into a low-dimensional subspace, dimensionality reduction is an important operation in information visualization, feature extraction, clustering, regression, and classification [1]. Linear dimensionality reduction approaches are frequently used for processing noisy high dimensional data, due to their low computational cost and simple geometric interpretations [2]. We divide linear dimensionality reduction approaches into three groups: neighborhood-free approaches, fixed-neighborhood based approaches, and adaptive-neighborhood based approaches.

Neighborhood-free dimensionality reduction approaches require no neighborhood information. For instance, as one of the first dimensionality reduction approaches in the literature, principal component analysis (PCA) [3] reduces the dimensionality of data by projecting them onto orthogonal directions of high variances. The traditional PCA is unsupervised, and hence cannot make use of label information. To extend it to supervised learning, supervised PCA [4] maximizes the Hilbert-Schmidt independence between the labels and the orthogonally projected data. To incorporate PCA with Laplacian eigenmaps [5], graph-Laplacian PCA (gLPCA) [6] adds a weighted Laplacian embedding loss to a variant formulation of PCA (vPCA) for closed-form solution. To extend vPCA to supervised sparse learning, supervised discriminative sparse PCA (SDSPCA) [7] adds a label-related term and a sparse $L_{2,1}$ regularization [8] to vPCA. As one of the most widely used supervised dimensionality reduction approaches, linear discriminant analysis (LDA) [9] seeks for directions of high separation between different classes. Robust LDA [10] reformulates the traditional LDA by minimizing the within class covariance and reducing the influence of outliers via $L_{2,1}$ regularization [8]. Self-weighted adaptive locality discriminant analysis [11] reformulates the traditional LDA by minimizing the within class covariance in a pairwise expression and adding $L_{2,1}$ regularization [8]. To utilize multi-view data in dimensionality reduction, canonical correlations analysis (CCA) [12] jointly maps data from two sources into the same subspace and maximizes the correlation between the projected data. To extend CCA to supervised learning, discriminative CCA [13] maximizes the within-class correlation between the projected data. Discriminative sparse generalized CCA [14] further adds sparsity to the discriminative CCA, and also extends it to more than two views.

To preserve neighborhood information in dimensionality reduction, fixed-neighborhood based dimensionality reduction approaches usually assign a fixed similarity graph of data via heat kernel, nearest neighbors, reconstruction weights [15], or local scaling [16]. For example, as a linear approximation of the Laplacian eigenmaps [5], locality preserving projections (LPP) [17] forces the projection of the connected points in the similarity graph to stay as close as possible. As a linear analogy to locally linear embedding [18], neighborhood preserving embedding (NPE) [15] represents each data point as a linear combination of its neighbors, and then forces the projection of the points to preserve this reconstruction relationship. Local Fisher discriminant analysis [19] reformulates LDA in a pairwise expression and assigns a weight to each pairwise distance via a similarity graph. Similarly, locality preserving CCA [20] reformulates CCA in a pairwise expression and adds weights to the pairwise distances via a similarity graph.

Different from fixed-neighborhood based dimensionality reduction approaches that rely on a fixed similarity graph, adaptive-neighborhood based dimensionality reduction ap-

proaches use an adaptive similarity graph. For instance, projected clustering with adaptive neighbors (PCAN) [21] allows for adaptive neighbors and is able to construct a predefined number of clusters via graphs [22], [23]. To extend PCAN to multi-view learning, multi-view feature extraction with structured graph [24] minimizes the differences between the adaptive similarity graph of all views and the fixed similarity graph of each view. To use PCAN for feature selection, structured optimal graph feature selection [25] adds a weighted $L_{2,p}$ regularization of orthogonal projection matrix to the objective function of PCAN. Projective unsupervised flexible embedding with optimal graph [26] combines PCAN and ridge regression for image and video representation. To extend PCAN to supervised learning, simultaneously learning neighborship and projection (SLNP) [27] learns class-wise similarity graphs and the projection matrix simultaneously.

In summary, neighborhood-free dimensionality reduction approaches that preserve the global data structure are usually more sensitive to outliers than neighborhood based approaches that preserve the local structure. To remedy this, SDSPCA [7] applies $L_{2,1}$ regularization [8] to reduce the influence of outliers. Adaptive-neighborhood based dimensionality reduction approaches that learn the similarity graph and projection matrix simultaneously are usually advantageous to fixed-neighborhood based approaches. However, most existing adaptive-neighborhood based dimensionality reduction approaches are unsupervised, leading to unsatisfactory classification performance. To remedy this, SLNP [27] extends PCAN to supervised learning, but it needs adequate data from each class for class-wise similarity graph construction.

This paper proposes supervised discriminative sparse PCA with adaptive neighbors (SDSPCAAN) for dimensionality reduction, which extends PCAN to supervised learning, following the approach in [28], and integrates it with the state-of-the-art SDSPCA.

The remainder of this paper is organized as follows: Section II introduces PCA, SDSPCA, PCAN and our proposed SDSPCAAN. Section III describes the nine high-dimensional datasets and our experimental results. Section IV draws conclusions.

II. Methods

In this paper, matrices and vectors are denoted by uppercase and lowercase boldface letters, respectively. Other important notations are summarized in Table I.

The training data matrix is $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^T \in \mathbb{R}^{n\times d}$, where n is the number of training samples, and d the feature dimensionality. Without loss of generality, we assume $\mathbf{X}$ is mean-centered, i.e., $\mathbf{1}_{1\times n}\mathbf{X} = \mathbf{0}_{1\times d}$. The one-hot coding label matrix of $\mathbf{X}$ is $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_n]^T \in \mathbb{R}^{n\times c}$, where c is the number of classes.

A. Principal Component Analysis (PCA)

PCA [3] reduces the dimensionality of data by projecting them onto orthogonal directions of high variances, which usually have higher signal-to-noise ratios than the directions of low variances [29]. Mathematically, it solves the following optimization problem:

$$\max_{\mathbf{W}} \mathrm{Tr}(\mathbf{W}^T\mathbf{X}^T\mathbf{X}\mathbf{W}) \quad \text{s.t. } \mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}_{\mathrm{k}}. \tag{1}$$

The optimal $\mathbf{W} \in \mathbb{R}^{d\times k}$ is formed by the k leading eigenvectors of $\mathbf{X}^T\mathbf{X}$, which also minimizes the Frobenius norm of the residual matrix:

$$\min_{\mathbf{W}} \|\mathbf{X} - \mathbf{X}\mathbf{W}\mathbf{W}^T\|_F^2 \quad \text{s.t. } \mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{I}_{\mathrm{k}}. \tag{2}$$

A variant formulation of PCA (vPCA) used in gLPCA [6] and SDSPCA [7] optimizes

$$\begin{aligned} &\min_{\mathbf{Q}} \|\mathbf{X} - \mathbf{Q}\mathbf{Q}^T\mathbf{X}\|_F^2 = \max_{\mathbf{Q}} \mathrm{Tr}(\mathbf{Q}^T\mathbf{X}\mathbf{X}^T\mathbf{Q}) \\ &\text{s.t. } \mathbf{Q}^{\mathrm{T}}\mathbf{Q} = \mathbf{I}_{\mathrm{k}}. \end{aligned} \tag{3}$$

The optimal $\mathbf{Q} \in \mathbb{R}^{n\times k}$ is formed by the k leading eigenvectors of $\mathbf{X}\mathbf{X}^T$. The projection matrix is then $\mathbf{W} = \mathbf{X}^T\mathbf{Q}$.

