You *et al. Complex Adapt Syst Model (2016) 4:22*
DOI 10.1186/s40294-016-0034-7

Complex Adaptive Systems Modeling

RESEARCH Open Access

Combining graph embedding and sparse regression with structure low-rank representation for semi-supervised learning

Cong-Zhe You[1,2*], Vasile Palade[2] and Xiao-Jun Wu[1]

*Correspondence:
youcongzhe@gmail.com
[1] School of IoT Engineering, Jiangnan University, Wuxi, China
Full list of author information is available at the end of the article

Abstract

In this paper, we propose a novel method for semi-supervised learning by combining graph embedding and sparse regression, termed as graph embedding and sparse regression with structure low rank representation (GESR-LR), in which the embedding learning and the sparse regression are performed in a combined approach. Most of the graph based semi-supervised learning methods take into account the local neighborhood information while ignoring the global structure of the data. The proposed GESR-LR method learns a low-rank weight matrix by projecting the data onto a low-dimensional subspace. The GESR-LR makes full use of the supervised learning information in the construction of the affinity matrix, and the affinity construction is combined with graph embedding in a single step to guarantee the global optimal solution. In the dimensionality reduction procedure, the proposed GESR-LR can preserve the global structure of the data, and the learned low-rank weight matrix can effectively reduce the influence of the noise. An effective novel algorithm to solve the corresponding optimization problem was designed and is presented in this paper. Extensive experimental results demonstrate that the GESR-LR method can obtain a higher classification accuracy than other state-of-the-art methods.

Keywords: Low-rank representation, Sparse representation, Graph embedding, Sparse regression, Semi-supervised classification

Introduction

Complex adaptive systems (CAS) research area is trying to establish a comprehensive and general understanding of the complex world around us (Niazi and Hussain 2013). Complex systems typically involve the generation of high dimensional data and rely on effective analysis and management of such high-dimensional data. High dimensional data exists in a wide variety of real applications, such as text mining, image retrieval, and visual object recognition. While the high performance of computers can address some of the problems of high dimensional data, for example, the time consuming problem, however, the processing of high-dimensional data often suffers from a series of other problems, such as the curse of dimensionality and the impact of noise and redundancy. Fortunately, it has been shown that the high dimensionality of the data is usually small in the intrinsic reduced space.

In the more or less recent past time, researchers have put forward a lot of efficient data dimensionality reduction algorithms (Wang et al. 2014; Zhou and Tao 2013; Nie et al. 2011; Xu et al. 2009; Li et al. 2008). Principal component analysis (PCA) (Belhumeur et al. 1997) is a traditional method that projects the high dimensional data onto a low dimensional space. Linear discriminant analysis (LDA) (Zuo et al. 2006) is a supervised dimensionality reduction method by maximizing the amount of between-class variance relative to the amount of within-class variance (Nie et al. 2009; Yang et al. 2010). Neighborhood component analysis (NCA) (Goldberger et al. 2004) learns a linear transformation by directly maximizing the stochastic variant of the expected leave-one-out classification accuracy on the training set. In order to find the intrinsic manifold structure of data samples, researchers also proposed some nonlinear dimension reduction methods, such as the locally linear embedding (LLE) (Roweis and Saul 2000) and the Laplacian eigenmap (LE) (Belkin and Niyogi 2003). If there are new data samples in the training set, the Laplacian methods need to learn the whole training set again, this is one of the disadvantages of these types of algorithms. In order to solve this problem, He et al. (2005a) put forward the algorithm of locality preserving projection (LPP), in which the linear projection is used to deal with new data samples. Wu et al. (2007) proposed the local learning projection (LLP) method to solve this problem. In addition, the neighborhood preserving embedding (NPE) (He et al. 2005b) algorithm was put forward to keep the local neighborhood structure on the manifold of the data samples. Some previous studies (Zhang et al. 2009; Tenenbaum et al. 2000; Yan et al. 2007) proved that many dimensionality reduction algorithms can be expressed as a unified framework.

However, in real applications, most of the methods mentioned above can only preserve the information of the local neighbors, while ignoring the global structure of the data. The local structure of the dataset may be easily affected by some factors such as noise, illumination or corruption. As a result, the performance of clustering or classification tasks will be reduced because of these. Fortunately, some researches have shown that the recently proposed low-rank representation (LRR) (Liu et al. 2010, 2013) algorithm has a good robustness for datasets that contain noise or corruption. In the past few years, a series of robust classification algorithms based on low-rank representation have been put forward. The Robust PCA (RPCA) (Wright et al. 2009; Candès et al. 2011) use the low-rank representation to recover the structure of subspaces from the dataset corrupted by noise. For subspace segmentation problem, Liu et al. (2010, 2013) use the nuclear norm to find the lowest rank representation of a dataset; in this way, the global structure of the dataset can be well preserved. Unlike the low-rank representation seeking the lowest rank of the dataset, sparse representation finds the sparest representation of a dataset. Zhuang et al. (2012) combine the sparsity and low-rankness together to put forward a non-negative low-rank and sparse representation (NNLRS) for dealing with the high-dimensional dataset. And then they use the representation coefficient matrix to construct the affinity graph for subspace segmentation. Through the combination of sparse representation and low-rank representation, the NNLRS method can both capture the global structure and the local structure of the dataset.

Through the analysis of the above problems, a novel method is proposed in this paper by combining the graph embedding and sparse regression method in a joint optimization framework. And the supervised learning information is also used in the framework

to guide the construction of the affinity graph. In this paper, the construction of the affinity and graph embedding are combined to ensure the overall optimal solution. In the whole learning process, the label information can be accurately propagated through the graph construction. Thus, the linear regression can learn the discriminative projection to better adapt to the labels of the samples and improve the classification rate of the new samples. In order to solve the corresponding optimization problem, this paper proposes an iterative optimization procedure.

In general, the main contributions of this paper are summarized as follows:

1. Different from conventional methods, by both using the low-rank representation and sparse representation which can preserve the global structure and the local structure of the data, the proposed GESR-LR method can learn a novel weight graph.
2. By unifying the graph learning, projection learning and label propagation into a joint optimization framework, the proposed GESR-LR method can guarantee an overall optimum solution.

The remaining of this paper is organized as follows: "Background and related work" section briefly reviews the background and some related works. The proposed GESR-LR method and the corresponding solution are described in "Combined graph embedding and sparse regression with structure low-rank representation" section. Extensive experiments are conducted in "Experiments" section. Finally, we conclude the paper in "Conclusion" section.

