



Dynamic graph learning for spectral feature selection

Wei Zheng¹ · Xiaofeng Zhu¹ · Yonghua Zhu² ·
Rongyao Hu¹ · Cong Lei¹

Received: 26 July 2017 / Revised: 20 September 2017 / Accepted: 28 September 2017 /

Published online: 14 October 2017

© Springer Science+Business Media, LLC 2017

Abstract Previous spectral feature selection methods generate the similarity graph via ignoring the negative effect of noise and redundancy of the original feature space, and ignoring the association between graph matrix learning and feature selection, so that easily producing suboptimal results. To address these issues, this paper joints graph learning and feature selection in a framework to obtain optimal selected performance. More specifically, we use the least square loss function and an $\ell_{2,1}$ -norm regularization to remove the effect of noisy and redundancy features, and use the resulting local correlations among the features to dynamically learn a graph matrix from a low-dimensional space of original data. Experimental results on real data sets show that our method outperforms the state-of-the-art feature selection methods for classification tasks.

Keywords Graph learning · Optimization · Spectral feature selection

1 Introduction

With the development of science and technology, big data have appeared in various fields, such as knowledge discovery and pattern recognition, and cause a surge of the size of the

✉ Xiaofeng Zhu
seanzhuxf@gmail.com

Wei Zheng
zwgxnu@163.com

¹ Guangxi Key Lab of Multi-source Information Mining and Security, Guangxi Normal University, Guilin, 541004, Guangxi, People's Republic of China

² School of Computer, Electronics and Information of Guangxi University, Nanning, Guangxi, 530004, People's Republic of China

database in either the samples or the dimensions of features. The issue of a large number of samples can be handled by sampling methods or others [4, 13, 22, 43, 49, 52]. Moreover, the advanced techniques still are desired to remove harmful effect of redundant and noisy features in high-dimensional data, with the expectation of accelerating execution time, reducing the storage cost, and enhancing the performance of learning models. As a consequence, dimensional reduction is a widely used way that aims to reduce the number of the features through removing uninformative features of high-dimensional data [16, 36, 41, 48, 50, 53].

Dimensionality reduction techniques can be divided into two categories, i.e., feature selection [14, 20, 40] and subspace learning [9, 10, 19, 38]. Feature selection methods remaining important features from the original feature space can output interpretable models. Subspace learning methods projecting the original feature space into a new subspace outputs the robust models of dimensionality reduction [44]. Spectral feature selection methods construct a framework which includes feature selection and subspace learning to obtain the interpretable and robust models, and then has been drawing a lot of attention in machine learning and data mining [24, 30].

Recently, a large number of spectral feature selection methods have been proposed [34, 35, 45, 51, 54, 55]. Cai et al. first conducts eigenvalue decomposition on the graph matrix constructing from original data to obtain the graph representation of the data, and then calculates the minimal error between the derived graph representation and the original data [2]. Gu et al. [5] and Shi et al. [25] first construct the graph matrix by the neighbor relationship among the data points, followed by selecting the useful features through the sparse regression model to find the importance feature.

It has been shown that the local structure may achieve better performance than the global structure [8]. Thus many spectral feature selection methods discovering the optimal local structure have been proposed [17, 23]. Previous spectral feature selection methods have two separated steps. Specifically, the first step is to explore the local structure to construct the graph matrix, while the second step is to select the important features by a sparse regression. However, there are at least two main issues in previous methods. Firstly, traditional spectral feature selection methods usually obtain the graph matrix on the original high-dimensional space which usually outputs an unsatisfactory neighbor relationship. Secondly, previous spectral feature selection methods select the features and build the graph matrix separately. In this way, the graph matrix is obtained from original data and remains unchanged for succeeding processes. However, the redundancy and noisy of original data may make the graph matrix low-quality. The low-quality graph matrix will preserve the imperfect local structure, and ultimately can not output optimal result.

In this paper, we propose a new spectral feature selection model to jointly learns the dynamical graph and sparse feature selection to select the important features from the optimal subspace of the high-dimensional original data. To do this, our method takes the dynamic neighborhood correlation among the samples into account to preserve optimum local structures of the data. This aims at getting a robust spectral feature selection model. More specifically, we obtain the feature weight matrix by a least square regression between original data and its labels, with a least square loss function, and use an $\ell_{2,1}$ -norm regularization to penalize the weight matrix. We also push an orthogonal constraint on the weight matrix to improve the accuracy for feature selection vis conducting subspace learning. We further propose to build the dynamic graph matrix, i.e., dynamically capturing the nearest neighbor relationship among data points. Moreover, we integrate the least square regression, the $\ell_{2,1}$ -norm regularization, the dynamic graph matrix learning, and the orthogonal

constraint in a unified framework. As a result, the redundancy and noisy could be removed from original data, which should reduce the influence of constructed graph matrix. The proposed spectral feature selection model can easily select the important features because of contain an interpretable and robust low-dimensional subspace.

By comparing previous spectral feature selection methods, we list the main contributions of our proposed method as follow:

- Our proposed method performs the sparse regression and the dynamic graph learning simultaneously. In this way the feature weight matrix deriving from the regression can remove the effect of redundancy and noisy to learn more a reliable similarity matrix. Meanwhile, reliable graph matrix can improve the performance of the sparse regression. As a consequence, our proposed method learns the optimal local structure of the low-dimensional data through an alternative optimization method to ultimately select important features.
- Our proposed method proposes a reasonable constraint, where the feature weight matrix derived by local structure and the sparse regression can be more accurate while using the orthogonal constraint to constrain feature weight matrix.

2 Related work

Subspace learning methods project high-dimensional data into its low-dimensional space to reduce the dimensionality of the high-dimensional data. Popular subspace learning methods include linear projected (such as Principal Component Analysis(PCA) [32] and Locality Preserving Projection(LPP)) and nonlinear projected (such as kernel LDA [37] and kernel CCA [47]). Feature selection methods tend to find a subset of the features that best express the samples from original features.

In this section, we first review recent studies of feature selection methods and then provide a brief analysis to previous spectral feature selection methods.

2.1 Feature selection

Feature selection is widely applied in real applications because of its interpretable ability [33, 35, 39, 42, 46]. We may partition previous feature selection into three subgroups, i.e., filter methods, wrapper methods, and embedded methods. Filter methods [6, 12, 21] which are independent on the learning models use proxy measures to rank the features and select valuable features. Wrapper methods [7, 11] search a best subset of feature guided by the performance of a given blurring model, so that achieving better feature selection results but higher computational cost than filter methods. Embedded methods [18, 28] are different from previous two kinds of feature selection methods, via integrating the process of feature selection into the learning models, and thus achieving the effect of improving the performance and reducing the computational cost. In particular, the sparsity regularization based embedded feature selection methods may achieve outstanding feature selection performance. More specifically, the $s\ell_1$ -norm regularization is widely applied in different feature selection models such as LASSO [27] and sparse SVM [15], aiming at preventing the issue of over-fitting and achieving the sparse results. On the other hand, the $\ell_{2,1}$ -norm regularization further enhances the capability of the sparsity and have been applied in multi-task learning or multi-class learning [16].