Let $\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{R}^T$ be the singular value decomposition of $\mathbf{X}$, where $\mathbf{U} \in \mathbb{R}^{n\times n}$ and $\mathbf{R} \in \mathbb{R}^{d\times d}$ are orthogonal, and $\mathbf{\Sigma} \in \mathbb{R}^{n\times d}$ is a diagonal matrix with non-negative singular values in descending order on the diagonal. Then, we can calculate the optimal projection matrix for PCA as $\mathbf{W}_{\text{PCA}} = \mathbf{R}_{1:k}$, the optimal projection matrix for vPCA as $\mathbf{W}_{\text{vPCA}} = \mathbf{R}_{1:k}\mathbf{\Sigma}_{1:k}$, where $\mathbf{R}_{1:k} \in \mathbb{R}^{d\times k}$ consists of the first k columns of $\mathbf{R}$, and $\mathbf{\Sigma}_{1:k} \in \mathbb{R}^{k\times k}$ is a diagonal matrix of the first k leading singular values (arranged in descending order). Thus,

$$\mathbf{W}_{\text{vPCA}} = \mathbf{W}_{\text{PCA}}\mathbf{\Sigma}_{1:k}. \tag{4}$$

vPCA is equivalent to PCA if we scale each column of $\mathbf{X}\mathbf{W}$ by the column standard deviation, which is a common practice in machine learning.

TABLE I
Notations used in this paper.

Notation	Meaning
$\mathbf{S} \in \mathbb{R}^{n\times n}$	The pairwise similarity matrix of $\mathbf{X}$
S_{ij}	The (i,j)th element of matrix $\mathbf{S}$
$\mathbf{S}^T$	The transpose of matrix $\mathbf{S}$
$\mathrm{Tr}(\mathbf{S})$	The trace of a square matrix $\mathbf{S}$
$\|\mathbf{Q}\|_F$	The Frobenius norm of matrix $\mathbf{Q}$
$\|\mathbf{Q}\|_{1,1}$	The $L_{1,1}$ norm of matrix $\mathbf{Q} \in \mathbb{R}^{n\times k}$, i.e., $\|\mathbf{Q}\|_{1,1} = \sum_{i=1}^{n}\left(\sum_{j=1}^{k}\|Q_{ij}\|\right)$
$\|\mathbf{Q}\|_{2,1}$	The $L_{2,1}$ norm of matrix $\mathbf{Q} \in \mathbb{R}^{n\times k}$, i.e., $\|\mathbf{Q}\|_{2,1} = \sum_{i=1}^{n}\left(\sum_{j=1}^{k}Q_{ij}^2\right)^{\frac{1}{2}}$
$\|\mathbf{s}_i\|_2$	The L_2 norm of vector $\mathbf{s}_i$
$\mathrm{diag}(\mathbf{v})$	The square diagonal matrix with the elements of vector $\mathbf{v}$ on the main diagonal
$\mathbf{I}_k$	A $k \times k$ identity matrix
$\mathbf{I}_{d\times k}$	A $d \times k$ matrix with ones in the main diagonal and zeros elsewhere
$\mathbf{1}_{n\times k}$	A $n \times k$ all-one matrix
$\mathbf{0}_{n\times k}$	A $n \times k$ all-zero matrix
$\mathbf{W} \in \mathbb{R}^{d\times k}$	The subspace projection matrix of $\mathbf{X}$, where k is the subspace dimensionality
$\mathbf{Q} \in \mathbb{R}^{n\times k}$	The auxiliary matrix of $\mathbf{W}$
$\mathbf{L} \in \mathbb{R}^{n\times n}$	The Laplacian matrix of $\mathbf{S} \in \mathbb{R}^{n\times n}$, i.e., $\mathbf{L} = \mathrm{diag}(\mathbf{S}\mathbf{1}_{n\times 1}) - \mathbf{S}$

B. Supervised Discriminative Sparse PCA (SDSPCA)

SDSPCA [7] extends vPCA to supervised sparse linear dimensionality reduction, by integrating data information, label information and sparse regularization. The projection matrix $\mathbf{W}$ is obtained by

$$\begin{aligned}\min_{\mathbf{W},\mathbf{G},\mathbf{Q}} &\|\mathbf{X}-\mathbf{Q}\mathbf{W}^T\|_F^2+\alpha\|\mathbf{Y}-\mathbf{Q}\mathbf{G}^T\|_F^2+\beta\|\mathbf{Q}\|_{2,1}\\ &\text{s.t. } \mathbf{Q}^T\mathbf{Q}=\mathbf{I}_k,\end{aligned} \tag{5}$$

where $\mathbf{G}\in\mathbb{R}^{c\times k}$, and α and β are scaling weights. Alternative optimization can be used to solve (5), as follows.

When $\mathbf{G}$ and $\mathbf{Q}$ are fixed, setting the partial derivative of (5) w.r.t. $\mathbf{W}$ to zero yields

$$\mathbf{W}=\mathbf{X}^T\mathbf{Q}. \tag{6}$$

When $\mathbf{Q}$ and $\mathbf{W}$ are fixed, similarly, we have

$$\mathbf{G}=\mathbf{Y}^T\mathbf{Q}. \tag{7}$$

When $\mathbf{W}$ and $\mathbf{G}$ are fixed, substituting (6) and (7) into (5) yields

$$\begin{aligned}&\min_{\mathbf{Q}}\|\mathbf{X}-\mathbf{Q}\mathbf{Q}^T\mathbf{X}\|_F^2+\alpha\|\mathbf{Y}-\mathbf{Q}\mathbf{Q}^T\mathbf{Y}\|_F^2+\beta\|\mathbf{Q}\|_{2,1}\\ &=\min_{\mathbf{Q}}-\operatorname{Tr}(\mathbf{Q}^T\mathbf{X}\mathbf{X}^T\mathbf{Q})-\alpha\operatorname{Tr}(\mathbf{Q}^T\mathbf{Y}\mathbf{Y}^T\mathbf{Q})+\beta\operatorname{Tr}(\mathbf{Q}^T\mathbf{D}\mathbf{Q})\\ &=\min_{\mathbf{Q}}\operatorname{Tr}\left(\mathbf{Q}^T(-\mathbf{X}\mathbf{X}^T-\alpha\mathbf{Y}\mathbf{Y}^T+\beta\mathbf{D})\mathbf{Q}\right)\\ &\text{s.t. } \mathbf{Q}^T\mathbf{Q}=\mathbf{I}_k.\end{aligned} \tag{8}$$

The optimal $\mathbf{Q}$ is formed by the k trailing eigenvectors of $\mathbf{Z}=-\mathbf{X}\mathbf{X}^T-\alpha\mathbf{Y}\mathbf{Y}^T+\beta\mathbf{D}$, where $\mathbf{D}\in\mathbb{R}^{n\times n}$ is a diagonal matrix with the ith diagonal element be [8]

$$D_{ii}=\frac{1}{2\sqrt{\sum_{j=1}^k Q_{ij}^2+\epsilon}}, \tag{9}$$

where ϵ is a small positive constant to avoid dividing by zero.

The pseudocode for optimizing SDSPCA is shown in Algorithm 1.