Background and related work

Since the proposed method in this paper is based on low-rank representation and manifold embedding (Nie et al. 2014), we briefly review the relevant methods. Given the dataset $X = [x_1, x_2, \ldots, x_u, x_{u+1}, x_n] \in R^{m\times n}$, where the labeled samples are denoted as $x_i|_{i=1}^{u}$ and the unlabeled samples are denoted as $x_j|_{j=u+1}^{n}$. The label information of the labeled samples is denoted as $y_i \in \{1, 2, \ldots, c\}$, where the number of the total classes is c. The label binary indicator matrix Y are defined as follows: given the training sample $x_i (i = 1, \ldots, n)$ and its label vector $y_i \in R^c$, if x_i is the sample from the kth class $(k = 1, \ldots, c)$, then the k-th entry of the label vector y_i is 1 and for the other entries, the value is 0. In this paper, the $l_{r,p}$-norm is defined as follows:

$$\|Q\|_{r,p} = \left(\sum_{i=1}^{u} \left(\sum_{j=1}^{v} |Q_{ij}| \right)^{p/r} \right)^{1/p}.$$

Low-rank representation (LRR)

Given the dataset $X \in R^{m\times n}$ which is drawn from a union of subspaces $\{\Pi_i\}_{i=1}^{c}$, where c is the dimension of the low-dimensional subspaces, and the dataset is corrupted by noise matrix E, the objective function of the LRR method is defined as follows:

$$\begin{aligned} &\min_{Z,E} rank(Z) + \gamma \|E\|_0 \\ &\text{s.t. } X = AZ + E \end{aligned} \tag{1}$$

where A is the dictionary for the low-rank representation, E is the error matrix of the noise or corruption and γ is the parameter to control the influence of the error matrix. Due to the optimization of the rank norm is NP-hard (Nie et al. 2014), in practice, we often use the nuclear norm for relaxation. Thus the objective function of the low-rank representation is defined as follows:

$$\begin{aligned} &\min_{Z,E} \|Z\|_* + \gamma \|E\|_1 \\ &\text{s.t. } X = AZ + E \end{aligned} \tag{2}$$

where $\|\cdot\|_*$ represents the nuclear norm which is a relaxation of the rank norm. $\|\cdot\|_1$ represents the l_1-norm which is a relaxation of the l_0-norm for error matrix. If let $A = I$, we can see that the objective function of LRR is equivalent to RPCA while the goal of RPCA is to recover an approximate matrix from a corrupted subspace. In real applications, we often use the original matrix X as the dictionary. Therefore, the objective of the optimization problem (2) can be rewritten as:

$$\begin{aligned} &\min_{Z,E} \|Z\|_* + \gamma \|E\|_1 \\ &\text{s.t. } X = XZ + E \end{aligned} \tag{3}$$

There are many optimization methods for solving the problem (3). After we get the final result of representation coefficient matrix Z, we can use is as a kind of similarity to construct an affinity graph ($|Z| + |Z^T|$). Then we use the spectral clustering method on the affinity graph to obtain the final clustering result.

Flexible Manifold Embedding (FME)

Given the dataset X, we assume the predicted label matrix is F, then we can have $F = X^T W + 1b^T$ if the label is strict to lie in the space of the give matrix X, in which $1 \in R^{n\times 1}$ is an all 1 vector. $W \in R^{m\times c}$ is the projection matrix. However, as the objective function $F = X^T W + 1b^T$ is a linear format, if the samples is from a nonlinear manifold, this may be too strict to fit the samples. Therefore, it is reasonable to add a residual item in the regression model of FME (Nie et al. 2010). Then the objective function of FME is relaxed to $F = X^T W + 1b^T + F_0$, where F_0 is the residual item between the predicted label matrix F and $X^T W + 1b^T$. The advantage of this kind of relaxation can make the processing of the sample data points on the nonlinear manifolds more flexible. The goal of FME is to predict the sample label matrix F and reduce the residual of regression F_0 at the same time. The objective function of FME is defined as follows:

$$\begin{aligned} \left(F_*, W_*, F_0^*\right) = \arg\min_{F,W,F_0} & tr(F-Y)^T U(F-Y) + tr\left(F^T L F\right) \\ & + \mu\left(\|W\|^2 + \gamma\|F_0\|^2\right) \end{aligned} \tag{4}$$

where the two parameters μ and γ are used to balance the influence of the two terms. $L \in R^{n\times n}$ is the Laplacian matrix and $U \in R^{n\times n}$ is the diagonal matrix. $tr(\cdot)$ represents

the trace of a matrix. The first two terms in (4) is used to propagate the labels from the labeled samples to unlabeled samples. The last two terms are the regression model. If we use the $X^T W + 1b^T - F$ to replace the regression residual F_0, then the objective function of FME can be expressed as follows:

$$\begin{aligned}\left(F_*, W_*, F_0^*\right) = \arg\min_{F,W,F_0} & tr(F-Y)^T U(F-Y) + tr\left(F^T LF\right) \\ & + \mu\left(\|W\|^2 + \gamma\left\|X^T W + 1b^T - F\right\|^2\right)\end{aligned} \tag{5}$$

Combined graph embedding and sparse regression with structure low-rank representation

In this section, we introduce the details of the proposed method in this paper. The objective of GESR-LR is to unify the graph embedding and regression into a unified framework. The objective of regression model is to find a projection matrix $W \in R^{m\times c}$ to match the sample labels $F \in R^{n\times c}$, and use it to classify the new samples. Thus, the objective function of the regression model can be defined as follows:

$$F = X^T W + F_0 \tag{6}$$

where F_0 is the regression residual (Nie et al. 2010).

In the following section, we first introduce the motivation of the proposed method of GESR-LR, summarize the objective function of the GESR-LR method and propose the optimization solution.

Motivations

For the label propagation problem, we usually have the following hypothesis: a data sample and its nearest neighbors usually belong to the same class, and the nearest neighbors would have a big influence in the determination of the labels of new data samples. In short, the labels of similar samples should be close and we can propagate the labels to similar samples. Therefore, in the construction of an ideal graph we should consider that similar data points and their nearest neighbors should be assigned larger weight values. However, the evaluation of similarity of most traditional graph construction methods mainly depends on the pair-wise Euclidean distance, while the Euclidean distance is very sensitive to noise and any other corruption of the data samples (Zhuang et al. 2012). However, these methods can only capture the local structure of the dataset, but ignore to preserve the global structure of the dataset. Fortunately, some recent studies show that the LRR method can preserve the global structure of the dataset, and it is robust to noise and the corruption of the dataset (Liu et al. 2010, 2013). As a result, these low-rank properties can be combined with the graph embedding problem, and thus it can address the sensitivity with respect to the local and neighbor properties. So, the main idea of constructing an informative graph is to use the low-rankness property to preserve the local and the global structure of the dataset with noise. Following the above analysis, we put forward a novel method of joint graph embedding and sparse regression with structure low-rank representation, named GESR-LR, presented in the next sections in this paper.