2.2 Spectral feature selection

The spectral feature selection methods are usually constituted by two main parts, i.e., graph matrix learning and sparse feature selection, respectively. According to the patterns of the graph matrix learning, we divide the previous spectral feature selection methods into two categories, i.e., fixed learning methods and dynamic learning methods, respectively.

Fixed learning methods construct the graph matrix learning from original data firstly, and then use a sparsity regularization to conduct sparse feature selection. For example, Unsupervised Feature Selection for Multi-Cluster Data (MCFS) method [2] and Joint Feature Selection and Subspace Learning (FSSL) method [5] use the ℓ_1 -norm regularization and the $\ell_{2,1}$ -norm regularization, respectively, to conduct spectral feature learning via first learning the graph matrix and then using the sparse feature selection framework. MCFS method utilizes group sparsity i.e., the $\ell_{2,1}$ -norm regularization, to consider the global correlation among the data, and thus has better performance than FSSL. Other fixed learning methods integrate the graph matrix learning into the feature selection to improve the reliable of feature weight matrix. For example, the Robust Spectral Learning for Unsupervised Feature Selection (RSLUFS) method utilizes the local kernel regression as predictor to capture the structure of the data [24]. The Non-Negative Spectral Learning and Sparse Regression-Based Dual-Graph Regularized Feature Selection (NSSRD) method utilizes the Gaussian function to structure dual-graph from data space and feature space [23].

Dynamic learning methods learn graph matrix highly depending on the feature weight matrix, which is obtained from the low-dimensional space of high-dimensional data, so that they obtain the optimal solution by iteratively update the graph matrix and feature weight matrix. For example, the Structured Optimal Graph Feature Selection (SOGFS) method [18] utilizes dynamic graph learning to capture the neighbor relationship and uses $\ell_{2,1}$ -norm regularization to penalize the projection matrix.

3 Methodology

In this section, we first list notations and definition of norms used in this paper in Section 3.1, and then the detail of our method is described in Sections 3.2 and 3.3. Finally, the optimization problem of the proposed method is elaborated in Sections 3.4 and 3.5.

3.1 Notations

In this paper, we use boldface uppercase letters, boldface lowercase letters, and normal italic letters, respectively, to denote matrices, vectors, and scalars. We denote $\mathbf{X} = \{x_i\}_{i=1}^n \in \mathbb{R}^{n \times d}$ as the feature matrix of the samples, where i -th row and j -th column of \mathbf{X} are denoted as x_i and x_j , respectively. The Frobenius norm of \mathbf{X} is denoted as $\|\mathbf{X}\|_F = \sqrt{\sum_{ij} |x_{ij}|^2}$ and the $\ell_{2,1}$ -norm of \mathbf{X} is denoted as $\|\mathbf{X}\|_{2,1} = \sum_i \sqrt{\sum_{ij} x_{ij}^2}$. Furthermore, we denote $tr(\mathbf{X})$ as the trace of \mathbf{X} , \mathbf{X}^T as the transpose of \mathbf{X} and \mathbf{X}^{-1} as the inverse of \mathbf{X} , respectively.

3.2 Dynamic graph learning

The previous literature has shown that the graph matrix learning can capture geometrical information of the samples to enhance the performance of dimensionality reduction.

Nevertheless, the redundancy and noise from samples or features can make the graph matrix unreliable and inaccurate. Thus, in this paper, we utilize the model of dynamic graph learning to preserve the optimal local structure. Given a data matrix \mathbf{X} , the similarity graph matrix \mathbf{S} is initialized as follows:

$$s_{ij} = \begin{cases} f(x^i, x^j) & \text{if } x^i \in \mathbf{N}(x^j) \\ & \text{or } x^j \in \mathbf{N}(x^i) \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where $\mathbf{N}(x)$ represents the k -nearest neighbors set of sample x . If the i -th sample x^i is contained in k -nearest neighbors set of sample x^j , then the value of \mathbf{S}_{ij} (i.e., the element of matrix \mathbf{S}) is determined by the heat kernel (i.e., $f(x^i, x^j) = \exp(-\frac{\|x^i - x^j\|_2^2}{2\sigma^2})$, where σ is a tuning parameter), otherwise $s_{ij} = 0$. According to the common sense, closer data points on the sample space have greater similarity, thus \mathbf{S}_{ij} can be revised by the square of Euclidean distance between x^i and x^j (i.e., $\|x^i - x^j\|_2^2$). Therefore, obtain the determining value of \mathbf{S}_{ij} from original data space can be seen as solving:

$$\min_{\mathbf{S}} \sum_{i,j} \|x^i - x^j\|_2^2 s_{ij} \tag{2}$$

Although (2) learns a fixed graph matrix \mathbf{S} from the high-dimensional space of original data to preserves the local structure of data, but it ignores the negative effect from redundancy and noisy. If original data contain noisy and redundancy (it always true in real world data), the graph matrix which learn from original data will become unreliable and inaccurate. Moreover, the quality of \mathbf{S} has been proved very sensitive to the tuning parameter σ . This led us to learn the graph matrix from low-dimensional space (it contained the possibility of redundancy and noisy is small) and to decrease the dependency on parameters (i.e., reduce the number of tuning parameters). To address this, we utilize a weight matrix project the original data into its low-dimensional space and learn the graph matrix by iteratively optimization instead of using the heat kernel function, which depend the tuning parameter σ . We rewrite (2) as follows:

$$\begin{aligned} \min_{\mathbf{S}, \mathbf{W}} \sum_{i,j} \|x^i \mathbf{W} - x^j \mathbf{W}\|_2^2 s_{ij} + \alpha \|s_i\|_2^2, \\ \text{s.t.}, \forall i, s_i^T \mathbf{1} = 1, s_{i,i} = 0, \\ s_{i,j} \geq 0 \text{ if } i \in \mathbf{N}(j), \text{ otherwise } 0 \end{aligned} \tag{3}$$

where α is a regularization parameter, the ℓ_2 -norm of s_i (i.e., $\|s_i\|_2^2$) is used to avoid the trivial solution and add a prior of uniform distribution, $\mathbf{1}$ represent a vector of all-one-element. By solving problem (3), the reliable graph matrix will be learning, thus obtained the information of local structure is more accurate.

