Pattern Recognition (xxxx) xxxx–xxxx

Contents lists available at ScienceDirect

Pattern Recognition

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

Subspace clustering guided unsupervised feature selection

Pengfei Zhu[a], Wencheng Zhu[a], Qinghua Hu[a,*], Changqing Zhang[a], Wangmeng Zuo[b]

[a] School of Computer Science and Technology, Tianjin University, Tianjin 300350, China
[b] School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China

ARTICLE INFO

Keywords:
Subspace clustering
Unsupervised feature selection
Spectral clustering
Group sparsity

ABSTRACT

Unsupervised feature selection (UFS) aims to reduce the time complexity and storage burden, improve the generalization ability of learning machines by removing the redundant, irrelevant and noisy features. Due to the lack of training labels, most existing UFS methods generate the pseudo labels by spectral clustering, matrix factorization or dictionary learning, and convert UFS to a supervised problem. The learned clustering labels reflect the data distribution with respect to classes and therefore are vital to the UFS performance. In this paper, we proposed a novel subspace clustering guided unsupervised feature selection (SCUFS) method. The clustering labels of the training samples are learned by representation based subspace clustering, and features that can well preserve the cluster labels are selected. SCUFS can well learn the data distribution in that it uncovers the underlying multi-subspace structure of the data and iteratively learns the similarity matrix and clustering labels. Experimental results on benchmark datasets for unsupervised feature selection show that SCUFS outperforms the state-of-the-art UFS methods.

1. Introduction

A tremendous amount of high-dimensional images, texts, and microarray data emerge in the information explosion era. The high-dimensionality of features brings about heavy storage burden, high time complexity and performance degradation of the learning machines [1]. In high-dimensional feature space, the distance concentration phenomenon makes the classical distance based models, e.g., KNN, fail to work [2]. The number of model parameters increases exponentially with the feature dimension and the number of samples for model training is also exponential with the number of features. Moreover, the intrinsic dimensionality of high-dimensional data is typically small [3–7]. Thus, feature selection aims to find a low-dimensional feature subspace while preserving the intrinsic data structure by discovering the noisy, irrelevant and redundant features.

Many feature selection methods have been developed for different priors of data. According to the availability of the label information, feature selection methods can be categorized as unsupervised [8–11], semi-supervised [12] and supervised [13–15] cases. Additionally, different features can be extracted for one sample [16], and one object can be described by different modalities as well [17]. Therefore, multi-view feature selection methods are also proposed [18,19,17]. As the group, overlapped group or tree group relationships may exist in image classification or gene expression data, structured feature selection algorithms are proposed by using the intrinsic relationships among features [20–22]. To select features for ultra-high dimensional data, online streaming feature selection methods are proposed to deal with sequentially added features while the number of samples is fixed [23–25]. For social media data, users are inherently linked and the linked information brings more challenges for feature selection [26].

Among all feature selection methods, unsupervised feature selection is the most challenging due to the lack of label information. Generally, there are three kinds of feature selection methods, i.e., filter, wrapper, and embedding methods. For filter methods, metrics that reflect the data properties are proposed to evaluate the importance of a single feature or a feature subset, e.g., variance, Laplacian score [8] and trace ratio [27]. Wrapper methods select features by the clustering or classification performance of the learning machines [28]. The performance of both filter and wrapper methods is affected by the searching strategies. Embedding methods combine feature selection and model reconstruction together, for example, a feature selection vector or matrix is learned in linear classifier (e.g., support vector machines [29] and least square regression). Compared with filter and wrapper methods, the advantage of embedding methods is that they can take different data properties into account, e.g., manifold structure, data distribution priors, data reconstruction.

The key solution to unsupervised feature selection is how to generate the pseudo labels in unsupervised scenario. Researchers proposed different label generation methods, including spectral embedding [30,31], spectral clustering [32], matrix factorization [9,33],

[*] Corresponding author.
E-mail address: huqinghua@tju.edu.cn (Q. Hu).

http://dx.doi.org/10.1016/j.patcog.2017.01.016
Received 22 June 2016; Received in revised form 4 January 2017; Accepted 9 January 2017

dictionary learning [34], consensus clustering [35], etc. Spectral embedding based methods consider the eigenvectors of the similarity matrix as the pseudo labels, which reflect the data distribution [30,31]. For spectral embedding based methods, label generation and feature selection are two independent stages. Then, spectral clustering is utilized to generate pseudo labels and feature selection is conducted simultaneously [32]. Both spectral embedding and spectral clustering based methods emphasize the preservation of sample similarity. Matrix factorization learns a set of bases and the cluster indicator matrix is used as the pseudo labels [33,9]. Dictionary learning can learn an overcomplete dictionary and the representation coefficients can reflect the data distribution as well. Compared with matrix factorization, there are less constraints and the data can be better reconstructed [34]. Both matrix factorization and dictionary learning underline data reconstruction. Fortunately, the priors of data structure, e.g., manifold structure, can be embedded in feature selection models by the regularization on the cluster indicator or representation coefficient matrix [36]. Consensus clustering is a kind of ensemble clustering method, which aims to fuse several existing partitions into the integrated one [37]. In [35], consensus clustering is integrated with feature selection in that it can obtain robust and clean pseudo labels.