SDSPCA integrates data information and label information elegantly to seek for a discriminative low-dimensional subspace, and it does not involve any matrix inversion operation. Additionally, the sparse constraint of $\mathbf{Q}$ makes it robust to outliers. However, SDSPCA fails to utilize the neighborhood information.

C. Projected Clustering with Adaptive Neighbors (PCAN)

Different from SDSPCA, which ignores neighborhood information, PCAN [21] learns the projection matrix and neighbourhood relations simultaneously to perform unsupervised linear dimensionality reduction. Its projection matrix $\mathbf{W}$ is obtained by:

$$\begin{aligned}&\min_{\mathbf{W},\mathbf{F},\mathbf{S}}\sum_{i,j=1}^n\left(\|\mathbf{W}^T\mathbf{x}_i-\mathbf{W}^T\mathbf{x}_j\|_2^2S_{ij}+\gamma_iS_{ij}^2+\lambda\|\mathbf{f}_i-\mathbf{f}_j\|_2^2S_{ij}\right)\\ &=\min_{\mathbf{W},\mathbf{F},\mathbf{S}}2\operatorname{Tr}\left(\mathbf{W}^T\mathbf{X}^T\mathbf{L}\mathbf{X}\mathbf{W}\right)+\operatorname{Tr}(\mathbf{S}^T\mathbf{\Gamma}\mathbf{S})+2\lambda\operatorname{Tr}\left(\mathbf{F}^T\mathbf{L}\mathbf{F}\right)\\ &\text{s.t. } \mathbf{S}\mathbf{1}_{n\times1}=\mathbf{1}_{n\times1},\mathbf{S}\geq0,\mathbf{W}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{W}=\mathbf{I}_{\mathrm{k}},\mathbf{F}^{\mathrm{T}}\mathbf{F}=\mathbf{I}_{\mathrm{c}},\end{aligned} \tag{10}$$

Algorithm 1 The SDSPCA training algorithm [7].

Input: $\mathbf{X}\in\mathbb{R}^{n\times d}$, the training data matrix;
$\mathbf{Y}\in\mathbb{R}^{n\times c}$, the one-hot coding label matrix of $\mathbf{X}$;
k, the subspace dimensionality;
α and β, the scaling weights;
ϵ, a small positive constant;
tol, the tolerance;
T, the maximum number of iterations.
Output: Projection matrix $\mathbf{W}\in\mathbb{R}^{d\times k}$.
Initialize $\mathbf{Z}_0=-\mathbf{X}\mathbf{X}^T-\alpha\mathbf{Y}\mathbf{Y}^T$, $\mathbf{D}=\mathbf{I}_n$, and $\mathbf{Q}_0=\mathbf{0}_{n\times k}$;
for $t=1$ to T **do**
  Calculate $\mathbf{Z}=\mathbf{Z}_0+\beta\mathbf{D}$;
  Construct $\mathbf{Q}$ by the k trailing eigenvectors of $\mathbf{Z}$;
  if $\|\mathbf{Q}-\mathbf{Q}_0\|_{1,1}<tol$ **then**
    break;
  end if
  Update $\mathbf{D}$ using (9);
  $\mathbf{Q}_0=\mathbf{Q}$;
end for
Calculate $\mathbf{W}$ using (6).

where $\mathbf{S}\in\mathbb{R}^{n\times n}$ is the pairwise similarity matrix, $\mathbf{L}=\operatorname{diag}(\mathbf{S}\mathbf{1}_{n\times1})-\mathbf{S}\in\mathbb{R}^{n\times n}$ is the Laplacian matrix, $\mathbf{\Gamma}\in\mathbb{R}^{n\times n}$ is a diagonal matrix with the ith diagonal element being γ_i, λ is a scaling weight, $\mathbf{F}\in\mathbb{R}^{n\times c}$ is an auxiliary matrix for minimizing the c smallest eigenvalues of $\mathbf{L}$. Since the multiplicity of 0 as an eigenvalue of $\mathbf{L}$ is equal to the number of connected components of $\mathbf{S}$ [22], [23], a proper λ will lead to exactly c clusters indicated by $\mathbf{S}$. Alternative optimization can be used to solve (10), as follows.

When $\mathbf{F}$ and $\mathbf{S}$ are fixed, (10) becomes

$$\min_{\mathbf{W}}\operatorname{Tr}\left(\mathbf{W}^T\mathbf{X}^T\mathbf{L}\mathbf{X}\mathbf{W}\right)\quad\text{s.t. }\mathbf{W}^{\mathrm{T}}\mathbf{X}^{\mathrm{T}}\mathbf{X}\mathbf{W}=\mathbf{I}_{\mathrm{k}}. \tag{11}$$

The optimal $\mathbf{W}$ is formed by the k trailing eigenvectors of $(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{L}\mathbf{X}$.

When $\mathbf{S}$ and $\mathbf{W}$ are fixed, (10) becomes

$$\min_{\mathbf{F}}\operatorname{Tr}\left(\mathbf{F}^T\mathbf{L}\mathbf{F}\right)\quad\text{s.t. }\mathbf{F}^{\mathrm{T}}\mathbf{F}=\mathbf{I}_{\mathrm{k}}. \tag{12}$$

The optimal $\mathbf{F}$ is formed by the c trailing eigenvectors of $\mathbf{L}$.

When $\mathbf{W}$ and $\mathbf{F}$ are fixed, (10) becomes

$$\begin{aligned}&\min_{\mathbf{S}}\sum_{i,j=1}^n\left(\|\mathbf{W}^T\mathbf{x}_i-\mathbf{W}^T\mathbf{x}_j\|_2^2S_{ij}+\gamma_iS_{ij}^2+\lambda\|\mathbf{f}_i-\mathbf{f}_j\|_2^2S_{ij}\right)\\ &\text{s.t. }\mathbf{S}\mathbf{1}_{n\times1}=\mathbf{1}_{n\times1},\mathbf{S}\geq0.\end{aligned} \tag{13}$$

Let $d_{ij}^x=\|\mathbf{W}^T\mathbf{x}_i-\mathbf{W}^T\mathbf{x}_j\|_2^2$, $d_{ij}^f=\|\mathbf{f}_i-\mathbf{f}_j\|_2^2$, $\mathbf{d}_i\in\mathbb{R}^{n\times1}$ be a vector with the j-th element being $d_{ij}=d_{ij}^x+\lambda d_{ij}^f$, and $\mathbf{s}_i$ be the transpose of the i-th row of $\mathbf{S}$. Then, (13) can be written in a vector form as

$$\min_{\mathbf{s}_i}\gamma_i\|\mathbf{s}_i\|_2^2+\mathbf{d}_i^T\mathbf{s}_i\qquad\text{s.t. }\mathbf{s}_{\mathrm{i}}^{\mathrm{T}}\mathbf{1}_{\mathrm{n}\times1}=1,\mathbf{s}_{\mathrm{i}}\geq0. \tag{14}$$

Differentiating the Lagrangian $\mathcal{L}(\mathbf{s}_i, \eta, \mathbf{b}) = \gamma_i \|\mathbf{s}_i\|_2^2 + \mathbf{d}_i^T \mathbf{s}_i - \eta(\mathbf{s}_i^T \mathbf{1}_{n\times 1} - 1) - \mathbf{b}^T \mathbf{s}_i$ corresponding to (14) with respect to $\mathbf{s}_i$ and setting it to zero leads to

$$\mathbf{s}_i = \frac{1}{2\gamma_i}(-\mathbf{d}_i + \eta \mathbf{1}_{n\times 1} + \mathbf{b}), \tag{15}$$

where $\eta, \mathbf{b} \geq 0$ are the Lagrangian multipliers. According to the Karush-Kuhn-Tucker (KKT) complementary condition, we have

$$\mathbf{b}^T \mathbf{s}_i = 0. \tag{16}$$

Then we can express the optimal $\mathbf{s}_i$ as

$$\mathbf{s}_i = \frac{1}{2\gamma_i}(-\mathbf{d}_i + \eta \mathbf{1}_{n\times 1})_+, \tag{17}$$

where $(x)_+ = \max\{0, x\}$.