GESR-LR model

The aim of the proposed GESR-LR method is to design an optimization framework to combine graph embedding and sparse regression in order to get a global overall optimum solution. Based on the above analysis of low-rank representation (LRR) and flexible manifold embedding (FME), the objective function of the proposed GESR-LR is defined as follows:

$$\begin{aligned} &\min \sum_{i=1}^{n} U_{ii}(F_{il} - Y_{il})^2 + \sum_{i=1}^{n}\sum_{j=1}^{n} \left\|F_i - F_j\right\|_2^2 Z_{ij} \\ &\quad + \alpha \left\|X^T W - F\right\|_2^2 + \beta \|W\|_{21} \\ &\quad + \lambda_1 \|Z\|_* + \lambda_2 tr(\Theta(Z \odot M)) + \gamma \|E\|_{21} \\ &\text{s.t. } X = AZ + E, \quad Z \geq 0 \end{aligned} \tag{7}$$

where $M_{ij} = \left\|X_i - X_j\right\|_2^2$, U is a diagonal matrix defined as

$$U = \begin{cases} \varsigma & \textit{if } x_i \textit{ is tagged} \\ 0 & \textit{otherwise} \end{cases} \tag{8}$$

ς is a large constant such that F_{*l} and Y_{*l} $(l = 1, 2, \ldots, c)$ can be approximately satisfied, and $F \in R^{n \times c}$ is the predicted labels of both labeled and unlabeled samples. In the objective function (7), the aim of the first term is to assess the fitness of labels which means that the predicted labels F should be close to the labels of the labeled data samples. The second term is the graph embedding and it aims at integrating the regression, graph embedding and label propagation for the unlabeled data samples from the labeled data samples. For the data point x_i, if we get a larger weight Z_{ij}, this means that the label F_{*j} has a bigger influence on the prediction of the label F_{*i} for the data point x_i. The third item is used to minimize the regression residual. The third and fourth items represent the regression model, the goal being to learn the projection for fitting the labels of the data samples and classifying new data points. In this method, we adopt the $l_{2,1}$-norm to regularize the projection matrix W, so that it is guaranteed that W is sparse in row for feature selection. The last three items adopt the low-rank representation to learn a weight graph. The five parameters α, β, λ_1, λ_2 and γ are used to balance the influence of the corresponding five terms. Therefore, the objective function of the proposed GESR-LR method can be formulated as follows:

$$\begin{aligned} &\min tr\left((F - Y)^T U (F - Y)\right) + tr(F^T L F) \\ &\quad + \alpha \left\|X^T W - F\right\|_2^2 + \beta \|W\|_{21} \\ &\quad + \lambda_1 \|Z\|_* + \lambda_2 tr(\Theta(Z \odot M)) + \gamma \|E\|_{21} \end{aligned} \tag{9}$$

where $L = D - S$ is the Laplacian matrix, and D is a diagonal matrix with $D_{ii} = \frac{\sum Z_{i*} + \sum Z_{*i}}{2}$.

The solution of GESR-LR

The optimization problem of (9) can be solved by calculating W independently and updating F and Z iteratively. In order to solve the optimization problem of (9), we introduce an auxiliary variable S to separate the objective function. We firstly convert the problem of (9) to the following equivalent optimization problem:

$$\begin{aligned} &\min tr\Big((F-Y)^T U(F-Y)\Big) + tr\Big(F^T LF\Big) \\ &\quad + \alpha\Big\|X^T W - F\Big\|_2^2 + \beta\|W\|_{21} \\ &\quad + \lambda_1\|Z\|_* + \lambda_2 tr(\Theta(Z\odot M)) + \gamma\|E\|_{21} \\ &\text{s.t. } X = AZ + E, \quad Z = S, \quad S \geq 0 \end{aligned} \tag{10}$$

In order to solve the optimization problem, we first transfer the optimization problem to the Lagrange function, and the Lagrange function of problem (10) is as follows:

$$\begin{aligned} L &= \min \sum_{i=1}^{n} U_{ii} \sum_{l=1}^{c} (F_{il} - Y_{il})^2 + \sum_{i=1}^{n}\sum_{j=1}^{n} \|F_i - F_j\|_2^2 S_{ij} + \alpha\Big\|X^T W - F\Big\|_2^2 + \beta\|W\|_{21} \\ &\quad + \lambda_1\|Z\|_* + \lambda_2 tr(\Theta(S\odot M)) + \gamma\|E\|_{21} \\ &\quad + \langle Y_1, X - AZ - E\rangle + \langle Y_2, Z - S\rangle \\ &\quad + \frac{\mu}{2}\Big(\|X - AZ - E\|_F^2 + \|Z - S\|_F^2\Big) \\ &= \min \sum_{i=1}^{n} U_{ii} \sum_{l1}^{c} (F_{il} - Y_{il})^2 + \sum_{i=1}^{n}\sum_{j=1}^{n} \|F_i - F_j\|_2^2 S_{ij} + \alpha\Big\|X^T W - F\Big\|_2^2 + \beta\|W\|_{21} \\ &\quad + \lambda_1\|Z\|_* + \lambda_2 tr(\Theta(S\odot M)) + \gamma\|E\|_{21} \\ &\quad + \langle Y_1, X - AZ - E\rangle + \langle Y_2, Z - S\rangle \\ &\quad + \varphi(Z, S, E, Y_1, Y_2, \mu) - \frac{1}{2\mu}\Big(\|Y_1\|_F^2 + \|Y_2\|_F^2\Big) \\ &\text{s.t. } S \geq 0 \end{aligned} \tag{11}$$

where $\varphi(Z,S,E,Y_1,Y_2,\mu) = \frac{\mu}{2}\left(\left\|X - AZ - E + \frac{Y_1}{\mu}\right\|_F^2 + \left\|Z - S + \frac{Y_2}{\mu}\right\|_F^2\right)$ and $\langle A, B\rangle = tr(A^T B)$. Y_1 and Y_2 are the Lagrange multipliers and $\mu \geq 0$ is a penalty parameter. For solving the optimization problem, we use the LADMAP method. By fixing the other variables, the LADMAP updates the variables W, F, Z, S and E alternately, and then it updates Y_1 and Y_2.

1. By fixing F, Z and S, W is solved by the following optimization problem:

$$L(W) = \arg\min_W \Big\|X^T W - F\Big\|_2^2 + \beta\|W\|_{21} \tag{12}$$

By setting the derivative $\frac{\partial L(W)}{\partial W} = 0$, we have the following equation:

$$\frac{\partial L(W)}{\partial W} = 2XX^T W - 2XF + 2\beta DW = 0 \tag{13}$$

where $D_{ii} = \frac{1}{2\|w_i\|_2}$

Or equivalently

$$W = \left(XX^T + \beta D\right)^{-1} XF = AF \tag{14}$$

where $A = (XX^T + \beta D)^{-1}X$.