While various techniques are used to generate the pseudo labels, they ignore the multi-subspace structure of the data, i.e., the collection of data from multiple classes or categories lies in a union of low-dimensional subspaces [38]. The spatial proximity of the data that is widely used in standard clustering algorithms, generally does not hold true, when intra-class variations are very large. Subspace clustering algorithms are then proposed to uncover the low-dimensional multi-subspace structure. There are four main categories for the existing subspace clustering methods, i.e., iterative, algebraic, statistical, and spectral clustering-based methods. Among all these methods, sparse and low rank representation based subspace clustering algorithms take advantage of the self-representation property of samples and achieve effective subspace segmentation results [39]. There are two steps, i.e., first learn a similarity graph by sparse or low rank representation and second conduct spectral clustering on this graph. Representation based subspace learning methods avoid choosing the right neighborhood size and dealing with points near the intersection of subspaces.

In this paper, inspired by the success of representation based subspace clustering, we proposed a novel subspace clustering guided unsupervised feature selection (SCUFS) algorithm. Different from the existing methods that generate a local similarity graph by kernel functions, SCUFS learns a global similarity matrix, which can capture the multi-subspace structure of data. Additionally, the similarity matrix and pseudo labels are iteratively updated, which can bring about more accurate pseudo labels. Experiments on six benchmark datasets illustrate that SCUFS outperforms the state-of-the-art unsupervised feature selection methods in terms of both the clustering and classification performance.

The structure of this paper is organized as follows: Section 2 introduces the related work of unsupervised feature selection. Section 3 presents the proposed unsupervised feature selection model. Section 4 conducts experiments and Section 5 concludes.

2. Related work

In this section, we will give a brief review of unsupervised feature selection and subspace clustering.

2.1. Unsupervised feature selection

Let $\mathbf{X} \in \mathbb{R}^{d\times n}$ be a set of training samples where d and n are the number of features and samples, respectively. $\mathcal{F} = \{f_1; \ldots; f_j; \ldots; f_d\}$ denotes the feature matrix, where f_j is the j^{th} feature vector. Unsupervised feature selection aims to select a feature subset from $\mathcal{F}$. Most existing UFS methods can be formulated as:

$$J = G(\mathbf{X}, \theta_1) + T(\mathbf{X}, \theta_2) \tag{1}$$

where $G(\mathbf{X}, \theta_1)$ is the function that generates pseudo labels and $T(\mathbf{X}, \theta_2)$ is the function to conduct feature selection. θ_1 and θ_2 are the model parameters.

Generally, $T(\mathbf{X}, \theta_2)$ is modeled as a loss minimization problem, i.e.,

$$T(\mathbf{X}, \theta_2) = loss(\mathbf{X}, \mathbf{W}) + \lambda R(\mathbf{W}) \tag{2}$$

where $loss(\mathbf{X}, \mathbf{W})$ is the loss function and $\mathbf{W} \in \mathbb{R}^{d\times c}$ is the feature selection matrix. c is the number of pseudo classes. Let $\mathbf{F} \in \mathbb{R}^{n\times c}$ be the pseudo label matrix. Then $loss(\mathbf{X}, \mathbf{W})$ can be formulated as $loss(\mathbf{X}, \mathbf{W}) = \|\mathbf{X}^T\mathbf{W} - \mathbf{F}\|_{2,1}$. Group sparsity regularization is usually imposed on the feature selection matrix $\mathbf{W}$ to remove noisy features.

The performance of a unsupervised feature selection algorithm is mainly up to $G(\mathbf{X}, \theta_1)$. If $G(\mathbf{X}, \theta_1)$ can effectively uncover the true data distribution, the performance of a UFS algorithm can be guaranteed. Researchers have introduced different $G(\mathbf{X}, \theta_1)$ for pseudo label generation. MCFS [30] and MRSF [31] use spectral embedding and consider the flat embedding of high-dimensional data as the pseudo labels. The similarity matrix $\mathbf{S}$ is computed and the eigenvectors of $\mathbf{S}$ reflect the data distribution along the corresponding dimensions. NDFS [32] exploits spectral clustering to learn cluster labels of the input samples. RUFS [33] and EFUS [9] introduce matrix factorization with non-negative orthogonal constraints to unsupervised feature selection. CDLFS [34] relaxes the constraints of matrix factorization and uses the representation coefficients of dictionary learning to represent data distribution.

2.2. subspace clustering

The goal of subspace clustering is to find the multi-cluster structure of data. Among all subspace clustering methods, spectral clustering based methods are quite effective. There are local and global spectral clustering based methods. Local methods rely on the similarity matrix. The disadvantage is that it is hard to deal with the points near the intersection of two subspaces and is sensitive to choose the neighborhood size [40]. The global methods aim to find a better similarity matrix to reflect the sample relationships in multi-subspaces. Sparse and low rank recovery methods assume that one sample can be linearly represented by a dictionary of the data itself [38,39], which is called self-representation. Let $\mathbf{x}_i \in \mathbb{R}^d$ denote the i^{th} sample of $\mathbf{X} \in \mathbb{R}^{d\times n}$. Then $\mathbf{x}_i$ can be represented as

$$\mathbf{x}_i = \mathbf{X}\mathbf{z}_i \quad s.\ t.\ z_{ii} = 0 \tag{3}$$

$z_{ii} = 0$ is required to avoid the trivial solution. For all samples $\mathbf{X}$, we have