Without loss of generality, suppose $d_{i1}, \ldots, d_{in}$ are ordered in ascending order. If the optimal $\mathbf{s}_i$ has only m nonzero elements, then according to (14) and(17), we have

$$\begin{cases} \sum_{j=1}^{m} \frac{1}{2\gamma_i}(-d_{ij} + \eta) = 1, \\ \frac{1}{2\gamma_i}(-d_{im} + \eta) > 0, \\ \frac{1}{2\gamma_i}(-d_{i,m+1} + \eta) \leq 0, \end{cases} \tag{18}$$

which lead to

$$\eta = \frac{1}{m}(2\gamma_i + \sum_{j=1}^{m} d_{ij}), \tag{19}$$

and

$$\frac{m}{2} d_{im} - \frac{1}{2}\sum_{j=1}^{m} d_{ij} < \gamma_i \leq \frac{m}{2} d_{i,m+1} - \frac{1}{2}\sum_{j=1}^{m} d_{ij}. \tag{20}$$

Substituting (17) and (19) into the objective function in (14) yields

$$\begin{aligned} &\gamma_i \|\mathbf{s}_i\|_2^2 + \mathbf{d}_i^T \mathbf{s}_i \\ =&\gamma_i \sum_{j=1}^{m} \left(\frac{1}{2\gamma_i}(-d_{ij} + \eta) \right)^2 + \sum_{j=1}^{m} \frac{d_{ij}}{2\gamma_i}(-d_{ij} + \eta) \\ =&\frac{\gamma_i}{m} + \frac{1}{4\gamma_i m}\left((\sum_{j=1}^{m} d_{ij})^2 - m\sum_{j=1}^{m} d_{ij}^2 \right) + \frac{1}{m}\sum_{j=1}^{m} d_{ij}, \end{aligned} \tag{21}$$

where $(\sum_{j=1}^{m} d_{ij})^2 \leq m \sum_{j=1}^{m} d_{ij}^2$ according to the Cauchy-Buniakowsky-Schwarz inequality. So, the objective function increases monotonously with respect to γ_i.

Taking γ_i as a dual variable, according to (20) and(21), the optimal γ_i can be expressed as

$$\gamma_i = \frac{m}{2} d_{i,m+1} - \frac{1}{2}\sum_{j=1}^{m} d_{ij}. \tag{22}$$

Substituting (19) and (22) into (17) yields the optimal $\mathbf{s}_i$, which can be expressed as

$$\mathbf{s}_i = \left(\frac{d_{i,m+1} - \mathbf{d}_i}{m d_{i,m+1} - \sum_{j=1}^{m} d_{ij} + \epsilon} \right)_+, \tag{23}$$

where ϵ is a small positive constant to avoid dividing by zero.

The detailed optimization routine of PCAN is shown in Algorithm 2.

Algorithm 2 The PCAN training algorithm [21].

Input: $\mathbf{X} \in \mathbb{R}^{n\times d}$, the training data matrix;
k, subspace dimensionality;
c, number of clusters;
m, number of nearest neighbors;
ϵ, a small positive constant;
tol, absolute tolerance;
T, maximum number of iterations.

Output: Projection matrix $\mathbf{W} \in \mathbb{R}^{d\times k}$.

Initialize d_{ij} as $\|\mathbf{x}_i - \mathbf{x}_j\|_2^2$, and $\mathbf{S}$ using (23);
$\lambda = 1$;
for $t = 1$ to T **do**
  $\mathbf{S} = (\mathbf{S} + \mathbf{S}^T)/2$;
  $\mathbf{L} = \mathrm{diag}(\mathbf{S}\mathbf{1}_n) - \mathbf{S}$;
  Construct $\mathbf{W}$ by the k trailing eigenvectors of $(\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{L}\mathbf{X}$;
  Construct $\mathbf{F}$ by the c trailing eigenvectors of $\mathbf{L}$;
  Calculate the $c+1$ smallest eigenvalues of $\mathbf{L}$ in ascending order as $e_1, e_2, \ldots, e_{c+1}$;
  if $\sum_{i=1}^{c} e_i > tol$ **then**
    $\lambda = 2\lambda$;
  else if $\sum_{i=1}^{c+1} e_i < tol$ **then**
    $\lambda = \lambda/2$;
  else
    break;
  end if
  Calculate $d_{ij} = \|\mathbf{W}^T\mathbf{x}_i - \mathbf{W}^T\mathbf{x}_j\|_2^2 + \lambda\|\mathbf{f}_i - \mathbf{f}_j\|_2^2$;
  Update $\mathbf{S}$ using (23);
end for

After being updated by (23), $\mathbf{S}$ is replaced by $(\mathbf{S} + \mathbf{S}^T)/2$ for symmetry. The sum of the c smallest eigenvalues of $\mathbf{L}$, $\sum_{i=1}^{c} e_i$, is used to restrict the rank of $\mathbf{L}$ since the eigenvalues of the Laplacian matrix $\mathbf{L}$ are non-negative. When $\sum_{i=1}^{c} e_i > tol$, the rank of $\mathbf{L}$ is larger than $n - c$, and the number of connected components of $\mathbf{S}$ is smaller than c [22], [23], so λ is multiplied by 2 to strengthen the impact of $\mathrm{Tr}\left(\mathbf{F}^T\mathbf{L}\mathbf{F}\right)$. When $\sum_{i=1}^{c+1} e_i < tol$, the opposite is performed. We did not use a global γ, whose value is the average of all γ_i, as in [21]; instead, we used the optimal γ_i to update the i-th row of $\mathbf{S}$ ($\mathbf{s}_i^T$) for faster convergence.

PCAN simultaneously learns the projection matrix and neighbourhood relations to perform dimensionality reduction and construct exactly c clusters based on $\mathbf{S}$. However, $\mathbf{X}^T\mathbf{X}$ can be singular, especially for high-dimensional data, so the construction of $\mathbf{W}$ may not be accurate. In addition, PCAN fails to utilize the label information for better discrimination.