2. By fixing W, S and Z, F is solved by the following optimization problem:

$$L(F) = \arg\min_F tr\left((F-Y)^T U(F-Y)\right) + tr\left(F^T LF\right) + \alpha\left\|X^T W - F\right\|_F^2 \tag{15}$$

This is an unconstrained optimization problem. Let $W = AF$ and integrate in the objective function and find the derivation of the problem (11) with respect to F, by making the value of the derivative to zero, and we have

$$\frac{\partial L(F)}{\partial F} = UF - UY + LF + \alpha BF = 0 \tag{16}$$

$$F = (U + L + \alpha B)^{-1} UY \tag{17}$$

where $B = ((X^T A - I)^T (X^T A - I))$.

3. By fixing W, S and F, Z is solved by the following optimization problem:

$$\begin{aligned} L\left(Z^{k+1}\right) &= \arg\min_Z \lambda_1 \|Z\|_* + \langle \nabla_Z \varphi(Z, S, E, Y_1, Y_2, \mu), Z - Z_k\rangle + \frac{\mu\theta}{2}\|Z - Z_k\|_F^2 \\ &= \arg\min_Z \|Z\|_* \\ &\quad + \frac{\mu\theta}{2}\left\| Z - Z^k + \frac{\left[-X^T\left(X - AZ^K - E + \frac{Y_1}{\mu}\right) + \left(Z^k - S + \frac{Y_2}{\mu}\right)\right]}{\theta}\right\|_F^2 \end{aligned} \tag{18}$$

where $\nabla_Z \varphi$ is the partial differential of φ with respect to Z, $\theta = \|A\|_F^2$, $\|\cdot\|_F$ represents the Frobenius norm.

$$Z^{k+1} = J_{\frac{\lambda_1}{\theta\mu}}\left(Z^k - \frac{\left[-X^T\left(X - AZ^K - E + \frac{Y_1}{\mu}\right) + \left(Z^k - S + \frac{Y_2}{\mu}\right)\right]}{\theta}\right) \tag{19}$$

where J is the thresholding operator with respect to the singular value $\frac{\lambda_1}{\theta\mu}$. A proximal optimization method can be used to find the solution of Z.

4. By fixing W, Z and F, S is solved by the following optimization:

$$L\left(S^{k+1}\right) = \arg\min_S tr\left(F^T LF\right) + \lambda_2 tr\left(\Theta\left(S^k \odot M\right)\right) + \frac{\mu}{2}\left\|S^k - \left(Z + \frac{Y_2}{\mu}\right)\right\|_F^2 \tag{20}$$

Let $R = \lambda_2 M + V$, $V_{ij} = \left\|F_i - F_j\right\|_2^2$. The optimization problem in (20) can be decomposed into n independent sub-problems, and each of these sub-problems can be formulated as a weight non-negative sparse coding problem as follows:

$$\min_{S_i} \sum_{g=1}^{n} \left(S^k\right)_g^i \odot R_g^i + \frac{\mu}{2}\left\|\left(S^k\right)^i - \left(Z^{k+1} + \frac{Y_2}{\mu}\right)\right\|_2^2 \tag{21}$$

s.t. $S \geq 0$

where $(S^k)_g^i$ and $(R)_g^i$ are the g-th elements of i-th columns of matrices S^k and R. Therefore, the problem of (21) has a closed form solution (Yang et al. 2013; Zhang et al. 2012).

5. By fixing W, Z, S and F, E is solved by the following optimization problem as follows:

$$L(E) = \arg\min_E \gamma\|E\|_{21} + \frac{\mu}{2}\left\|X - AZ + \frac{Y_1}{\mu} - E\right\|_F^2 \tag{22}$$

from the above analysis, we can find that, on one hand, the deduction of the variables F, Z, S and E are closely dependent. On the other hand, the solution of variable W is only related to the variable F. Therefore, we can update the variables F, Z, S and E, iteratively, by fixing the other variables fixed. We can calculate the variable W by $W = AF$ after getting the optimal solution of F.

The overall optimization framework for the proposed GESR-LR method is described in Algorithm 1.

Algorithm 1: GESR-LR proposed method

Input: Data set matrix X; Matrix U; Label indicator matrix Y; parameters α, β, λ_1, λ_2 and γ;

Initialization: $Z = S = 0$; $E = 0$; $F = 0$; $Y_1 = Y_2 = 0$; $\mu_0 = 0.1$, $\mu_{\max} = 10^7$, $\rho_0 = 1.01$, $\varepsilon = 10^{-7}$

While not converged Do

1. Fix the other variables and update F by solving (17)
2. Fix the other variables and update Z by solving (18)
3. Fix the other variables and update S by solving (20)
4. Fix the other variables and update E by solving (22)
5. Update the Lagrange multipliers as follows

$$\begin{cases} Y_1^{k+1} = Y_1^k + \mu^k \left(X - AZ^k - E^k \right) \\ Y_2^{k+1} = Y_2^k + \mu^k \left(Z^k - S^k \right) \end{cases}$$

6. Update the parameter μ as follows

$\mu^{k+1} = \min\left(\mu_{\max}, \rho\mu^k\right)$, where

$$\rho = \begin{cases} \rho_0 & if\ \mu^k \Omega / \|X\|_F \le \varepsilon \\ 1 & ifotherwise \end{cases}$$

7. Check the convergence conditions

$\|X - AZ^k - E^k\|_F / \|X\|_F \le \varepsilon$ or $\mu^k \Omega / \|X\|_F \le \varepsilon$

Where $\Omega = \max\left(\sqrt{\theta}\|Z^k - Z^{k+1}\|_F, \|S^k - S^{k+1}\|_F, \|E^k - E^{k+1}\|_F, \|F^k - F^{k+1}\|_F\right)$

8. Update $k = k + 1$

End while

Calculate $W = AF$

Output: The predicted label matrix F; The projection matrix W

Experiments

Human social systems and human facial structure recognition is the emergent outcome of adaptation over a period of time (Holland and John 2012). Here, in the experiments described in this paper, we have used several datasets to evaluate the performance of the proposed GESR-LR method (http://www.cad.zju.edu.cn/home/dengcai/Data/data.html), including two human face images datasets (i,e., the ORL and the extended Yale B datasets) in addition to an object dataset (COIL-20), a spoken letter recognition dataset (Isolet 5) and a handwritten digit dataset (USPS dataset). The datasets contain the common images information in daily life, and they are widely used in the areas of image processing, machine learning, etc. The computing platform is matlab R2015B in a PC with CPU i7 2600, RAM 16G,

Datasets descriptions

1. *ORL dataset* The ORL dataset consists of 400 face images of 40 people. These face images are taken under different situations, such as different time, varying lighting, facial details (glasses/no glasses) and facial expressions (open/closed eyes, smiling/not smiling).
2. *The extended Yale B dataset* The extended Yale B dataset contains the face images of 38 people, each individual has around 64 frontal face images which are taken under different illuminations. For computing efficiency, we adjust the size of each image to 32×32 pixels in this experiments.
3. *COIL-20 dataset* The COIL-20 dataset contains the images of 20 objects, each object has 72 images and the images are collected from varying every five degrees. For computation efficiency purposes, we adjust the size of each image to 32×32 pixels in this experiments.
4. *ISOLET 5 dataset* The ISOLET spoken letter recognition dataset consists of 150 subjects, where each person speaks each letter from the alphabet twice. The speakers are divided into 5 groups, each group has 30 speakers, and this is marked as ISOLET 5 dataset. In this work, the ISOLET 5 dataset contains 1559 images, with images from 26 people, each speaker providing 60 images.
5. *USPS dataset* The USPS dataset is a handwritten digit dataset, which contains two parts: the training set with 7291 samples, and the test set with 2007 samples. In this experiment, we randomly selected 7000 images of the 10 letters. Thus, there are 700 images in each category. The size of each images is 16×16 pixels.