3.3 Dynamic graph learning for feature selection

Let the $\mathbf{Y} = \{y_i\}_{i=1}^n \in \mathbb{R}^{n \times c}$ as the response matrix, where represents the label i -th sample. Motivated by the generally used supervised learning method, we utilize the regression

to fit each sample to its label and adding a regularization, thus have the following sparse regression model:

$$\min_{\mathbf{W}} \beta \|\mathbf{Y} - \mathbf{X}\mathbf{W}\|_{\rho^*} + \gamma \|\mathbf{W}\|_{\omega^*} \tag{4}$$

where $\mathbf{W} = [w_1, w_2, \dots, w_d] \in \mathbb{R}^{d \times c}$ is the feature weight matrix. Although there are many different case to choose ρ^* and ω^* , in this paper, we select $\|\cdot\|_F^2$ as ρ^* and $\|\cdot\|_{2,1}$ as ω^* , respectively. The reason is that the F-norm regularization has steady fitting performance and the $\ell_{2,1}$ -norm regularization may conduct the row sparsity. In particular, $\|\mathbf{W}\|_{2,1}$ can be easily optimized. The sparse regression model is rewritten as follow:

$$\min_{\mathbf{W}} \beta \|\mathbf{Y} - \mathbf{X}\mathbf{W}\|_F^2 + \gamma \|\mathbf{W}\|_{2,1} \tag{5}$$

When graph matrix learning and sparse feature selection are jointly performed, the feature weight matrix will affect the graph matrix. This is, the feature weight matrix not only select the important feature, but also conduct the graph matrix learning. We combine the objective function (3) with (5) as:

$$\begin{aligned} \min_{\mathbf{S}, \mathbf{W}} \quad & \sum_{i,j}^n \|x^i \mathbf{W} - x^j \mathbf{W}\|_2^2 s_{ij} + \alpha \|\mathbf{S}\|_F^2 \\ & + \beta \|\mathbf{Y} - \mathbf{X}\mathbf{W}\|_F^2 + \gamma \|\mathbf{W}\|_{2,1} \\ \text{s.t.}, \quad & \forall i, s_i^T \mathbf{1} = 1, s_{i,i} = 0, \\ & s_{i,j} \geq 0 \text{ if } i \in \mathbf{N}(j), \text{ otherwise } 0 \end{aligned} \tag{6}$$

In feature selection tasks, PCA method can not be used as high-dimensional data preprocessing. Therefore, we use the constraint $\mathbf{W}^T \mathbf{W} = \mathbf{I}$ to promote the performance of feature selection [18]. Our final objective function as follows:

$$\begin{aligned} \min_{\mathbf{S}, \mathbf{W}} \quad & \sum_{i,j}^n \|x^i \mathbf{W} - x^j \mathbf{W}\|_2^2 s_{ij} + \alpha \|\mathbf{S}\|_F^2 \\ & + \beta \|\mathbf{Y} - \mathbf{X}\mathbf{W}\|_F^2 + \gamma \|\mathbf{W}\|_{2,1} \\ \text{s.t.}, \quad & \forall i, s_i^T \mathbf{1} = 1, s_{i,i} = 0, s_{i,j} \geq 0 \text{ if } i \in \mathbf{N}(j) \\ & \text{otherwise } 0, \mathbf{W}^T \mathbf{W} = \mathbf{I} \end{aligned} \tag{7}$$

Equation (7) iteratively updates the graph matrix \mathbf{S} and the feature weight matrix until achieve their individually optimal result. In this way, the optimal local structure can be preserved by iteratively updated graph matrix, and obtain excellent performance in feature selection model ultimately. As a consequence, given the optimal \mathbf{W} , we sort the value of $\|w_i\|_2, i = 1, 2, \dots, d$ in descending order, and we are based on the top r ranked ℓ_2 -norm values to select top r features as the final result of our proposed method.

3.4 Optimization

In this paper, we employ the alternative optimization strategy that fixed a variable while iteratively optimizing the others until the algorithm converges. We list the pseudo code in Algorithm 1.

- 1). Update \mathbf{S} by fixed \mathbf{W}

The fixed \mathbf{W} can be seen as constant, (7) becomes:

$$\begin{aligned} \min_{\mathbf{S}, \mathbf{W}} \quad & \sum_{i,j}^n \|x^i \mathbf{W} - x^j \mathbf{W}\|_2^2 s_{ij} + \alpha \sum_{i,j}^n s_{i,j}^2 \\ \text{s.t.}, \quad & \forall i, s_i^T \mathbf{1} = 1, s_{i,i} = 0, \\ & s_{i,j} \geq 0 \text{ if } i \in \mathbf{N}(j), \text{ otherwise } 0 \end{aligned} \tag{8}$$

For easy the optimization (8), we optimize each vector $s_i, i = 1, \dots, n$, independently instead of the optimization of \mathbf{S} , the optimization problem be changed as follows:

$$\min_{s_i^T \mathbf{1}=1, s_{i,i}=0, s_{i,j} \geq 0} \sum_{i,j}^n \left(\|x^i \mathbf{W} - x^j \mathbf{W}\|_2^2 s_{i,j} + \alpha s_{i,j}^2 \right) \tag{9}$$

We calculate the Euclidean distance between all samples to yield k -nearest neighbors, and then set the values of $s_{i,j}$ by optimizing (9) if the j -th sample is one of nearest neighbors of the i -th sample, otherwise, the value of $s_{i,j}$ is 0. By denoting $\mathbf{G} \in \mathbb{R}^{n \times n}$, where $g_i = \sum_j \|x^i \mathbf{W} - x^j \mathbf{W}\|_2^2$, we rewrite (9) as follows:

$$\min_{s_i^T \mathbf{1}=1, s_{i,i}=0, s_{i,j} \geq 0} \sum_i^n \left\| s_i + \frac{1}{2\alpha} g_i \right\|_2^2 \tag{10}$$

The Lagrangian function of (10) is:

$$\min_{s_i, \lambda, \nu} \left\| s_i + \frac{1}{2\alpha} g_i \right\|_2^2 - \lambda (s_i^T \mathbf{1} - 1) - \nu^T s_i \tag{11}$$

where λ and ν are the Lagrangian multipliers. The optimal solution of s_i yield by the Karush-Kuhn-Tucker (KKT) conditions [1] is:

$$s_{i,j} = \left(-\frac{1}{2\alpha} g_{i,j} + \lambda \right)_+ \tag{12}$$

2). Udata \mathbf{W} by fixed \mathbf{S}

By fixed \mathbf{S} , we have the following objective function:

$$\begin{aligned} \min_{\mathbf{S}, \mathbf{W}} \quad & \sum_{i,j}^n \|x^i \mathbf{W} - x^j \mathbf{W}\|_2^2 s_{ij} \\ & + \beta \|\mathbf{Y} - \mathbf{XW}\|_F^2 + \gamma \|\mathbf{W}\|_{2,1} \\ \text{s.t.}, \quad & \mathbf{W}^T \mathbf{W} = \mathbf{I} \end{aligned} \tag{13}$$