D. Supervised Discriminative Sparse PCA with Adaptive Neighbors (SDSPCAAN)

To take the advantages of SDSPCA and PCAN and avoid their limitations, we propose SDSPCAAN to integrate SDSPCA and PCAN together. Its projection matrix is obtained by

$$\begin{aligned}&\min_{\mathbf{Q},\mathbf{S}} \|\mathbf{X}-\mathbf{Q}\mathbf{Q}^T\mathbf{X}\|_F^2+\alpha\|\mathbf{Y}-\mathbf{Q}\mathbf{Q}^T\mathbf{Y}\|_F^2+\beta\|\mathbf{Q}\|_{2,1}\\&+\frac{1}{2}\delta\left[2\,\mathrm{Tr}(\mathbf{Q}^T\mathbf{X}\mathbf{X}^T\mathbf{L}\mathbf{X}\mathbf{X}^T\mathbf{Q})+\mathrm{Tr}(\mathbf{S}^T\mathbf{\Gamma}\mathbf{S})+2\lambda\,\mathrm{Tr}(\mathbf{Y}^T\mathbf{L}\mathbf{Y})\right]\\&\text{s.t. }\mathbf{Q}^{\mathrm{T}}\mathbf{Q}=\mathbf{I}_{\mathrm{k}},\mathbf{S}\mathbf{1}_{\mathrm{n}\times 1}=\mathbf{1}_{\mathrm{n}\times 1},\mathbf{S}\geq 0,\end{aligned}\tag{24}$$

where $\delta>0$ is a scaling weight.

We construct SDSPCA based on (8), and PCAN based on (10). We replace $\mathbf{W}$ in PCAN with $\mathbf{X}^T\mathbf{Q}$ based on (6) to avoid matrix inversion error, and $\mathbf{F}$ in PCAN with $\mathbf{Y}$ to utilize label information, following [28]. Alternative optimization can be used to solve (24), as follows.

When $\mathbf{S}$ is fixed, (24) becomes

$$\begin{aligned}&\min_{\mathbf{Q}}\|\mathbf{X}-\mathbf{Q}\mathbf{Q}^T\mathbf{X}\|_F^2+\alpha\|\mathbf{Y}-\mathbf{Q}\mathbf{Q}^T\mathbf{Y}\|_F^2+\beta\|\mathbf{Q}\|_{2,1}\\&\quad+\delta\,\mathrm{Tr}\left(\mathbf{Q}^T\mathbf{X}\mathbf{X}^T\mathbf{L}\mathbf{X}\mathbf{X}^T\mathbf{Q}\right)\\&=\min_{\mathbf{Q}}-\mathrm{Tr}(\mathbf{Q}^T\mathbf{X}\mathbf{X}^T\mathbf{Q})-\alpha\,\mathrm{Tr}(\mathbf{Q}^T\mathbf{Y}\mathbf{Y}^T\mathbf{Q})+\beta\,\mathrm{Tr}(\mathbf{Q}^T\mathbf{D}\mathbf{Q})\\&\quad+\delta\,\mathrm{Tr}\left(\mathbf{Q}^T\mathbf{X}\mathbf{X}^T\mathbf{L}\mathbf{X}\mathbf{X}^T\mathbf{Q}\right)\\&=\min_{\mathbf{Q}}\mathrm{Tr}\left(\mathbf{Q}^T(-\mathbf{X}\mathbf{X}^T-\alpha\mathbf{Y}\mathbf{Y}^T+\beta\mathbf{D}+\delta\mathbf{X}\mathbf{X}^T\mathbf{L}\mathbf{X}\mathbf{X}^T)\mathbf{Q}\right)\\&\text{s.t. }\mathbf{Q}^T\mathbf{Q}=\mathbf{I}_k.\end{aligned}\tag{25}$$

The optimal $\mathbf{Q}$ is formed by the k trailing eigenvectors of $\mathbf{Z}=-\mathbf{X}\mathbf{X}^T-\alpha\mathbf{Y}\mathbf{Y}^T+\beta\mathbf{D}+\delta\mathbf{X}\mathbf{X}^T\mathbf{L}\mathbf{X}\mathbf{X}^T$, where $\mathbf{D}\in\mathbb{R}^{n\times n}$ is a diagonal matrix expressed in (9).

When $\mathbf{Q}$ is fixed, (24) becomes

$$\begin{aligned}&\min_{\mathbf{S}}2\,\mathrm{Tr}\left(\mathbf{Q}^T\mathbf{X}\mathbf{X}^T\mathbf{L}\mathbf{X}\mathbf{X}^T\mathbf{Q}\right)+\mathrm{Tr}(\mathbf{S}^T\mathbf{\Gamma}\mathbf{S})+2\lambda\,\mathrm{Tr}\left(\mathbf{Y}^T\mathbf{L}\mathbf{Y}\right)\\&\text{s.t. }\mathbf{S}\mathbf{1}_{n\times 1}=\mathbf{1}_{n\times 1},\mathbf{S}\geq 0.\end{aligned}\tag{26}$$

Same as in PCAN, the optimal $\mathbf{s}_i$ can be expressed as

$$\mathbf{s}_i=\left(\frac{d_{i,m+1}-\mathbf{d}_i}{md_{i,m+1}-\sum_{j=1}^m d_{ij}+\epsilon}\right)_+,\tag{27}$$

where $d_{ij}=\|\mathbf{Q}^T\mathbf{X}\mathbf{x}_i-\mathbf{Q}^T\mathbf{X}\mathbf{x}_j\|_2^2+\lambda\|\mathbf{y}_i-\mathbf{y}_j\|_2^2$, and ϵ is a small positive constant to avoid dividing by zero.

The detailed optimization routine of SDSPCAAN is given in Algorithm 3. When δ in SDSPCAAN is set to zero, it degrades to SDSPCA (Section II-B). When δ in SDSPCAAN is set to infinity, SDSPCAAN degrades to supervised PCAN (SPCAN). When fixing the similarity graph $\mathbf{S}$ at its initial value, SDSPCAAN degrades to SDSPCA-LPP, a combination of SDSPCA and LPP.

Algorithm 3 The proposed SDSPCAAN training algorithm.

Input: $\mathbf{X}\in\mathbb{R}^{n\times d}$, the training data matrix;
$\mathbf{Y}\in\mathbb{R}^{n\times c}$, the corresponding one-hot coding label matrix of $\mathbf{X}$;
k, subspace dimensionality;
m, number of nearest neighbors;
α, β and δ, scaling weights;
ϵ, small positive constant;
tol, absolute tolerance;
T, maximum number of iterations.
Output: Projection matrix $\mathbf{W}\in\mathbb{R}^{d\times k}$.

$\mathbf{Z}_0=-\mathbf{X}\mathbf{X}^T-\alpha\mathbf{Y}\mathbf{Y}^T$, $\mathbf{D}=\mathbf{I}_n$, $\mathbf{Q}_0=\mathbf{0}_{n\times k}$, $\lambda=1$;
Initialize d_{ij} as $\|\mathbf{x}_i-\mathbf{x}_j\|_2^2$, and $\mathbf{S}$ using (27);
for $t=1$ to T **do**
  $\mathbf{S}=(\mathbf{S}+\mathbf{S}^T)/2$;
  $\mathbf{L}=\mathrm{diag}(\mathbf{S}\mathbf{1}_n)-\mathbf{S}$;
  $\mathbf{Z}=\mathbf{Z}_0+\beta\mathbf{D}+\delta\mathbf{X}\mathbf{X}^T\mathbf{L}\mathbf{X}\mathbf{X}^T$;
  Construct $\mathbf{Q}$ by the k trailing eigenvectors of $\mathbf{Z}$;
  Calculate the $c+1$ smallest eigenvalues of $\mathbf{L}$ in ascending order as $e_1,e_2,\ldots,e_{c+1}$;
  if $\sum_{i=1}^c e_i>tol$ **then**
    $\lambda=2\lambda$;
  else if $\sum_{i=1}^{c+1} e_i<tol$ **then**
    $\lambda=\lambda/2$;
  else if $\|\mathbf{Q}-\mathbf{Q}_0\|_{1,1}<tol$ **then**
    break;
  end if
  Update $\mathbf{D}$ using (9);
  Calculate $d_{ij}=\|\mathbf{Q}^T\mathbf{X}\mathbf{x}_i-\mathbf{Q}^T\mathbf{X}\mathbf{x}_j\|_2^2+\lambda\|\mathbf{y}_i-\mathbf{y}_j\|_2^2$;
  Update $\mathbf{S}$ using (27);
  $\mathbf{Q}_0=\mathbf{Q}$;
end for
$\mathbf{W}=\mathbf{X}^T\mathbf{Q}$.