Classification results

In this section, we evaluate the performance of the proposed GESR-LR method. For the semi-supervised problem, we compare the proposed GESR-LR method with the following algorithms: FME (Nie et al. 2010), GFHF (Zhu et al. 2003), NNSG, SDA (Cai et al. 2007), LapRLS/L (Belkin et al. 2006), Transductive component analysis (TCA) (Liu et al. 2008), and MFA (Yan et al. 2007). We also use the learned projection matrix to classify the new samples. The classification method used in our experiments is the nearest neighbor (NN) classification. For the NNSG and GFHF methods, the classification method is as indicated in the corresponding research paper in (Zhou et al. 2004). For some embedding algorithms, we first learn the graph Laplacian matrix L while the graph weight matrix is defined as $S_{ij} = e^{\frac{-\left\|x_i - x_j\right\|^2}{\sigma}}$. The number of the nearest neighbors are chosen from the set of $\{3, 4, 5, 6, 7, 8, 9, 10\}$, and the kernel parameters are from the set of $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}, 10^{3}, 10^{4}\}$. The final dimensions of some algorithms, such as FME, LapRLS/L, SDA, TCA and MFA are set to the number of the classes and the parameters in these methods are set to the best value according to the related research papers. While the parameters of the proposed method GESR-LR (α, β, λ_1, λ_2 and γ) are chosen from the range of $(10^{-4}, 10^{0})$. For the sake of computational efficiency, all data in these data sets were eventually reduced to 60D vectors.

We performed the experiments on the above datasets: ORL, the extended Yale B, COIL-20, Isolet5 and USPS. For every dataset, we randomly selected 50 % samples of each subject as the training sample set, while the remaining samples are selected as the testing set. For the semi-supervised classification, we select p samples per subject as the

labeled data samples, while the remaining formed the unlabeled data samples. The unlabeled data samples are used to test the performance of semi-supervised classification, while the testing sample set is used to test the performance of classifying the new data samples with the learned projection matrix.

For the dataset of ORL, COIL-20, Isolet5 and USPS, the number of the labeled data sample is set to p = 1, 2 and 3, respectively. For the dataset of the extended Yale B, the number of the labeled data samples is set to p = 5, 10 and 15, respectively. In addition to the MFA algorithm, where the labeled samples were used for subspace learning, for the other algorithms, the training samples are used to learn the projection matrix. We run the experiments 30 times on the unlabeled data samples and the test data samples, and we obtain the mean classification accuracy and standard deviation (%). In the Tables 1, 2, 3, 4, 5, the corresponding experiments are referred as Semi and Test respectively. From these experimental results, we can get the following conclusions:

1. In terms of classification accuracy, the semi-supervised classification algorithms TCA, LapRLS/L, SDA get a higher classification accuracy than the supervised classification algorithm MFA. This shows that the unlabeled data samples help to improve the performance of the semi-supervised classification.
2. In some datasets, the GFHF algorithm achieves higher semi-supervised classification accuracy than that of TCA, LapRLS/L and SDA algorithms, especially on the datasets which have some strong variations. For example, the extended Yale B dataset has

Table 1 Semi-supervised classification results of different algorithms on the COIL-20 dataset

Method	P = 1		P = 2		P = 3	
	Semi (%)	Test (%)	Semi (%)	Test (%)	Semi (%)	Test (%)
GFHF	78.65 ± 2.07	–	81.32 ± 1.77	–	84.56 ± 2.02	–
MFA	–	–	69.87 ± 2.24	70.10 ± 2.52	76.54 ± 2.28	76.27 ± 2.37
SDA	64.92 ± 2.07	65.80 ± 2.54	72.24 ± 2.19	73.19 ± 2.15	78.89 ± 2.05	78.19 ± 2.66
TCA	71.08 ± 2.23	70.83 ± 2.51	78.17 ± 3.15	77.29 ± 2.18	81.15 ± 2.32	80.96 ± 2.27
LapRLS	69.46 ± 2.58	69.73 ± 2.76	75.21 ± 2.66	75.16 ± 2.31	79.61 ± 2.54	79.85 ± 2.59
FME	76.31 ± 2.09	74.46 ± 2.13	82.35 ± 2.18	79.14 ± 2.39	85.86 ± 1.92	84.70 ± 2.03
NNSG	79.15 ± 2.86	75.31 ± 2.01	83.79 ± 2.69	80.88 ± 2.43	86.62 ± 2.29	82.13 ± 2.24
GESR-LR	81.09 ± 2.33	76.79 ± 2.18	85.29 ± 2.62	81.07 ± 2.59	87.12 ± 2.15	83.32 ± 2.16

Table 2 Semi-supervised classification results of different algorithms on the USPS dataset

Method	P = 1		P = 2		P = 3	
	Semi (%)	Test (%)	Semi (%)	Test (%)	Semi (%)	Test (%)
GFHF	72.39 ± 3.60	–	79.66 ± 3.67	–	83.39 ± 3.07	–
MFA	–	–	68.74 ± 3.82	66.52 ± 4.23	72.76 ± 4.21	70.57 ± 3.19
SDA	56.86 ± 3.11	54.91 ± 3.92	67.37 ± 3.26	67.43 ± 2.91	72.66 ± 2.64	69.32 ± 3.20
TCA	70.39 ± 3.38	65.36 ± 3.17	76.52 ± 3.21	71.27 ± 3.28	79.58 ± 3.37	72.76 ± 2.94
LapRLS	57.89 ± 4.08	58.42 ± 4.36	69.03 ± 3.86	69.39 ± 2.49	76.02 ± 3.28	74.08 ± 2.79
FME	74.75 ± 6.52	67.91 ± 5.04	79.64 ± 3.41	73.26 ± 3.19	82.15 ± 2.26	74.97 ± 2.72
NNSG	76.98 ± 3.80	68.92 ± 3.37	81.17 ± 2.59	76.85 ± 2.57	84.50 ± 2.13	76.38 ± 2.54
GESR-LR	78.49 ± 3.65	69.56 ± 3.18	83.61 ± 2.36	77.28 ± 2.29	86.07 ± 2.73	78.20 ± 2.17