The optimization of (13) on \mathbf{W} is nonsmooth but convex due to the regularization term $\|\mathbf{W}\|_{2,1}$. Thus we employ the the Iteratively Reweighted Least Square (IRLS) [3] to optimize (13) on the variable \mathbf{W} , iteratively optimizing \mathbf{W} until (13) converges. We change (9) as:

$$\begin{aligned} \min_{\mathbf{W}} \quad & tr(\mathbf{W}^T \mathbf{X}^T \mathbf{LXW}) \\ & + \beta \|\mathbf{Y} - \mathbf{XW}\|_F^2 + \gamma tr(\mathbf{W}^T \mathbf{QW}) \\ \text{s.t.} \quad & \mathbf{W}^T \mathbf{W} = \mathbf{I} \end{aligned} \tag{14}$$

where \mathbf{L} is a Laplacian matrix and \mathbf{P} is a diagonal matrix which i -th element $p_{ii} = \sum_{j=1}^n s_{i,j}$. $\mathbf{Q} \in \mathbb{R}^{d \times d}$ is diagonal matrix with its i -th element define as:

$$q_{ii} = \frac{1}{2\|w^i\|_2^2}, i = 1, \dots, d \tag{15}$$

where w^i is the i -th row of \mathbf{W} . Equation (14) contain an orthogonal constraint, we use [31] to solve it and list the pseudo code of the optimization algorithm of \mathbf{W} in Algorithm 2, where the derivative of (10) is $\nabla F = \mathbf{X}^T \mathbf{L} \mathbf{X} \mathbf{W} + \beta(\mathbf{X}^T \mathbf{X} \mathbf{W} - \mathbf{X}^T \mathbf{Y}) + \gamma \mathbf{Q} \mathbf{W}$.

Algorithm 1 The pseudo code of solving (7)

- Input:** $\mathbf{X} \in \mathbb{R}^{n \times d}, \mathbf{Y} \in \mathbb{R}^{n \times c}, \alpha, \beta, \gamma$, and k ;
Output: $\mathbf{W} \in \mathbb{R}^{d \times c}, \mathbf{S} \in \mathbb{R}^{n \times n}$;
 1. Calculate k nearest neighbors of all samples;
 2. Use (3) to \mathbf{S} , where \mathbf{W} is identity matrix;
 3. **repeat:**
 3.1 Update \mathbf{W} via Algorithm 2;
 3.2 Update \mathbf{S} by (12);
 3.3 Calculate $\mathbf{L} = \mathbf{P} - \frac{\mathbf{S}^T + \mathbf{S}}{2}$;
until (7) convergence
-

Algorithm 2 The pseudo code of solving \mathbf{W}

- Input:** $\mathbf{X} \in \mathbb{R}^{n \times d}, \mathbf{Y} \in \mathbb{R}^{n \times c}, \alpha, \beta, \gamma$, and k ;
Output: $\mathbf{W} \in \mathbb{R}^{d \times c}$;
 1. Initialize $\mathbf{W}^{(t)}$;
 2. Initialize $t = 1$;
 3. **repeat:**
 3.1 $\mathbf{H} = \nabla F^{(t)} \mathbf{W}^{(t)T} - \mathbf{W}^{(t)} \nabla F^{(t)T}$;
 3.2 τ is non-monotonic grid search;
 3.3 Calculate $\mathbf{W}^{(t+1)} = (\mathbf{I} - \frac{\tau}{2} \mathbf{H})^{-1} (\mathbf{I} + \frac{\tau}{2} \mathbf{H}) \mathbf{X}$;
until convergence
-

3.5 Convergence analysis, complexity, and parameters’ determination

3.5.1 Proof of convergence

Wen and Yin [31] proved the convergence of Algorithm 2. Thus, we prove the convergence of Algorithm 1. We have the following Lemma:

Lemma 1 *The inequality*

$$\sqrt{u} - \frac{u}{2\sqrt{v}} \leq \sqrt{v} - \frac{u}{2\sqrt{v}} \tag{16}$$

is always hold for all positive real numbers of u and v, according to the literatures [31].

Theorem 1 *The objective function value of (7) monotonically decreases until Algorithm 1 converges.*

Proof After the t -th iteration, we have obtained the current optimal $\mathbf{W}^{(t)}$ and $\mathbf{S}^{(t)}$. In the $(t + 1)$ -th iteration, we fix $\mathbf{W}^{(t)}$ to optimize $\mathbf{S}^{(t)}$. According to (12), we know that $s_{i,j}^{(t+1)}$ has global solution for all $i, j = 1, \dots, n$. Thus we have the inequality as follows:

$$\begin{aligned} & \sum_{i,j}^n \|x^i \mathbf{W}^{(t)} - x^j \mathbf{W}^{(t)}\|_{2s_{i,j}^{(t+1)}}^2 \\ & + \alpha \sum_i^n \|s_i^{(t+1)}\|_2^2 + \beta \|\mathbf{Y} - \mathbf{XW}^{(t)}\|_F^2 \\ & + \gamma \|\mathbf{W}^{(t)}\|_{2,1} \\ & \leq \sum_{i,j}^n \|x^i \mathbf{W}^{(t)} - x^j \mathbf{W}^{(t)}\|_{2s_{i,j}^{(t)}}^2 \\ & + \alpha \sum_i^n \|s_i^{(t)}\|_2^2 + \beta \|\mathbf{Y} - \mathbf{XW}^{(t)}\|_F^2 \\ & + \gamma \|\mathbf{W}^{(t)}\|_{2,1} \end{aligned} \tag{17}$$

When update $\mathbf{W}^{(t+1)}$ by fixing $\mathbf{S}^{(t+1)}$, we get the following inequality according to Theorem 1:

$$\begin{aligned} & \sum_{i,j}^n \|x^i \mathbf{W}^{(t+1)} - x^j \mathbf{W}^{(t+1)}\|_{2s_{i,j}^{(t+1)}}^2 \\ & + \alpha \sum_i^n \|s_i^{(t+1)}\|_2^2 + \beta \|\mathbf{Y} - \mathbf{XW}^{(t+1)}\|_F^2 \\ & + \gamma \|\mathbf{W}^{(t+1)}\|_{2,1} \\ & \leq \sum_{i,j}^n \|x^i \mathbf{W}^{(t)} - x^j \mathbf{W}^{(t)}\|_{2s_{i,j}^{(t+1)}}^2 \\ & + \alpha \sum_i^n \|s_i^{(t+1)}\|_2^2 + \beta \|\mathbf{Y} - \mathbf{XW}^{(t)}\|_F^2 \\ & + \gamma \|\mathbf{W}^{(t)}\|_{2,1} \end{aligned} \tag{18}$$