$$\mathbf{X} = \mathbf{X}\mathbf{Z} \quad s.\ t.\ \ diag(\mathbf{Z}) = 0 \tag{4}$$

where $\mathbf{Z} \in \mathbb{R}^{n\times n}$ is the representation matrix. Sparse subspace clustering (SSC) or low rank representation (LRR) seeks for a self-representation matrix that can well capture the multi-subspace structure. Then a similarity graph $\mathbf{S} \in \mathbb{R}^{n\times n}$ is constructed as $\mathbf{S} = \frac{|\mathbf{Z}| + |\mathbf{Z}^T|}{2}$. Spectral clustering is conducted on the similarity graph to get the segmentation of the data. Sparse and low rank recovery methods can also learn the multi-subspace structure when there are noise, outliers, corruptions or missing entries in the data.

3. The proposed model

In this section, we present the proposed model, i.e., subspace clustering guided unsupervised feature selection (SCUFS).

3.1. Model

Similar to the framework of the existing unsupervised feature selection (UFS), we also generate the pseudo labels and transform

UFS to a supervised problem. In this paper, we use representation based subspace learning to obtain the data distribution of the training samples.

Given the data matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$, the proposed model is formulated as:

$$\min_{\mathbf{W},\mathbf{F},\mathbf{Z}} \left\{ \begin{array}{l} \|\mathbf{X} - \mathbf{X}\mathbf{Z}\|_F^2 + \|\mathbf{X}^T\mathbf{W} - \mathbf{F}\|_{2,1} \\ + \lambda_1 tr(\mathbf{F}^T\mathbf{L}\mathbf{F}) + \lambda_2 \|\mathbf{W}\|_{2,1} \end{array} \right\} s.\ t.\ \ \mathbf{Z}^T\mathbf{1} = \mathbf{1},\ \ \mathbf{Z}(i,\ i) = 0,$$
$$\mathbf{F}^T\mathbf{F} = \mathbf{I},\ \ \mathbf{F} \geq \mathbf{0} \qquad (5)$$

where $\mathbf{Z} \in \mathbb{R}^{n\times n}$ is the self-representation matrix, $\mathbf{F} \in \mathbb{R}^{n\times c}$ is the cluster indicator matrix, c is the number of clusters, $\mathbf{W} \in \mathbb{R}^{d\times c}$ is the feature selection matrix, and $\mathbf{L} \in \mathbb{R}^{n\times n}$ is the Laplacian matrix, $\mathbf{L} = \mathbf{D} - \mathbf{S}$, $\mathbf{S} = \frac{|\mathbf{Z}| + |\mathbf{Z}^T|}{2}$ and $\mathbf{D}$ is the diagonal matrix, $\mathbf{D}_{ii} = \sum_j \mathbf{S}_{ij}$, λ_1 and λ_2 are the tradeoff parameters.

The model in Eq. (5) is composed of three parts, i.e., self-representation, spectral clustering and feature selection. $\|\mathbf{X} - \mathbf{X}\mathbf{Z}\|_F^2$ is to learn the self-representation matrix $\mathbf{Z}$. There are two constraints, $\mathbf{Z}^T\mathbf{1} = \mathbf{1}$ and $\mathbf{Z}(i,\ i) = 0$. $\mathbf{Z}^T\mathbf{1} = \mathbf{1}$ denotes the data point lies in a union of affine subspaces. $\mathbf{Z}(i,\ i) = 0$ implies that the data point can not be represented by the data point itself. Spectral clustering is to minimize $tr(\mathbf{F}^T\mathbf{L}\mathbf{F})$ with non-negative orthogonal constraints. Features are ranked by $\|\mathbf{w}_i\|_2$ in descending order, where $\mathbf{w}_i$ is the i^{th} row of $\mathbf{W}$.

As shown in Fig. 1, we present the differences between the proposed model and the existing works. Firstly, we learn a similarity graph rather than directly use a similarity matrix computed by the kernel functions, e.g., cosine similarity or heat kernel. The learned similarity graph by self-representation can well reflect the underlying multi-subspace structure of data. However, for the existing models, e.g., RUFS [33], the neighbors may contain the data points from other subspaces, especially when the intra variations are very large. Secondly, we iteratively update the similarity graph $\mathbf{S}$, the pseudo label matrix $\mathbf{F}$ and the feature selection matrix $\mathbf{W}$. As the number of iterations increases, the $\mathbf{S}$ can be more accurate, and therefore we can obtain better $\mathbf{F}$ to model the data distribution.

To show the difference between the learned similarity graph by self-representation and the similarity matrix by kernel functions, in Fig. 2(a), we collect 140 face images of 20 subjects from AR dataset [41]. Fig. 2(b) is the similarity matrix learned by Eq. (5). Fig. 2(c) is the similarity matrix learned by RBF kernel function. From the figures we can see that because of the large within-class variations, the neighbors searched by the kernel similarity do not necessarily come form the same subspace. However, by self-representation, the multi-subspace structure of data can be exactly discovered.

3.2. Optimization and algorithms

In this subsection, we present efficient implementations of the iterative approach to solve the optimization problem in Eq. (5). In the algorithm, we update $\mathbf{Z}$, $\mathbf{F}$ and $\mathbf{W}$ iteratively. In the sequel, we will give a clear description of the optimization method.