III. Experiments

Experiments on nine real-world datasets are performed in this section to demonstrate the performance of the proposed SDSPCAAN.

A. Datasets

The following nine high-dimensional benchmark classification datasets were used in the experiments:

1) Musk1 [30], which consists of 476 conformations belonging to 207 musk molecules and 269 non-musk molecules. Each conformation is described by 166 features.
2) MSRA25, which contains 1,799 front-face images of 12 distinct subjects with different background and illumination conditions. In our experiment, all images were resized to 16×16.
3) Palm, which includes 2,000 images of palm prints from 100 distinct individuals. In our experiment, all images were down-sampled to 16×16.
4) USPST, which contains 2,007 images of handwritten digits from 0 to 9. This dataset was sampled from the original USPS dataset. In our experiment, all images were down-sampled to 16×16.
5) Isolet [30], which contains 1,560 samples from 30 subjects who spoke the name of each alphabet letter

twice. Each sample is described by 617 features.

6) Yale, which contains 165 gray-scale face images of 15 distinct subjects. Each subject has 11 images with different facial expressions or configurations: center-light, with glasses, happy, left-light, without glasses, normal, right-light, sad, sleepy, surprised, and wink. In our experiment, all images were down-sampled to 32×32.
7) ORL, which contains 400 face images from 40 distinct subjects. Each subject has 10 images with varying shooting time, lighting, facial expressions and facial details. In our experiment, all images were down-sampled to 32×32.
8) COIL20 [31], which contains 1,440 gray-scale images from 20 distinct objects. Each object has 72 images taken at pose interval of 5 degrees. In our experiment, all images were down-sampled to 32×32.
9) YaleB, which contains 2,414 near frontal face images from 38 distinct subjects. Each subject has 64 images under different illuminations. In our experiment, all images were cropped and resized to 32×32.

A summary of the nine datasets is shown in Table II.

TABLE II
SUMMARY OF THE NINE HIGH-DIMENSIONAL CLASSIFICATION DATASETS.

Dataset	No. of Samples	No. of Features	No. of Classes
Musk1[1]	476	166	2
MSRA25[2]	1,799	256	12
Palm[2]	2,000	256	100
USPST[2]	2,007	256	10
Isolet[3]	1,560	617	2
Yale[4]	165	1,024	15
ORL[5]	400	1,024	40
COIL20[6]	1,440	1,024	20
YaleB[7]	2,414	1,024	38

[1] http://archive.ics.uci.edu/ml/datasets/musk+(version+1)
[2] http://www.escience.cn/people/fpnie/index.html
[3] http://archive.ics.uci.edu/ml/datasets/ISOLET
[4] http://www.cad.zju.edu.cn/home/dengcai/Data/Yale/Yale_32x32.mat
[5] http://www.cad.zju.edu.cn/home/dengcai/Data/ORL/ORL_32x32.mat
[6] http://www.cad.zju.edu.cn/home/dengcai/Data/COIL20/COIL20.mat
[7] http://www.cad.zju.edu.cn/home/dengcai/Data/YaleB/YaleB_32x32.mat

B. Algorithms

We compared the performance of eight different dimensionality reduction approaches:

1) *Baseline*, which uses $\mathbf{I}_k$ as the projection matrix, i.e., the first k features are used in classification.
2) *PCA*, the most popular unsupervised dimensionality reduction approach, introduced in Section II-A.
3) *JPCDA* [32], which unifies PCA and LDA. It first performs PCA to reduce the feature dimensionality to k, then LDA to further reduce the feature dimensionality to c. The two steps are optimized simultaneously.
4) *SDSPCA* [7], a supervised sparse extension of PCA, introduced in Section II-B. It was implemented by setting δ in SDSPCAAN to zero.
5) *SLNP* [27], a supervised version of PCAN. SLNP learns the class-wise similarity graphs and the projection matrix simultaneously.
6) *SPCAN*, a supervised PCAN, implemented by removing the first three terms in the objective function of (24).
7) *SDSPCA-LPP*, a combination of SDSPCA and LPP, implemented by fixing the similarity graph $\mathbf{S}$ in SDSPCAAN at its initial value.
8) *SDSPCAAN*, our proposed algorithm, introduced in Section II-D.

A comparison of the eight algorithms is shown in Table III.

TABLE III
COMPARISON OF THE EIGHT ALGORITHMS.

Algorithm	Preserve Global Data Structure	Preserve Local Data Structure	Adaptive Neighborhood	Supervised
Baseline	–	–	–	–
PCA	✓	–	–	–
JPCDA	✓	–	–	✓
SDSPCA	✓	–	–	✓
SLNP	–	✓	✓	✓
SPCAN	–	✓	✓	✓
SDSPCA-LPP	✓	✓	–	✓
SDSPCAAN	✓	✓	✓	✓

C. Experimental Setup

We used 1-nearest neighbor and standardized Euclidean distance (so that PCA and vPCA are equivalent) as the base classifier. The subspace dimensionality k was set to c for LDA in JPCDA, and tuned from $\{10, 20, \ldots, 100\}$ for other approaches, with the constraint that k must be no larger than n and d, and no smaller than c. We set $\epsilon = 2^{-52} = 2.2204 \times 10^{-16}$ (*eps* in Matlab), $tol = 10^{-3}$, and $T = 500$ for all iterative approaches (JPCDA, SDSPCA, SLNP, SPCAN, SDSPCA-LPP and SDSPCAAN). For JPCDA, η was tuned from $\{0.01, 0.1, 1, 10, 100\}$. For SDSPCA, SDSPCA-LPP and SDSPCAAN, α and β were tuned from $\{0.01, 0.1, 1, 10, 100\} \cdot \mathrm{Tr}(\mathbf{X}\mathbf{X}^T)/\mathrm{Tr}(\mathbf{Y}\mathbf{Y}^T)$ and $\{0.01, 0.1, 1, 10, 100\} \cdot \mathrm{Tr}(\mathbf{X}\mathbf{X}^T)/\mathrm{Tr}(\mathbf{D})$, respectively. For SDSPCA-LPP and SDSPCAAN, δ was also tuned from $\{0.01, 0.1, 1, 10, 100\} \cdot \mathrm{Tr}(\mathbf{X}\mathbf{X}^T)/\mathrm{Tr}(\mathbf{X}\mathbf{X}^T\mathbf{L}\mathbf{X}\mathbf{X}^T)$.

We randomly partitioned each dataset into three subsets: 20% for training, 40% for validation, and the remaining 40% for test. We repeated this process 10 times for each of the nine datasets, and recorded the test balanced classification accuracies (BCAs; the average of the per-class classification accuracies) [33] as our performance measure.