Through the integration of (17) and (18), we obtain:

$$\begin{aligned} & \sum_{i,j}^n \|x^i \mathbf{W}^{(t+1)} - x^j \mathbf{W}^{(t+1)}\|_{2s_{i,j}^{(t+1)}}^2 \\ & + \alpha \sum_i^n \|s_i^{(t+1)}\|_2^2 + \beta \|\mathbf{Y} - \mathbf{XW}^{(t+1)}\|_F^2 \\ & + \gamma \|\mathbf{W}^{(t+1)}\|_{2,1} \\ & \leq \sum_{i,j}^n \|x^i \mathbf{W}^{(t)} - x^j \mathbf{W}^{(t)}\|_{2s_{i,j}^{(t)}}^2 \\ & + \alpha \sum_i^n \|s_i^{(t)}\|_2^2 + \beta \|\mathbf{Y} - \mathbf{XW}^{(t)}\|_F^2 \\ & + \gamma \|\mathbf{W}^{(t)}\|_{2,1} \end{aligned} \tag{19}$$

According the (19), we can see that the objective function values of (7) are reduced after each iteration of the Algorithm 1. Thus, Theorem 1 has been proved. \square

3.5.2 Complexity analysis

In each iteration, the time cost of Algorithm 1 focuses on the computation cost of $\mathbf{X}^T \mathbf{LXW} + \beta(\mathbf{X}^T \mathbf{XW} - \mathbf{X}^T \mathbf{Y}) + \gamma \mathbf{QW}$ in Algorithm 2 and $f_{i,j}$ in (12), and their corresponding complexity are $O(n^2d)$ and $O(nd^2)$, where n, d , respectively, are the number of the data points and the features. In our experiments, our method usually converges within 30 iterations, so the time complexity of Algorithm is $\max \{O(n^2d), O(nd^2)\}$.

3.5.3 Parameters' determination

The size of value of the parameter α determines the number of nearest neighbors of samples. Specifically, $\alpha = 0$ means the number of nearest neighbors k is 1. $\alpha \rightarrow \infty$ means the number of nearest neighbors k is n (n is the number of samples). Without loss of generality, we suppose $g_i = \{g_{i,1}, \dots, g_{i,n}\}$ as a ascending order of $g_i, i = 1, \dots, n$, according to (12) we know $s_{ik} > 0$ and $s_{i,k+1} = 0$. Therefore, we have

$$\begin{cases} -\frac{1}{2\alpha}g_{i,k+1} + \lambda \leq 0 \\ -\frac{1}{2\alpha}g_{i,k} + \lambda > 0 \end{cases} \tag{20}$$

Based on the constraint $s_i^T \mathbf{1} = 1$, we have

$$\begin{aligned} \sum_{j=1}^k \left(-\frac{1}{2\alpha}g_{i,j} + \lambda \right) &= 1 \\ \Rightarrow \lambda &= \frac{1}{k} + \frac{1}{2k\alpha} \sum_{j=1}^k g_{i,j} \end{aligned} \tag{21}$$

Combining (20) and (21), we have the following inequality:

$$\frac{k}{2}g_{i,k} - \frac{1}{2} \sum_{j=1}^k g_{i,j} < \alpha < \frac{k}{2}g_{i,k+1} - \frac{1}{2} \sum_{j=1}^k g_{i,j} \tag{22}$$

In order to obtain an optimal solution s_i which has k nonzero elements, we set α as:

$$\alpha_i = \frac{k}{2}g_{i,k+1} - \frac{1}{2} \sum_{j=1}^k g_{i,j} \tag{23}$$

The mean of $\alpha_1, \dots, \alpha_n$ could be set as final α . We have

$$\alpha = \frac{1}{n} \sum_{i=1}^n \left(\frac{k}{2}g_{i,k+1} - \frac{1}{2} \sum_{j=1}^k g_{i,j} \right) \tag{24}$$

After fixing α , the parameters k, β, γ in (7) still needs to tuning. In this paper, we follow the literature [29] to determine the value of k , and employ a cross-validation method to estimate other.

4 Experiments

In this section, we use the classification performance to evaluate our proposed method and compare with five comparison methods on eight data sets.

4.1 Data sets

The data sets (such as Umist, Ecoli, Cane, and Isolet) and the data sets (such as Colon, Coil, Or1) come from UCI Machine Learning Repository¹ and the website of Feature Selection Data sets.² We use the data set Lung from [26]. The data sets of our used can be divided into text data (Cane, Isolet), biological data (Colon, Ecoli, Lung) and image data (Coil, Or1, Umist). The number of data features is greater than 343, where half of all data sets our used (such as Colon, Lung, Coil, Or1) exceed 1000 dimension. The detail of data sets is listed in Table 1.

¹<http://archive.ics.uci.edu/ml/>.

²<http://featureselection.asu.edu/datasets.php>.

Table 1 The summarization of the used datasets

Data sets	#(Samples)	#(Features)	#(Classes)
Umist	575	644	20
Cane	1080	856	9
Colon	62	2000	2
Ecoli	336	343	8
Lung	203	3312	5
Isolet	1559	618	26
Coil	1440	1024	20
Orl	400	1024	40

4.2 Comparison methods and experimental setting

We list the detail of the comparison methods as follows:

- Baseline method uses all features to conduct classification tasks via SVM.
- Convex Semi-supervised multi-label Feature Selection (CSFS) use the least square regression to select feature and employs an $\ell_{2,1}$ -norm to conduct row sparsity.
- $\ell_{2,0}$ -norm ALM (FSRobustALM) uses an $\ell_{2,1}$ -norm to deal with the minimization loss problem and tackles the sparse problem with $\ell_{2,0}$ -norm constraint.
- Regularized Self-Representation (RSR) joints the feature-level self-representation loss function and a $\ell_{2,1}$ -norm regularization to flitter the unimportant features.
- Robust Unsupervised Feature Selection (RUFs) uses local learning regularization to learn pseudo cluster label and minimizes the fitting error both of feature learning and label learning by employing an $\ell_{2,1}$ -norm regularization.
- Structured Optimal Graph Feature Selection (SOGFS) selects important features by learn the local structure between the samples from low-dimensional feature space.