Update Z

To update the self-representation matrix $\mathbf{Z}$, we fix $\mathbf{F}$ and $\mathbf{W}$ and ignore irrelevant terms. The optimization problem is rewritten as follows:

$$\min_{\mathbf{Z}} \|\mathbf{X} - \mathbf{X}\mathbf{Z}\|_F^2 + \lambda_1 tr(\mathbf{F}^T\mathbf{L}\mathbf{F}) s.\ t.\ \mathbf{Z}^T\mathbf{1} = \mathbf{1},\ \mathbf{Z}(i,\ i) = 0. \qquad (6)$$

Note that we can eliminate the first equality constraint in Eq. (6) by introducing a Lagrange multiplier α,

$$\min_{\mathbf{Z}} \|\mathbf{X} - \mathbf{X}\mathbf{Z}\|_F^2 + \lambda_1 tr(\mathbf{F}^T\mathbf{L}\mathbf{F}) + \alpha \|\mathbf{1}^T - \mathbf{1}^T\mathbf{Z}\|_F^2 s.\ t.\ \mathbf{Z}(i,\ i) = 0. \qquad (7)$$

According to [16], $\mathbf{X}$ can be replaced with $[\mathbf{X}^T, \alpha * \mathbf{1}]^T$ where α approximates infinity. Thus, the optimization problem in Eq. (7) is equivalent to the following problem:

$$\min_{\mathbf{Z}} \|\mathbf{X} - \mathbf{X}\mathbf{Z}\|_F^2 + \lambda_1 tr(\mathbf{F}^T\mathbf{L}\mathbf{F}) s.\ t.\ \mathbf{Z}(i,\ i) = 0. \qquad (8)$$

As shown in [16], the optimization problem (8) is equivalent to the following problem:

$$\min_{\mathbf{Z}} \|\mathbf{X} - \mathbf{X}\mathbf{Z}\|_F^2 + \frac{\lambda_1}{2} tr(|\mathbf{Z}|^\mathbf{T}\mathbf{P}) s.\ t.\ \mathbf{Z}(i,\ i) = 0. \qquad (9)$$

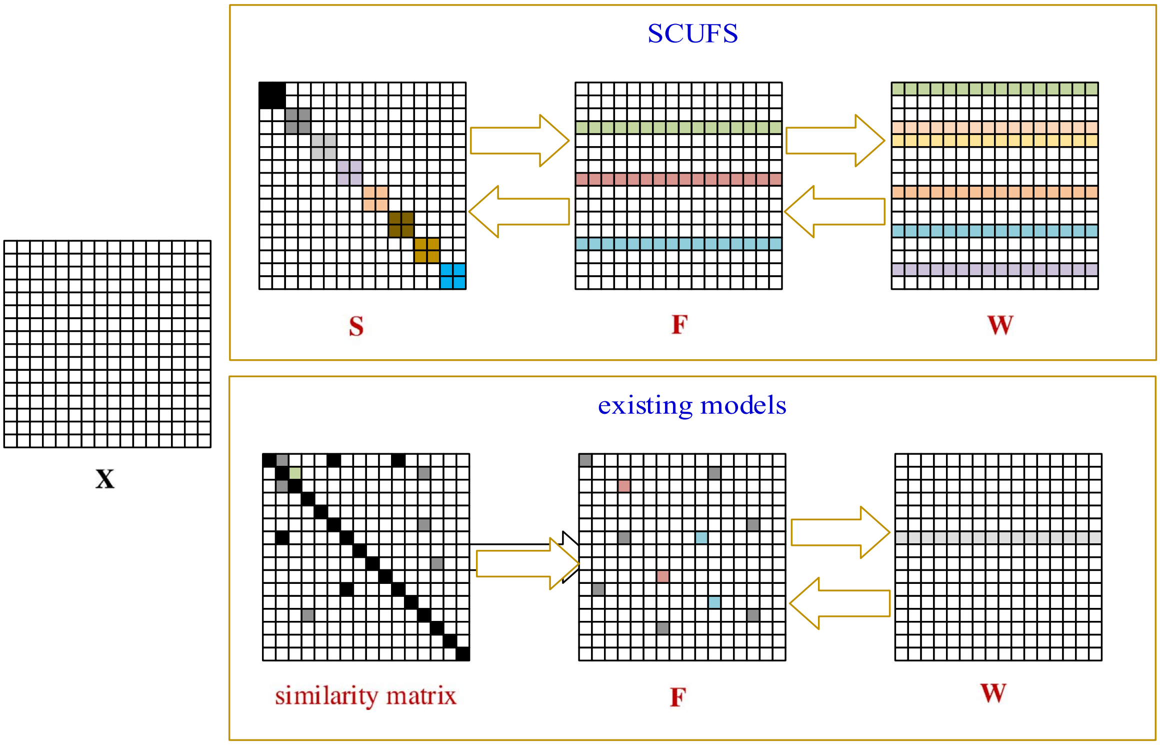


Fig. 1. The differences between SCUFS and the existing models. $\mathbf{X}$ is the data matrix, $\mathbf{S}$ is the similarity matrix, $\mathbf{F}$ is the pseudo label matrix, and $\mathbf{W}$ is the feature selection matrix. The existing models assign the similarity matrix in advance, they only update $\mathbf{F}$ and $\mathbf{W}$ iteratively. However, in our SCUFS, we update $\mathbf{S}$, $\mathbf{F}$ and $\mathbf{W}$ iteratively.