D. Experimental Results

The mean and standard deviation of the test BCAs in 10 runs are shown in Table IV. The largest value (best performance) on each dataset is marked in bold. Note that SLNP cannot run on datasets Palm, Yale and ORL, because there are no adequate samples in each class.

Table IV shows that:

TABLE IV
MEAN AND STANDARD DEVIATION OF BCAS(%) OF THE EIGHT APPROACHES ON THE NINE DATASETS.

Dataset	Baseline	PCA	JPCDA	SDSPCA	SLNP	SPCAN	SDSPCA-LPP	SDSPCAAN
Musk1	**79.47±2.35**	76.03±3.75	74.06±4.14	77.32±1.89	66.78±4.47	71.33±4.49	78.04±2.59	77.31±3.24
MSRA25	98.15±0.61	98.89±0.45	99.73±0.14	99.77±0.20	99.76±0.17	43.36±11.54	99.68±0.32	**99.78±0.13**
Palm	90.58±1.65	96.71±0.80	**97.36±0.97**	96.94±0.92	–	61.45±3.15	97.14±1.01	96.88±0.92
USPST	68.06±1.69	85.28±1.45	86.04±1.48	87.76±0.81	81.66±1.71	17.91±2.49	**87.89±1.22**	87.52±1.00
Isolet	66.01±1.95	83.98±2.01	81.19±1.56	84.26±1.56	**91.78±1.39**	75.73±2.00	86.29±2.01	86.66±2.40
Yale	23.57±6.31	40.44±4.55	44.67±5.87	39.83±3.55	–	47.16±4.82	38.95±3.74	**48.24±4.40**
ORL	30.56±4.00	58.32±5.19	61.85±3.76	63.35±4.21	–	**69.70±5.71**	63.72±4.71	69.69±5.78
COIL20	62.29±1.58	93.21±1.63	94.90±0.98	94.42±1.54	92.78±0.96	92.77±1.51	95.80±0.59	**97.12±0.96**
YaleB	50.07±1.70	78.94±0.96	**83.71±1.37**	78.84±0.96	80.86±1.06	78.93±1.80	79.88±1.55	80.08±1.13
Average	63.20±1.02	79.09±0.67	80.39±1.17	80.27±0.63	–	62.04±1.94	80.82±0.65	**82.59±0.92**

1) Our proposed SDSPCAAN performed the best on three out of the nine datasets, and close to the best on the remaining six datasets. On average, SDSPCAAN performed the best.
2) SDSPCAAN outperformed SDSPCA on six out of the nine datasets, and slightly under-performed SDSPCA on the remaining three datasets. These results suggested that the features learnt by SDSPCA may not be adequate since it did not utilize the local data structure information, which is particularly evident on the Yale and ORL datasets.
3) SDSPCAAN outperformed SPCAN on eight of the nine datasets, and slightly under-performed SPCAN on the remaining one dataset. These results suggested that the futures learnt by SPCAN may not be adequate, since it did not utilize the global data structure information, which is particularly evident on the MSRA25 and USPST datasets.
4) SDSPCAN outperformed SDSPCA-LPP on six of the nine datasets, and under-performed it on the remaining three datasets. These indicated that the fixed similarity graph in SDSPCA-LPP may lead to suboptimal results, and our proposed SDSPCAAN can improve it by effectively utilizing local data structure information through adaptive-neighborhood.

In summary, SDSPCAAN outperformed other state-of-the-art dimensionality reduction approaches, because it can effectively utilize both global and local data structure information by combining SDSPCA and PCAN.

E. Effect of the Subspace Dimensionality

To study the effect of the subspace dimensionality k, we varied k in $[10, 100]$ while keeping other parameters (α, β and δ) at their best value, and recorded the averaged test BCA of all nine datasets, as shown in Fig. 1. For $k \in [10, 100]$, our proposed SDSPCAAN always outperformed the state-of-the-art JPCDA and SDSPCA. This again indicated that SDSPCAAN can effectively utilize both global and local data structure information, by combining SDSPCA and PCAN.

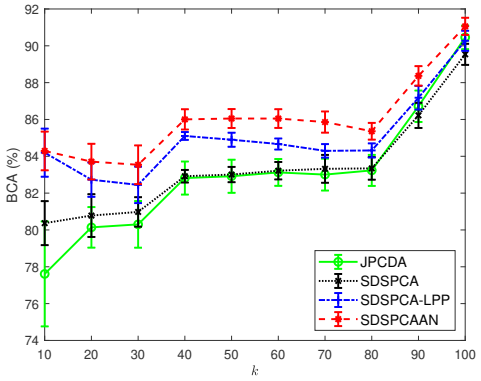


Fig. 1. BCA versus the subspace dimensionality k.

F. SDSPCAAN Parameters Sensitivity

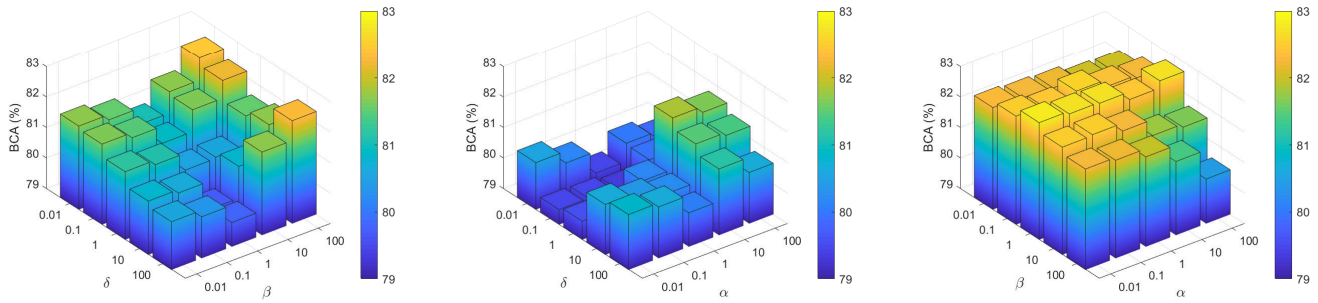
SDSPCAAN has three parameters, α, β and δ. It is important to analyze how these parameters affect its performance. The results are shown in Fig. 2. Take Fig. 2(a) as an example. We changed β and δ, while keeping other parameters (k and α) at their best value, and recorded the averaged test BCA of all nine datasets. We may conclude that SDSPCAAN is robust to α and β in the range $[0.01, 100]$, but sensitive to δ.

IV. CONCLUSION

In this paper, we have proposed a novel linear dimensionality reduction approach, SDSPCAAN, that unifies SDSPCA and PCAN to extract the most discriminant features for classification. Our experiments demonstrated that SDSPCAAN can effectively utilize both global and local data structure information in dimensionality reduction, and learning the similarity graph from adaptive neighbors can further improves its performance. When the extracted features were used in a 1-nearest neighbor classifier, SDSPCAAN outperformed several state-of-the-art linear dimensionality reductions approaches.

ACKNOWLEDGEMENT

This research was supported by the National Natural Science Foundation of China Grant 61873321 and Hubei Technology Innovation Platform Grant 2019AEA171.



(a) (b) (c)

Fig. 2. BCA of SDSPCAAN versus its parameters. (a) β and δ; (b) α and δ; (c) α and β.