For avoided the generalization error on classifier, we compare all methods by used 10-fold cross-validation. We repeated the whole process 10 times to avoid the possible bias during data set partitioning for cross-validation. The final result was computed by averaging results from all experiments. We conduct 5-fold cross-validation on the training data to conduct model selection. That is, we further separated the training data into five parts, where one of parts is used to validate the model built by the left four parts. In the validation step, we search the best parameters' combination by grid search method in the given ranges of the parameters. The best parameters' combination has best classification performance on testing data. We set the value of k and β, γ as 15 and $\{10^{-3}, \dots, 10^3\}$, respectively.

In our experiments, we used the Average Classification Accuracy (ACA) as evaluation metric to evaluate our method. We investigated the robustness of our proposed method based on parameters' sensitivity and convergence of Algorithm 1.

4.3 Experimental result

The ACA result of all methods is revealed in Fig. 1, where the horizontal axis represented the dimension of feature of performing feature selection.

Obviously, compare with other method (such as SOGFS, RUFs, RSR, FSRobustALM, CSFS and Baseline) our method achieved the best performance. For example, our proposed method improved by 12.1% and 17.5%, respectively, compared to CSFS and RSR in data set Lung. Moreover, other observations are as follows.

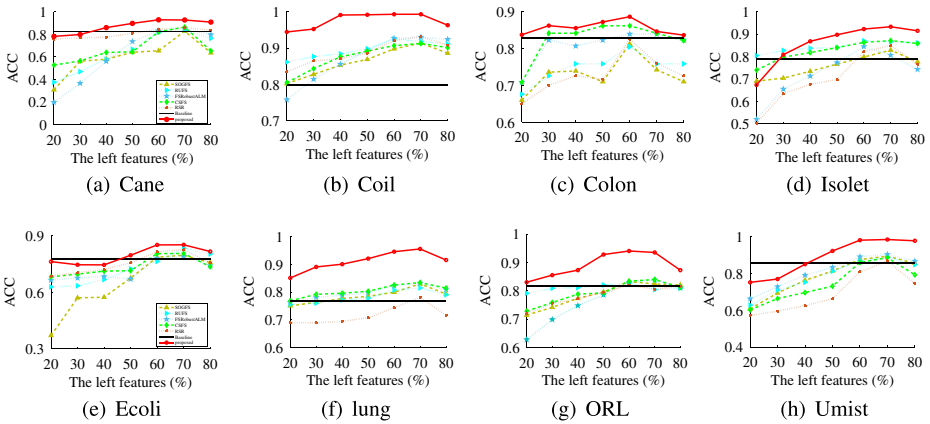


Fig. 1 ACC of all methods on all data sets at different number of selected features

First, with the increase of features number, the classification performance of all feature selection methods first increase to optimal and then start to fall. For example, while selection the 20% and 60% features of sample, the ACA result were about 83.7% and 88.6%, respectively, but while keeping the features as 80% features of sample, the ACA went down to 83.6% at the data set Colon. High-dimensional data which contain redundancy and noisy may affected the classification performance, so conducted the feature selection is necessary.

Second, most of feature selection methods have outstanding performance than baseline, which conduct classification by use all features, where our proposed method improved on average by 15.2% compared to Baseline. This verified that feature selection is necessary to deal with high-dimensional data again.

4.4 Parameters’ sensitivity

We set the range of the parameters β and γ as $\{10^{-3}, 10^{-2}, \dots, 10^3\}$, and the result listed by Fig. 2.

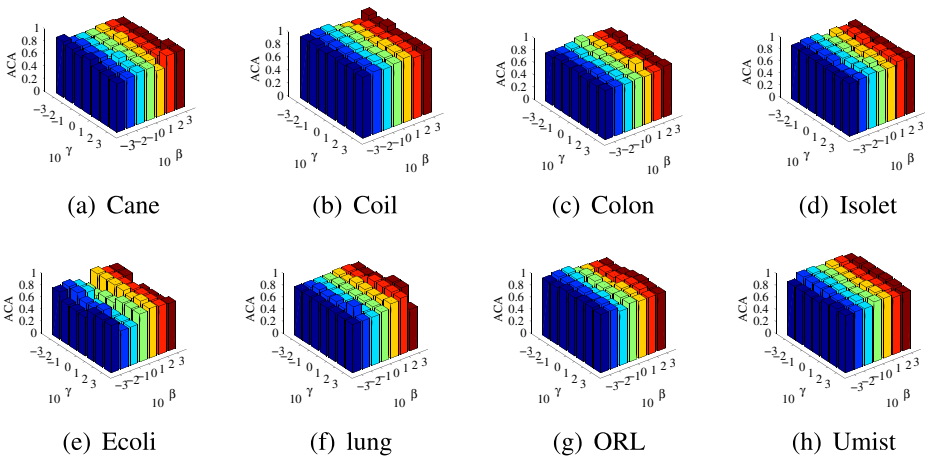


Fig. 2 ACC result of our proposed method at different parameters’ setting

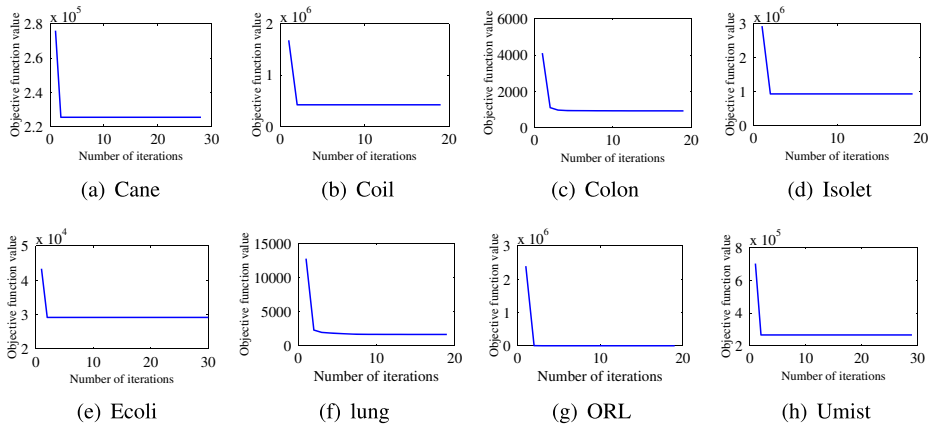


Fig. 3 The convergence of our proposed Algorithm 1

From Fig. 2 we can see that our proposed method is sensitive to the parameters’ setting. Namely, the best classification results rely on the suitable parameter combinations. Hence, tuning the parameters is necessary to our method. More specifically, β is a control parameter to tune the magnitude between the local structure learning $\sum_{i,j} ||x^i \mathbf{W} - x^j \mathbf{W}||_2^2 s_{ij}$ and least square regression $||\mathbf{Y} - \mathbf{X}\mathbf{W}||_F^2$, while γ is used to adjust the sparsity term $||\mathbf{W}||_{2,1}$. When setting $\beta = 10$ and $\gamma = 10$, our method obtain the best performance on the data sets Colon and Ecoli. However, for the data set Coil, our method achieve the best ACA 99.1% with $\beta = 100$ and $\gamma = 100$.