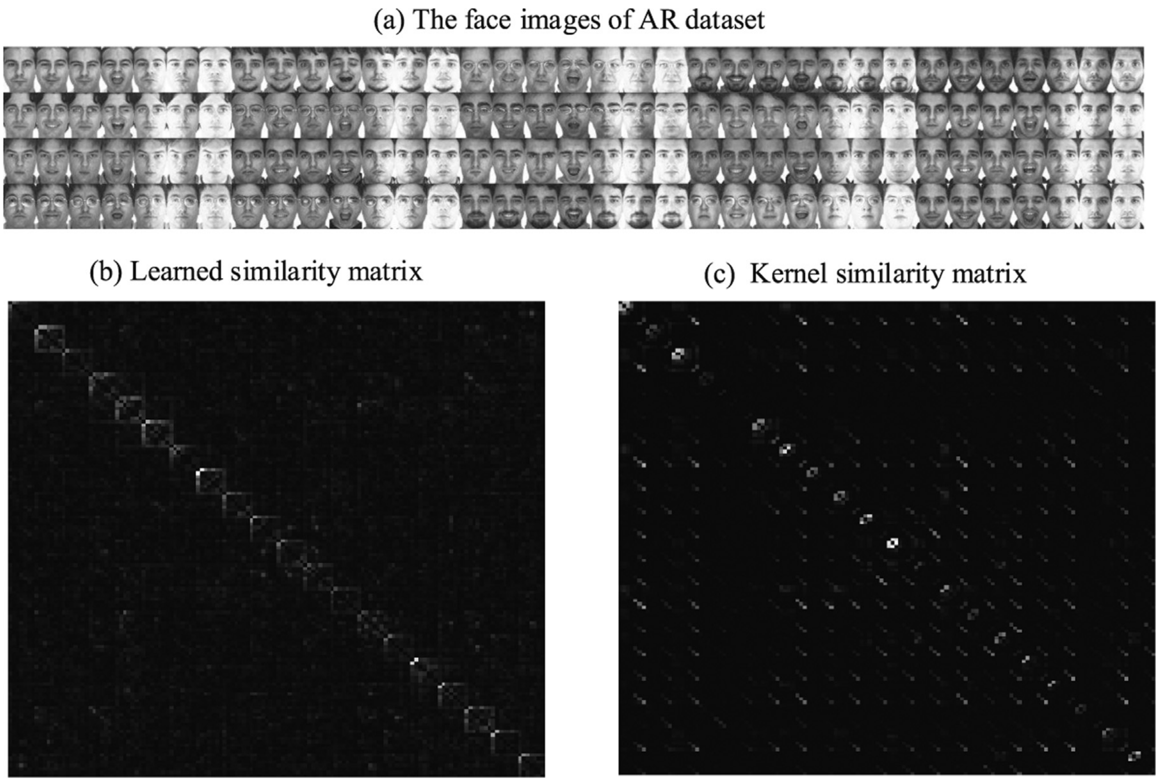


Fig. 2. The comparison between the learned similarity matrix by Eq. (5) and that computed by kernel functions. (a) shows the 140 training images of AR dataset. (b) is the learned similarity by our method. (c) is the kernel similarity matrix used RBF kernel and the parameter is 0.1.

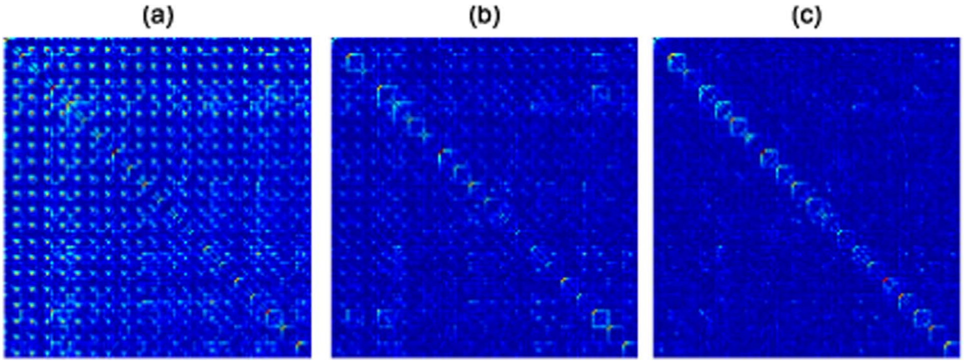


Fig. 3. The learned similarity matrices in 1, 5 and 51 iterations respectively shown in (a), (b) and (c).

Table 1
Summary of the benchmark datasets.

Data	Instances	Features	Classes	Type
warpAR10P	130	2400	10	Image, Face
warpPIE10P	210	2420	10	Image, Face
TOX-171	171	5748	4	Microarray, Bio
Prostate-GE	102	5996	2	Microarray, Bio
ALLAML	72	7192	2	Microarray, Bio
GLI-85	85	22283	2	Microarray, Bio

Table 2
Classification results (accuracy %) of comparison methods.