Table 3 Semi-supervised classification results of different algorithms on the ISOLET5 dataset

Method	P = 1		P = 2		P = 3	
	Semi (%)	Test (%)	Semi (%)	Test (%)	Semi (%)	Test (%)
GFHF	49.29 ± 2.15	–	56.26 ± 2.44	–	61.13 ± 2.14	–
MFA	–	–	61.19 ± 2.14	61.46 ± 2.89	65.52 ± 2.27	65.19 ± 2.36
SDA	52.01 ± 2.38	51.19 ± 2.54	61.31 ± 2.28	61.57 ± 2.35	67.55 ± 2.28	67.91 ± 2.06
TCA	49.19 ± 2.94	49.30 ± 2.13	59.77 ± 2.36	59.16 ± 2.42	64.72 ± 2.37	65.01 ± 2.38
LapRLS	51.71 ± 3.03	50.98 ± 2.84	61.63 ± 2.37	61.85 ± 2.21	65.19 ± 1.89	65.25 ± 2.05
FME	49.92 ± 2.40	50.17 ± 2.49	59.92 ± 2.45	59.88 ± 2.56	65.98 ± 1.64	66.13 ± 2.29
NNSG	53.39 ± 2.26	51.75 ± 2.37	62.84 ± 2.57	62.63 ± 2.26	67.33 ± 2.21	67.94 ± 2.15
GESR-LR	55.01 ± 2.25	52.26 ± 2.82	63.09 ± 2.12	63.13 ± 2.43	69.26 ± 2.24	70.03 ± 1.78

Table 4 Semi-supervised classification results of different algorithms on the ORL dataset

Method	P = 1		P = 2		P = 3	
	Semi (%)	Test (%)	Semi (%)	Test (%)	Semi (%)	Test (%)
GFHF	52.81 ± 4.31	–	63.26 ± 3.78	–	68.97 ± 3.54	–
MFA	–	–	78.22 ± 4.25	79.11 ± 3.76	85.40 ± 3.89	84.78 ± 2.54
SDA	65.29 ± 2.72	65.32 ± 2.83	75.84 ± 3.61	76.92 ± 3.25	82.44 ± 2.54	82.95 ± 2.26
TCA	64.75 ± 2.05	64.61 ± 2.29	77.02 ± 3.15	78.80 ± 2.57	84.49 ± 3.12	84.27 ± 2.67
LapRLS	61.49 ± 3.31	59.88 ± 3.10	78.29 ± 2.54	77.86 ± 2.71	85.83 ± 2.75	85.94 ± 2.39
FME	68.25 ± 2.58	66.69 ± 3.24	80.80 ± 3.25	80.73 ± 2.76	85.92 ± 3.67	84.35 ± 2.64
NNSG	71.86 ± 3.29	67.77 ± 3.73	82.57 ± 2.65	82.91 ± 2.15	86.38 ± 3.83	85.52 ± 2.97
GESR-LR	73.08 ± 3.17	69.29 ± 3.68	85.52 ± 2.14	85.64 ± 2.89	87.45 ± 3.54	86.12 ± 2.99

Table 5 Semi-supervised classification results of different algorithms on the extended Yale B dataset

Method	P = 5		P = 10		P = 15	
	Semi (%)	Test (%)	Semi (%)	Test (%)	Semi (%)	Test (%)
GFHF	27.49 ± 1.27	–	34.76 ± 2.11	–	40.13 ± 2.02	–
MFA	–	–	69.52 ± 3.19	70.08 ± 3.26	73.90 ± 2.72	74.15 ± 3.42
SDA	51.92 ± 2.36	52.06 ± 1.58	66.76 ± 1.65	67.49 ± 1.41	73.40 ± 1.19	73.08 ± 1.78
TCA	51.47 ± 2.19	52.56 ± 2.34	65.94 ± 1.95	66.76 ± 2.25	74.38 ± 1.76	74.28 ± 2.37
LapRLS	60.16 ± 2.24	59.47 ± 1.83	74.85 ± 1.67	74.19 ± 1.47	78.64 ± 2.54	78.08 ± 2.67
FME	63.46 ± 2.14	63.75 ± 1.89	76.92 ± 2.38	74.37 ± 1.22	80.38 ± 1.77	78.19 ± 2.03
NNSG	72.37 ± 2.25	66.92 ± 1.64	82.25 ± 1.64	75.42 ± 1.27	83.38 ± 1.93	79.06 ± 1.25
GESR-LR	75.26 ± 2.59	68.13 ± 1.54	84.11 ± 1.57	76.61 ± 1.95	85.87 ± 1.69	80.52 ± 1.28

strong illumination changes and expression. In this case, the label propagation may not perform well. This phenomenon is more obvious on the extended Yale B dataset.

3. On the unlabeled dataset, the performance of the proposed GESR-LR algorithm is obviously better than the compared methods. This indicates that the structure of the graph obtained by the GESR-LR method has more discriminant information, which is more effective for the label propagation. This also suggests that simultaneously performing label propagation and graph learning is necessary and effective.

The GESR-LR method requires five parameters (α, β, λ_1, λ_2 and γ) to be set in advance. Figure 1 shows the classification accuracy versus the variations of the five parameters, respectively, on the extended Yale B dataset. 50 % of samples per subject were randomly selected as training samples and remaining samples were used as test samples. We report the mean recognition accuracy over 20 random splits. Obviously, it can be found that when the parameters vary in a relatively large ranges, the performance of the proposed GESR-LR method is more stable.

Next, we consider the effectiveness of the algorithm when different dimension sizes are used. The experiment is conduct on the extended Yale B dataset. We also report the mean recognition accuracy over 20 random splits. We can see from the Fig. 2, when use