4.5 Convergence of algorithm

Figure 3 illustrates the variation of the objective values of our proposed method (i.e., Algorithm 1) associated with the increase of the iterations. In experiments, we set the stop criteria as $\frac{||obj(t+1)-obj(t)||_2^2}{obj(t)} \leq 10^{-3}$ to both of Algorithms 1 and 2, where indicate the objective function value of the t -th iteration on (7).

From Fig. 3 we can discover that 1) with iteration of Algorithm 1 the objective function values are monotonically decreases until Algorithm 1 converges; 2) the proposed Algorithm 1 reach the convergence only needed few iterations (i.e., less than 20), which proof the efficient of our method. Moreover, the Algorithm 2 of our proposed also achieves convergence within 30 iterations at all data sets.

5 Conclusion

This paper has proposed a novel spectral feature selection method, which dynamic learning graph matrix and selecting the features simultaneously, to obtain more reliable similarity between the data, we use an orthogonal constraint on \mathbf{W} to our method. Compared with the other feature selection methods, the experimental results on real data sets demonstrated our proposed method achieved the best performance.

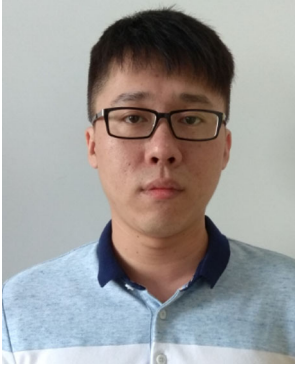
In the future work, we will extend our proposed framework to conduct unsupervised learning on the high-dimensional data since the missing label data sets are often found in real world.

Acknowledgments This work was supported in part by the China Key Research Program (Grant No: 2016YFB1000905), the China 1000-Plan National Distinguished Professorship, the Nation Natural Science Foundation of China (Grants No: 61573270, and 61672177), the Guangxi Natural Science Foundation (Grant No: 2015GXNSFCB139011), the Guang-xi High Institutions Program of Introducing 100 High-Level Overseas Talents, the Guangxi Collaborative Innovation Center of Multi-Source Information Integration and Intelligent Processing, the Guangxi Bagui Teams for Innovation and Research, the Research Fund of Guangxi Key Lab of MIMS (16-A-01-01 and 16-A-01-02), the Guangxi Bagui Teams for Innovation and Research, and Innovation Project of Guangxi Graduate Education under grant XYCSZ2017064, XYCSZ2017067 and YCSW2017065.

References

1. Boyd S, Vandenberghe L, Foybusovich L (2006) Convex optimization. *IEEE Trans Autom Control* 51(11):1859–1859
2. Cai D, Zhang C, He X (2010) Unsupervised feature selection for multi-cluster data. In: *ACM SIGKDD international conference on knowledge discovery and data mining*, pp 333–342
3. Daubechies I, Devore R, Fornasier M, Güntürk CS (2008) Iteratively reweighted least squares minimization for sparse recovery. *Commun Pure Appl Math* 63(1):1–38
4. Gentile C (2001) A new approximate maximal margin classification algorithm. *J Mach Learn Res* 2(2):213–242
5. Gu Q, Li Z, Han J (2011) Joint feature selection and subspace learning. In: *International joint conference on artificial intelligence*, pp 1294–1299
6. Guyon I, Elisseeff A (2003) An introduction to variable feature selection. *J Mach Learn Res* 3:1157–1182
7. Guyon I, Weston J, Barnhill S, Vapnik V (2002) Gene selection for cancer classification using support vector machines. *Mach Learn* 46(1–3):389–422
8. He X, Niyogi P (2003) Locality preserving projections. *Adv Neural Inf Process Syst* 16(1):186–197
9. Hu R, Zhu X, Cheng D, He W, Yan Y, Song J, Zhang S (2017) Graph self-representation method for unsupervised feature selection. *Neurocomputing* 220:130–137
10. Jia Y, Wang Y, Lin H, Jin X, Cheng X (2016) Locally adaptive translation for knowledge graph embedding. In: *Thirtieth AAAI conference on artificial intelligence*, pp 992–998
11. Kohavi R, John GH (1997) Wrappers for feature subset selection. *Artif Intell* 97(1–2):273–324
12. Lewis DD (2013) Feature selection and feature extraction for text categorization. In: *The workshop on speech & natural language*, pp 212–217
13. Ling CX, Yang Q, Wang J, Zhang S (2004) Decision trees with minimal costs. In: *International conference on machine learning*, p 69
14. Liu H, Ma Z, Zhang S, Wu X (2015) Penalized partial least square discriminant analysis with l_1 - norm for multi-label data. *Pattern Recogn* 48(5):1724–1733
15. Mangasarian OL (2006) Exact l_1 -norm support vector machines via unconstrained convex differentiable minimization. *J Mach Learn Res* 7(3):1517–1530
16. Nie F, Huang H, Cai X et al (2010) Efficient and robust feature selection via joint $l_{2,1}$ -norms minimization. In: *International conference on neural information processing systems*, pp 1813–1821
17. Nie F, Xu D, Tsang WH, Zhang C (2010) Flexible manifold embedding: a framework for semi-supervised and unsupervised dimension reduction. *IEEE Trans Image Process* 19(7):1921–1932
18. Nie F, Zhu W, Li X (2016) Unsupervised feature selection with structured graph optimization. In: *Thirtieth AAAI conference on artificial intelligence*, pp 1302–1308
19. Nie F, Zhu W, Li X (2017) Unsupervised large graph embedding. In: *Thirtieth AAAI conference on artificial intelligence*. AAAI Press, pp 2422–2428
20. Peng H, Fan Y (2017) A general framework for sparsity regularized feature selection via iteratively reweighted least square minimization. In: *Thirtieth AAAI conference on artificial intelligence*. AAAI Press, pp 2471–2477
21. Peng H, Long F, Ding C (2005) Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Trans Pattern Anal Mach Intell* 27(8):1226
22. Qin B, Xia Y, Prabhakar S, Tu Y (2009) A rule-based classification algorithm for uncertain data. In: *IEEE international conference on data engineering*, pp 1633–1640
23. Shang R, Wang W, Stolkin R, Jiao L (2017) Non-negative spectral learning and sparse regression-based dual-graph regularized feature selection. *IEEE Trans Cybern PP*(99):1–14