Data	Laplacian	MCFS	UDFS	SPEC	RUFS	EUFS	SCUFS
warpAR10P	65.78	73.15	72.45	74.67	**85.73**	80.95	81.65
warpPIE10P	86.96	99.12	94.24	86.48	98.99	96.86	**99.37**
TOX-171	54.03	66.04	53.94	54.11	65.17	54.32	**68.89**
Prostate-GE	66.07	76.63	74.28	79.96	75.51	77.52	**82.54**
ALLAML	71.70	75.77	80.80	84.04	81.72	86.89	**92.65**
GLI-85	79.72	77.20	84.39	74.55	84.70	78.09	**86.63**

where $P_{ij} = \|\mathbf{f}^i - \mathbf{f}^j\|_2^2$, $\mathbf{f}^i$ is the i^{th} row of $\mathbf{F}$. Then, the problem (9) can be solved by using alternative optimization strategy. We fix all the rows of $\mathbf{Z}$ except i^{th} row and we solve i^{th} row of $\mathbf{Z}$:

$$\min_{\mathbf{z}} \ \|\mathbf{X}_1 - \mathbf{x}\mathbf{z}^T\|_F^2 + \frac{\lambda_1}{2}|\mathbf{z}|^T\mathbf{p}s.\ t. \quad z_i = 0 \tag{10}$$

where $\mathbf{z}^T$ is the i^{th} row of $\mathbf{Z}$, $\mathbf{p}$ is the i^{th} column of $\mathbf{P}$, $\mathbf{X}_1 = \mathbf{X} - (\mathbf{X}\mathbf{Z} - \mathbf{x}\mathbf{z}^T)$ and z_i is the i^{th} element of $\mathbf{z}$. The objective in (10) can be equivalently transformed to the following problem:

$$\min_{\mathbf{z}} \ \|\mathbf{z} - \mathbf{v}\|_2^2 + \frac{\lambda_1}{2}|\mathbf{z}|^T\mathbf{p}s.\ t. \quad z_i = 0 \tag{11}$$

where $\mathbf{v} = \frac{\mathbf{X}_1^T\mathbf{z}}{\mathbf{z}^T\mathbf{z}}$. As shown in ([16]), the solution of the minimization problem in Eq. (11) is as follows: if k=i, z_k=0 and if $k \neq i$,

Table 3
Clustering results (NMI %) of comparison methods.

Data	Laplacian	MCFS	UDFS	SPEC	RUFS	EUFS	SCUFS
warpAR10P	19.76	18.57	44.14	47.95	47.30	53.96	**55.26**
warpPIE10P	20.60	54.64	32.22	39.54	48.87	**66.23**	66.20
TOX-171	11.86	12.46	10.17	9.83	27.25	17.54	**30.54**
Prostate-GE	2.22	2.03	6.32	1.96	5.58	5.09	**7.64**
ALLAML	10.84	11.33	2.48	20.32	15.47	11.36	**45.64**
GLI-85	14.19	18.34	12.18	9.20	24.09	12.14	**24.65**

Table 4
Clustering results (accuracy %) of comparison methods.

Data	Laplacian	MCFS	UDFS	SPEC	RUFS	EUFS	SCUFS
warpAR10P	21.03	22.17	39.49	45.65	43.49	51.23	**52.25**
warpPIE10P	20.57	42.02	30.91	36.18	40.77	**57.71**	57.55
TOX-171	40.49	40.91	38.98	38.83	49.29	44.03	**51.92**
Prostate-GE	58.06	58.11	63.50	57.91	60.78	60.95	**66.48**
ALLAML	69.02	68.24	55.42	75.26	73.61	68.45	**83.92**
GLI-85	65.87	65.42	69.13	60.20	73.75	64.58	**74.27**

$$z_k = sign(v_k)(|v_k| - \frac{\lambda_1 p_k}{4})_+ = \begin{cases} v_k - \frac{\lambda_1 p_k}{4}, & if\ v_k > \frac{\lambda_1 p_k}{4} \\ v_k + \frac{\lambda_1 p_k}{4}, & if\ v_k < -\frac{\lambda_1 p_k}{4} \\ 0, & otherwise \end{cases} \tag{12}$$

where z_k, v_k and p_k denote the k^{th} element of **z**, **v** and **p**, respectively.

Update F

To update the cluster indicator matrix **F**, we fix **Z** and **W**, moreover, omit unrelated items and consider the following optimization problem:

$$\min_{\mathbf{F}} tr(\mathbf{F}^T\mathbf{L}\mathbf{F}) + \frac{1}{\lambda_1}\|\mathbf{X}^T\mathbf{W} - \mathbf{F}\|_{2,1} s.\ t.\ \ \mathbf{F}^T\mathbf{F} = \mathbf{I}, \mathbf{F} \geq \mathbf{0} \tag{13}$$

To eliminate the orthogonal constraint, we add a penalty term $\gamma\|\mathbf{F}^T\mathbf{F} - \mathbf{I}\|_F^2$ (we set $\gamma = 10^6$ in our experiment). Thus, we have the following optimization problem:

$$\min_{\mathbf{F}} \left\{ \begin{array}{l} tr(\mathbf{F}^T\mathbf{L}\mathbf{F}) + \frac{1}{\lambda_1}\|\mathbf{X}^T\mathbf{W} - \mathbf{F}\|_{2,1} \\ + \gamma\|\mathbf{F}^T\mathbf{F} - \mathbf{I}\|_F^2 \end{array} \right\} s.\ t.\ \mathbf{F} \geq \mathbf{0} \tag{14}$$