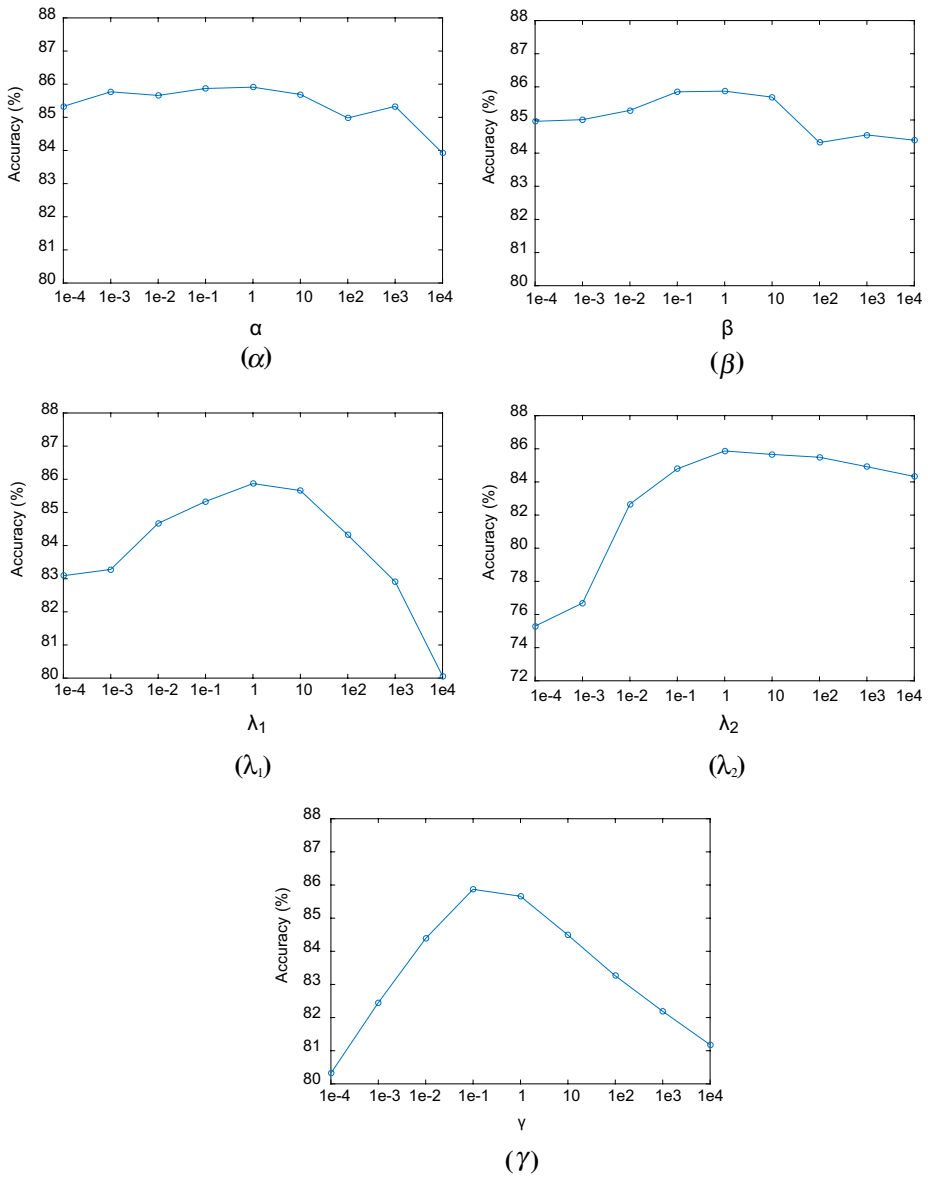

Fig. 1 Classification accuracy versus parameters on the extended Yale B dataset

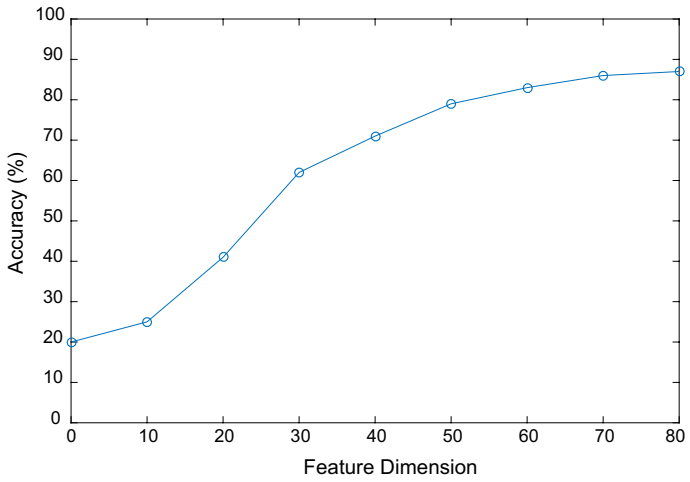

Fig. 2 Classification accuracy versus different feature dimension on the extended Yale B dataset

larger dimensions of the feature, the accuracy increase, while when the number of features is over 60, the accuracy increases slowly and it is much stable.

Conclusion

Complex adaptive systems (CAS) involve the processing of large amounts of high dimensional data. It is thus paramount to develop and employ effective machine learning techniques to deal with such high dimensional and large datasets generated from the CAS area. In this paper, we proposed a novel semi-supervised learning method termed as graph embedding and sparse regression with structure low rank representation (GESR-LR), by combing graph embedding and sparse regression, which are performed simultaneously in order to get an optimal solution. Different from some traditional methods, the proposed GESR-LR method takes into account both the local and global structure of the dataset to construct an informative graph. Extensive experiments on five datasets demonstrate that the GESR-LR method outperform the state-of-the-art methods. In our future work, we will extend the ideas presented in this paper and will apply the proposed GESR-LR method to other challenging problems.

Abbreviations

GESR-LR: graph embedding and sparse regression with structure low rank representation; CAS: complex adaptive systems; PCA: Principal component analysis; LDA: linear discriminant analysis; NCA: neighborhood component analysis; LLE: locally linear embedding; LE: Laplacian eigenmap; LPP: locality preserving projection; LLP: local learning projection; NPE: neighborhood preserving embedding; LRR: low-rank representation; RPCA: robust principal component analysis; NNLRS: non-negative low-rank and sparse representation; FME: flexible manifold embedding; LADMAP: linearized alternating direction method with adaptive penalty; TCA: transductive component analysis; NN: the nearest neighbor.

Authors' contributions

CZY conceived the study, performed the experiments and wrote the paper. VP and XJW reviewed and edited the manuscript. All authors read and approved the final manuscript.

Authors' information

Cong-Zhe You is a Ph.D. Candidate in Jiangnan University, now he is a visiting student in Coventry University sponsored by the China Scholarship Council. His research interests are in the area of Machine Learning and Pattern Recognition.

Vasile Palade received the Ph.D. degree from the University of Galati, Galaţi, Romania, in 1999. He is currently a Reader with the School of Computing, Electronics and Maths, Coventry University, Coventry, U.K. His research interests include computational intelligence with application to bioinformatics, fault diagnosis, web usage mining, among others. He published more than 100 papers in Journals and conference proceedings as well as several books.

Xiao-Jun Wu received the B.Sc. degree in mathematics from Nanjing Normal University, Nanjing, China, in 1991. He received the M.S. degree in 1996, and the Ph.D. degree in pattern recognition and intelligent systems in 2002, both from Nanjing University of Science and Technology, Nanjing, China. He joined Jiangnan University in 2006, where he is currently a Professor. He has published more than 150 papers in his fields of research. He was a visiting researcher in the Centre for Vision, Speech, and Signal Processing (CVSSP), University of Surrey, U.K., from 2003 to 2004. His current research interests include pattern recognition, computer vision, fuzzy systems, neural networks, and intelligent systems.