24. Shi L, Du L, Shen YD (2015) Robust spectral learning for unsupervised feature selection. In: IEEE international conference on data mining, pp 977–982
25. Shi X, Guo Z, Lai Z, Yang Y, Bao Z, Zhang D (2015) A framework of joint graph embedding and sparse regression for dimensionality reduction. *IEEE Trans Image Process: A Publication of the IEEE Signal Processing Society* 24(4):1341–1355
26. Singh D, Febbo PG, Ross K, Jackson DG, Manola J, Ladd C, Tamayo P, Renshaw AA, D'Amico AV, Richie JP (2002) Gene expression correlates of clinical prostate cancer behavior. *Cancer Cell* 1(2):203
27. Tibshirani R (2011) Regression shrinkage and selection via the lasso. *J R Stat Soc* 73(3):273–282
28. Wang L, Zhu J, Zou H (2008) Hybrid huberized support vector machines for microarray classification and gene selection. *Bioinformatics* 24(3):412
29. Wang D, Nie F, Huang H (2014) Unsupervised feature selection via unified trace ratio formulation and k-means clustering (track). In: *Ecml/pkdd*, pp 306–321
30. Wang X, Zhang X, Zeng Z et al (2016) Unsupervised spectral feature selection with l_1 -norm graph. volume C, pp 47–54
31. Wen Z, Yin W (2013) A feasible method for optimization with orthogonality constraints. *Math Program* 142(1–2):397–434
32. Wold S, Esbensen K, Geladi P (1987) Principal component analysis. *Chemom Intell Lab Syst* 2(1):37–52
33. Wu X, Zhang S (2003) Synthesizing high-frequency rules from different data sources. *IEEE Trans Knowl Data Eng* 15(2):353–367
34. Wu X, Zhang C, Zhang S (2004) Efficient mining of both positive and negative association rules. *ACM Trans Inf Syst* 22(3):381–405
35. Wu X, Zhang C, Zhang S (2005) Database classification for multi-database mining. *Inf Syst* 30(1):71–88
36. Yan X, Zhang C, Zhang S (2009) Genetic algorithm-based strategy for identifying association rules without specifying actual minimum support. *Expert Syst Appl* 36(2):3066–3076
37. Yang J, Frangi AF, Yang JY, Zhang D, Jin Z (2005) Kpc plus lda: a complete kernel fisher discriminant framework for feature extraction and recognition. *IEEE Trans Pattern Anal Mach Intell* 27(2):230
38. Zhu X, Suk H-I, Lee S-W, Shen D (2016) Subspace regularized sparse multitask learning for multiclass neurodegenerative disease identification. *IEEE Trans Biomed Eng* 63(3):607–618
39. Zhang S, Zhang C (2002) Anytime mining for multiuser applications. *IEEE Trans Syst Man Cybern Part A Syst Hum* 32(4):515–521
40. Zhang S, Zhang C, Yang Q (1999) Data preparation for data mining. Academic, New York
41. Zhang S, Wu X, Zhang C (2003) Multi-database mining. *IEEE Comput Intell Bull* 2(1):5–13
42. Zhang S, Zhang C, Yan X (2003) Post-mining: maintenance of association rules by weighting? *Inf Syst* 28(7):691–707
43. Zhang S, Qin Z, Ling CX, Sheng S (2005) Missing is useful?: missing values in cost-sensitive decision trees. *IEEE Trans Knowl Data Eng* 17(12):1689–1693
44. Zhao Z, Liu H (2007) Spectral feature selection for supervised and unsupervised learning. In: *Machine learning, proceedings of the twenty-fourth international conference*, pp 1151–1157
45. Zhao Y, Zhang S (2005) Generalized dimension-reduction framework for recent-biased time series analysis. *IEEE Trans Knowl Data Eng* 18(2):231–244
46. Zhu X, Zhang S, Jin Z, Zhang Z (2011) Missing value estimation for mixed-attribute data sets. *IEEE Trans Knowl Data Eng* 23(1):110–121
47. Zhu X, Huang Z, Shen HT, Cheng J, Xu C (2012) Dimensionality reduction by mixed kernel canonical correlation analysis. *Pattern Recogn* 45(8):3003–3016
48. Zhu X, Huang Z, Yang Y, HT Shen CXu, Luo J (2013) Self-taught dimensionality reduction on the high-dimensional small-sized data. *Pattern Recogn* 46(1):215–229
49. Zhu X, Zhang L, Huang Z (2014) A sparse embedding and least variance encoding approach to hashing. *IEEE Trans Image Process* 23(9):3737–3750
50. Zhu P, Zuo W, Zhang L, Hu Q, Shiu SCK (2015) Unsupervised feature selection by regularized self-representation. *Pattern Recogn* 48(2):438–446
51. Zhu X, Li X, Zhang S (2016) Block-row sparse multiview multilabel learning for image classification. *IEEE Trans Cybern* 46(2):450
52. Zhu X, Li X, Zhang S, Ju C, Wu X (2017) Robust joint graph sparse coding for unsupervised spectral feature selection. *IEEE Trans Neural Netw Learn Syst* 28(6):1263–1275
53. Zhu X, Li X, Zhang S, Xu Z, Yu L, Wang C (2017) Graph pca hashing for similarity search. *IEEE Trans Multimed*. <https://doi.org/10.1109/TMM.2017.2703636>
54. Zhu X, Suk H-I, Huang H, Shen D (2017) Low-rank graph-regularized structured sparse regression for identifying genetic biomarkers. *IEEE Trans Big Data*. <https://doi.org/10.1109/TBDDATA.2017.2735991>
55. Zhu X, Suk H-I, Wang L, Lee S-W, Shen D (2017) A novel relational regularization feature selection method for joint regression and classification in AD diagnosis. *Meds Image Anal* 38:205–214



Wei Zheng is with the Guangxi Key Lab of Multisource Information Mining & Security, Guangxi Normal University, P.R. China.



Xiaofeng Zhu is with the School of Mathematics and Statistics, Xi'an Jiaotong University, P. R. China, and also with the Guangxi Key Lab of Multisource Information Mining & Security, Guangxi Normal University, P. R. China.



Yonghua Zhu is currently pursuing the M.S. degree in college of Computer Science and Engineering, Guangxi University, Nanning, China.



Rongyao Hu is with the Guangxi Key Lab of Multisource Information Mining & Security, Guangxi Normal University, P.R. China.



Cong Lei is with the Guangxi Key Lab of Multisource Information Mining & Security, Guangxi Normal University, P.R. China.