We introduce Lagrange multipliers **Φ** to remove inequality constraint and obtain the Lagrange function:

$$\varphi(\mathbf{F}, \mathbf{\Phi}) = \left\{ \begin{array}{l} tr(\mathbf{F}^T\mathbf{L}\mathbf{F}) + \frac{1}{\lambda_1}\|\mathbf{X}^T\mathbf{W} - \mathbf{F}\|_{2,1} \\ + \gamma\|\mathbf{F}^T\mathbf{F} - \mathbf{I}\|_F^2 - tr(\mathbf{\Phi}^T\mathbf{F}) \end{array} \right\} \tag{15}$$

We take the derivative of Eq. (15) with **F** to zero $\frac{\partial\varphi(\mathbf{F}, \Phi)}{\partial\mathbf{F}} = 0$ and we have:

$$\frac{\partial\varphi(\mathbf{F}, \mathbf{\Phi})}{\partial\mathbf{F}} = \left\{ \begin{array}{l} 2\frac{1}{\lambda_1}\mathbf{Q}(\mathbf{X}^T\mathbf{W} - \mathbf{F}) + 2\mathbf{L}\mathbf{F} \\ + 4\gamma\mathbf{F}(\mathbf{F}^T\mathbf{F} - \mathbf{I}) - \mathbf{\Phi} \end{array} \right\} = 0 \tag{16}$$

Then, we get **Φ**:

$$\mathbf{\Phi} = 2\frac{1}{\lambda_1}\mathbf{Q}(\mathbf{X}^T\mathbf{W} - \mathbf{F}) + 2\mathbf{L}\mathbf{F} + 4\gamma\mathbf{F}(\mathbf{F}^T\mathbf{F} - \mathbf{I}) \tag{17}$$

where **Q** is a diagonal matrix and the i^{th} element of **Q** is $\mathbf{Q}_{ii} = \frac{1}{2\|(\mathbf{X}^T\mathbf{W} - \mathbf{F})_{i:}\|_2}$. According to [18,42], the Karush-Kuhn-Tucker condition $\mathbf{\Phi}_{ij}\mathbf{F}_{ij} = 0$ is applied. Thus, we get the following equation:

$$\left\{ \begin{array}{l} 2\frac{1}{\lambda_1}\mathbf{Q}(\mathbf{X}^T\mathbf{W} - \mathbf{F}) \\ + 2\mathbf{L}\mathbf{F} + 4\gamma\mathbf{F}(\mathbf{F}^T\mathbf{F} - \mathbf{I}) \end{array} \right\}_{ij} \mathbf{F}_{ij} = 0 \tag{18}$$