REFERENCES

[1] G. Chao, Y. Luo, and W. Ding, "Recent advances in supervised dimension reduction: A survey," *Machine Learning and Knowledge Extraction*, vol. 1, no. 1, pp. 341–358, 2019.

[2] J. P. Cunningham and Z. Ghahramani, "Linear dimensionality reduction: Survey, insights, and generalizations," *Journal of Machine Learning Research*, vol. 16, no. 89, pp. 2859–2900, 2015.

[3] K. Pearson, "On lines and planes of closest fit to systems of points in space," *Philosophical Magazine*, vol. 2, no. 11, pp. 559–572, 1901.

[4] E. Barshan, A. Ghodsi, Z. Azimifar, and M. Z. Jahromi, "Supervised principal component analysis: Visualization, classification and regression on subspaces and submanifolds," *Pattern Recognition*, vol. 44, no. 7, pp. 1357–1371, 2011.

[5] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Proc. 14th Conf. on Neural Information Processing Systems*, Vancouver, Canada, Dec. 2001, pp. 585–591.

[6] B. Jiang, C. H. Q. Ding, B. Luo, and J. Tang, "Graph-Laplacian PCA: Closed-form solution and robustness," in *Proc. 26th IEEE Conf. on Computer Vision and Pattern Recognition*, Portland, OR, Jun. 2013, pp. 3492–3498.

[7] C. Feng, Y. Xu, J. Liu, Y. Gao, and C. Zheng, "Supervised discriminative sparse PCA for com-characteristic gene selection and tumor classification on multiview biological data," *IEEE Trans. on Neural Networks and Learning Systems*, vol. 30, no. 10, pp. 2926–2937, 2019.

[8] F. Nie, H. Huang, X. Cai, and C. H. Q. Ding, "Efficient and robust feature selection via joint $L_{2,1}$-norms minimization," in *Proc. 23th Conf. on Neural Information Processing Systems*, Vancouver, Canada, Dec. 2010, pp. 1813–1821.

[9] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, no. 2, pp. 179–188, 1936.

[10] H. Zhao, Z. Wang, and F. Nie, "A new formulation of linear discriminant analysis for robust dimensionality reduction," *IEEE Trans. on Knowledge and Data Engineering*, vol. 31, no. 4, pp. 629–640, 2019.

[11] M. Guo, F. Nie, and X. Li, "Self-weighted adaptive locality discriminant analysis," in *Proc. 25th IEEE Int'l Conf. on Image Processing*, Athens, Greece, Oct. 2018, pp. 3378–3382.

[12] H. Hotelling, "Relations between two sets of variates," *Biometrika*, vol. 28, no. 3/4, pp. 321–377, 1936.

[13] T. Sun, S. Chen, J. Yang, and P. Shi, "A novel method of combined feature extraction for recognition," in *Proc. 8th IEEE Int'l Conf. on Data Mining*, Pisa, Italy, Dec. 2008, pp. 1043–1048.

[14] C. Guo and D. Wu, "Discriminative sparse generalized canonical correlation analysis (DSGCCA)," in *Proc. Chinese Automation Congress*, Hangzhou, China, Nov. 2019.

[15] X. He, D. Cai, S. Yan, and H. Zhang, "Neighborhood preserving embedding," in *Proc. 10th IEEE Int'l Conf. on Computer Vision*, Beijing, China, Oct. 2005, pp. 1208–1213.

[16] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," in *Proc. 17th Conf. on Neural Information Processing Systems*, Vancouver, Canada, Dec. 2004, pp. 1601–1608.

[17] X. He and P. Niyogi, "Locality preserving projections," in *Proc. 16th Conf. on Neural Information Processing Systems*, Vancouver and Whistler, Canada, Dec. 2003, pp. 153–160.

[18] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[19] M. Sugiyama, "Dimensionality reduction of multimodal labeled data by local Fisher discriminant analysis," *Journal of Machine Learning Research*, vol. 8, no. May, pp. 1027–1061, 2007.

[20] T. Sun and S. Chen, "Locality preserving CCA with applications to data visualization and pose estimation," *Image and Vision Computing*, vol. 25, no. 5, pp. 531–543, 2007.

[21] F. Nie, X. Wang, and H. Huang, "Clustering and projected clustering with adaptive neighbors," in *Proc. 20th ACM Int'l Conf. on Knowledge Discovery and Data Mining*, New York, NY, Aug. 2014, pp. 977–986.

[22] B. Mohar, " The Laplacian spectrum of graphs," in *Graph Theory, Combinatorics, and Applications*, Y. Alavi, G. Chartrand, O. Oellermann, and A. Schwenk, Eds. New York: Wiley, 1991, vol. 2, pp. 871–898.

[23] F. R. K. Chung, *Spectral Graph Theory*, ser. Regional Conference Series in Mathematics. Providence, RI: Amer. Math. Soc., 1997, no. 92.

[24] W. Zhuge, F. Nie, C. Hou, and D. Yi, "Unsupervised single and multiple views feature extraction with structured graph," *IEEE Trans. on Knowledge and Data Engineering*, vol. 29, no. 10, pp. 2347–2359, 2017.

[25] F. Nie, W. Zhu, and X. Li, "Structured graph optimization for unsupervised feature selection," *IEEE Trans. on Knowledge and Data Engineering*, 2019, in press.

[26] W. Wang, Y. Yan, F. Nie, S. Yan, and N. Sebe, "Flexible manifold learning with optimal graph for image and video representation," *IEEE Trans. on Image Processing*, vol. 27, no. 6, pp. 2664–2675, 2018.

[27] Y. Pang, B. Zhou, and F. Nie, "Simultaneously learning neighborship and projection matrix for supervised dimensionality reduction," *IEEE Trans. on Neural Networks and Learning Systems*, vol. 30, no. 9, pp. 2779–2793, 2019.

[28] F. Nie, G. Cai, and X. Li, "Multi-view clustering and semi-supervised classification with adaptive neighbours," in *Proc. 31st AAAI Conf. on Artificial Intelligence*, San Francisco, CA, Feb. 2017, pp. 2408–2414.

[29] T. D. Bie, N. Cristianini, and R. Rosipal, "Eigenproblems in pattern recognition," in *Handbook of Geometric Computing: Applications in Pattern Recognition, Computer Vision, Neuralcomputing, and Robotics*, E. Bayro-Corrochano, Ed. Heidelberg, Germany: Springer-Verlag, Aug. 2005, pp. 129–167.

[30] D. Dua and C. Graff, "UCI machine learning repository." [Online]. Available: http://archive.ics.uci.edu/ml

[31] S. A. Nene, S. K. Nayar, and H. Murase, "Columbia object image library (COIL-20)," Dept. Comput. Sci., Columbia Univ., New York, NY, Tech. Rep. CUCS-006-96, 1996.

[32] X. Zhao, J. Guo, F. Nie, L. Chen, Z. Li, and H. Zhang, "Joint principal component and discriminant analysis for dimensionality reduction," *IEEE Trans. on Neural Networks and Learning Systems*, 2019, in press.

[33] D. Wu, V. J. Lawhern, S. Gordon, B. J. Lance, and C.-T. Lin, "Agreement rate initialized maximum likelihood estimator for ensemble classifier aggregation and its application in brain-computer interface," in *Proc. IEEE Int'l Conf. on Systems, Man, and Cybernetics*, Budapest, Hungary, Oct. 2016, pp. 724–729.