Author details
[1] School of IoT Engineering, Jiangnan University, Wuxi, China. [2] School of Computing, Electronics and Maths, Coventry University, Coventry, UK.

Acknowledgements
The authors would like to thank the anonymous reviewers and editors for their valuable suggestions.

Competing interests
The authors declare that they have no competing interests.

Funding
This study was funded by the National Natural Science Foundation of China (Grant No. 61373055) and the Specialized Research Fund for the Doctoral Program of Higher Education of China (Grant No. 20130093110009).

Received: 8 August 2016 Accepted: 1 October 2016
Published online: 07 October 2016

References

Belhumeur PN, Hespanha JP, Kriegman DJ (1997) Eigenfaces vs. fisherfaces: recognition using class specific linear projection. IEEE Trans Pattern Anal Mach Intell 19(7):711–720
Belkin M, Niyogi P (2003) Laplacian eigenmaps for dimensionality reduction and data representation. Neural Comput 15(6):1373–1396
Belkin M, Niyogi P, Sindhwani V (2006) Manifold regularization: a geometric framework for learning from labeled and unlabeled examples. J Mach Learn Res 1(7):2399–2434
Cai D, He X, Han J (2007) Semi-supervised discriminant analysis. In: IEEE 11th international conference on computer vision, 2007, ICCV 2007. IEEE, New York, pp 1–7
Candès EJ, Li X, Ma Y, Wright J (2011) Robust principal component analysis? J ACM 58(3):11
Goldberger J, Hinton GE, Roweis ST, Salakhutdinov R (2004) Neighborhood components analysis. In: Advances in neural information processing systems, pp 513–520
He X, Yan S, Hu Y, Niyogi P, Zhang HJ (2005a) Face recognition using Laplacianfaces. IEEE Trans Pattern Anal Mach Intell 27(3):328–340
He X, Cai D, Yan S, Zhang HJ (2005b) Neighborhood preserving embedding. In: Tenth IEEE international conference on computer vision, 2005. ICCV 2005, vol 2. IEEE, New York, pp 1208–1213
Holland, John H (2012) Signals and boundaries: building blocks for complex adaptive systems. Mit Press, Cambridge
Li X, Lin S, Yan S, Xu D (2008) Discriminant locally linear embedding with high-order tensor data. IEEE Trans Syst Man Cybern Part B 38(2):342–352
Liu W, Tao D, Liu J (2008) Transductive component analysis. In: Eighth IEEE international conference on data mining, 2008, ICDM'08. IEEE, New York, pp 433–442
Liu G, Lin Z, Yu Y (2010) Robust subspace segmentation by low-rank representation. In: ICML, pp 663–670
Liu G, Lin Z, Yan S, Sun J, Yu Y, Ma Y (2013) Robust recovery of subspace structures by low-rank representation. IEEE Trans Pattern Anal Mach Intell 35(1):171–184
Niazi Muaz A, Hussain Amir (2013) Complex adaptive systems. Cognitive agent-based computing-I. Springer, Amsterdam, pp 21–32
Nie F, Xiang S, Jia Y, Zhang C (2009) Semi-supervised orthogonal discriminant analysis via label propagation. Pattern Recogn 42(11):2615–2627
Nie F, Xu D, Tsang IW, Zhang C (2010) Flexible manifold embedding: a framework for semi-supervised and unsupervised dimension reduction. IEEE Trans Image Process 19(7):1921–1932
Nie F, Xu D, Li X, Xiang S (2011) Semi-supervised dimensionality reduction and classification through virtual label regression. IEEE Trans Syst Man Cybern Part B 41(3):675–685
Nie F, Yuan J, Huang H (2014) Optimal mean robust principal component analysis. In: Proceedings of the 31st international conference on machine learning (ICML-14), pp 1062–1070
Roweis ST, Saul LK (2000) Nonlinear dimensionality reduction by locally linear embedding. Science 290(5500):2323–2326
Tenenbaum JB, De Silva V, Langford JC (2000) A global geometric framework for nonlinear dimensionality reduction. Science 290(5500):2319–2323
Wang SJ, Yan S, Yang J, Zhou CG, Fu X (2014) A general exponential framework for dimensionality reduction. IEEE Trans Image Process 23(2):920–930
Wright J, Ganesh A, Rao S, Peng Y, Ma Y (2009) Robust principal component analysis: exact recovery of corrupted low-rank matrices via convex optimization. In: Advances in neural information processing systems, pp 2080–2088
Wu M, Yu K, Yu S, Schölkopf B (2007) Local learning projections. In: Proceedings of the 24th international conference on machine learning. ACM, New York, pp 1039–1046

Xu D, Yan S, Lin S, Huang TS, Chang SF (2009) Enhancing bilinear subspace learning by element rearrangement. IEEE Trans Pattern Anal Mach Intell 31(10):1913–1920
Yan S, Xu D, Zhang B, Zhang HJ, Yang Q, Lin S (2007) Graph embedding and extensions: a general framework for dimensionality reduction. IEEE Trans Pattern Anal Mach Intell 29(1):40–51
Yang Y, Xu D, Nie F, Yan S, Zhuang Y (2010) Image clustering using local discriminant models and global integration. IEEE Trans Image Process 19(10):2761–2773
Yang J, Chu D, Zhang L, Xu Y, Yang J (2013) Sparse representation classifier steered discriminative projection with applications to face recognition. IEEE Trans Neural Networks Learn Syst 24(7):1023–1035
Zhang T, Tao D, Li X, Yang J (2009) Patch alignment for dimensionality reduction. IEEE Trans Knowl Data Eng 21(9):1299–1313
Zhang L, Zhou WD, Chang PC, Liu J, Yan Z, Wang T, Li FZ (2012) Kernel sparse representation-based classifier. IEEE Trans Signal Process 60(4):1684–1695
Zhou T, Tao D (2013) Double shrinking sparse dimension reduction. IEEE Trans Image Process 22(1):244–257
Zhou D, Bousquet O, Lal TN, Weston J, Schölkopf B (2004) Learning with local and global consistency. Adv Neural Inform Process Syst 16(16):321–328
Zhu X, Ghahramani Z, Lafferty J (2003) Semi-supervised learning using Gaussian fields and harmonic functions. In: ICML, vol 3, pp. 912–919
Zhuang L, Gao H, Lin Z, Ma Y, Zhang X, Yu N (2012) Non-negative low rank and sparse graph for semi-supervised learning. In: IEEE conference on computer vision and pattern recognition (CVPR), 2012. IEEE, New York, pp 2328–2335
Zuo W, Zhang D, Yang J, Wang K (2006) BDPCA plus LDA: a novel fast feature extraction technique for face recognition. IEEE Trans Syst Man Cybern Part B 36(4):946–953