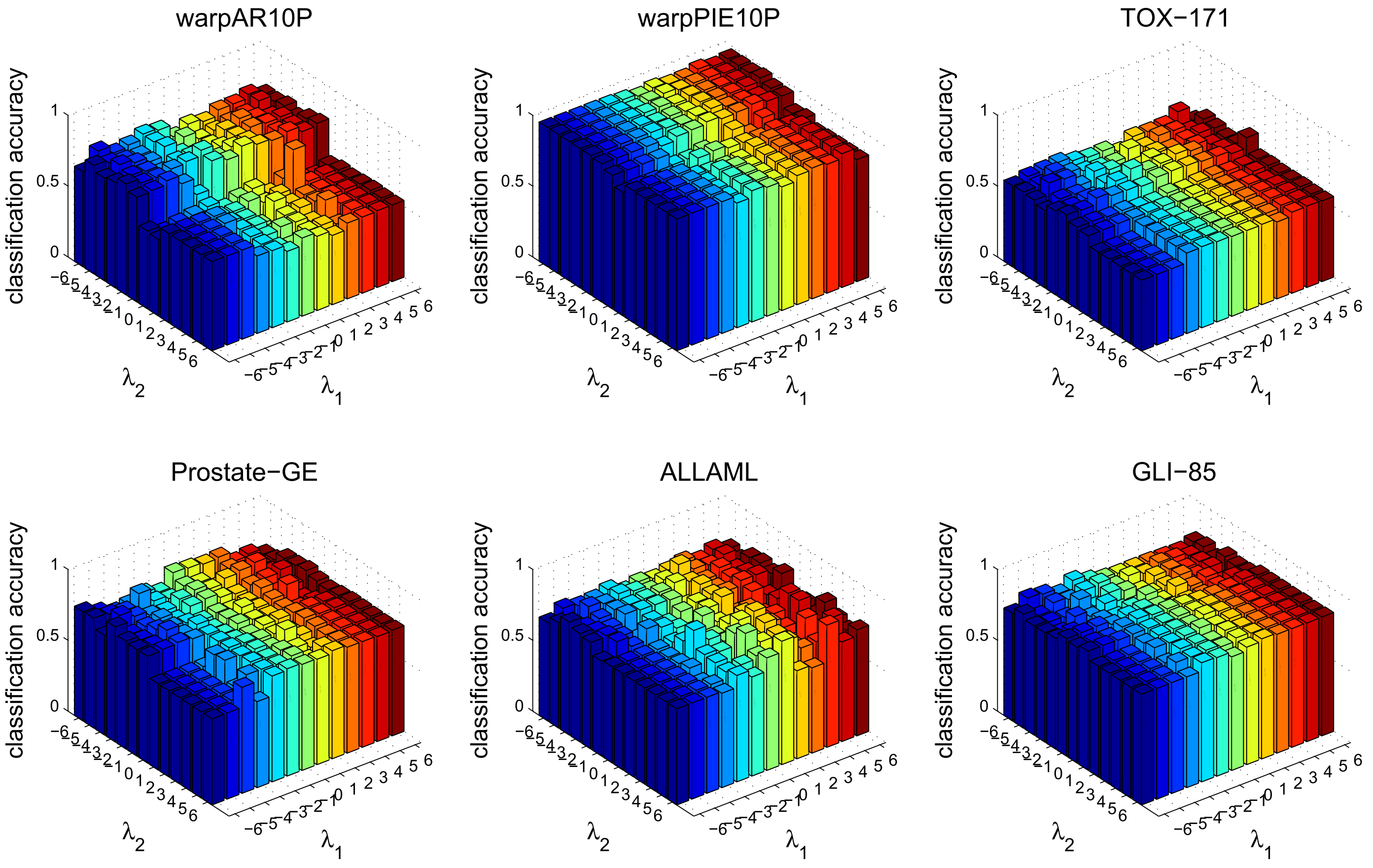


Fig. 4. Classification accuracy on six data sets with different parameters for SCUFS.

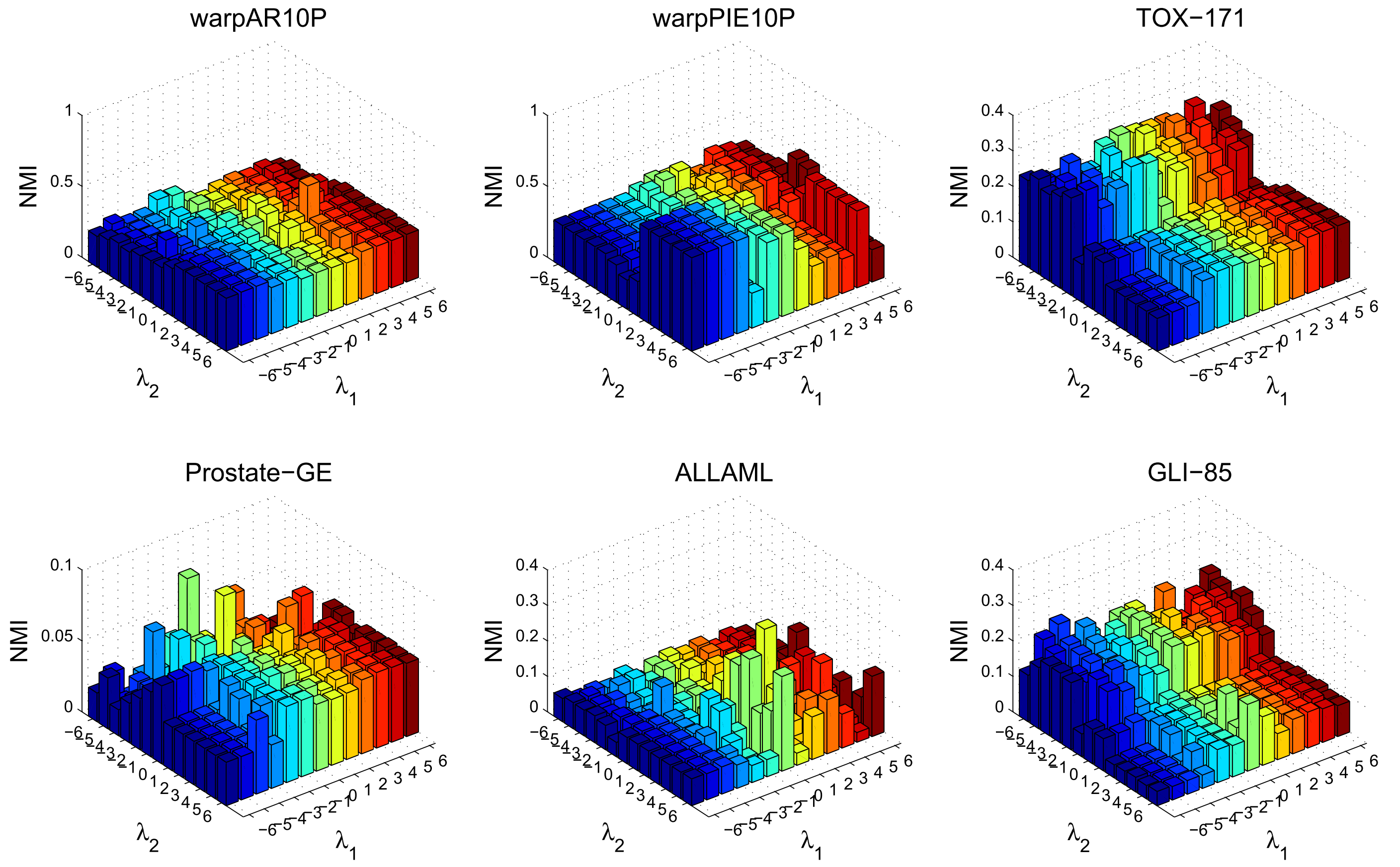


Fig. 5. Clustering NMI on six data sets with different parameters for SCUFS.

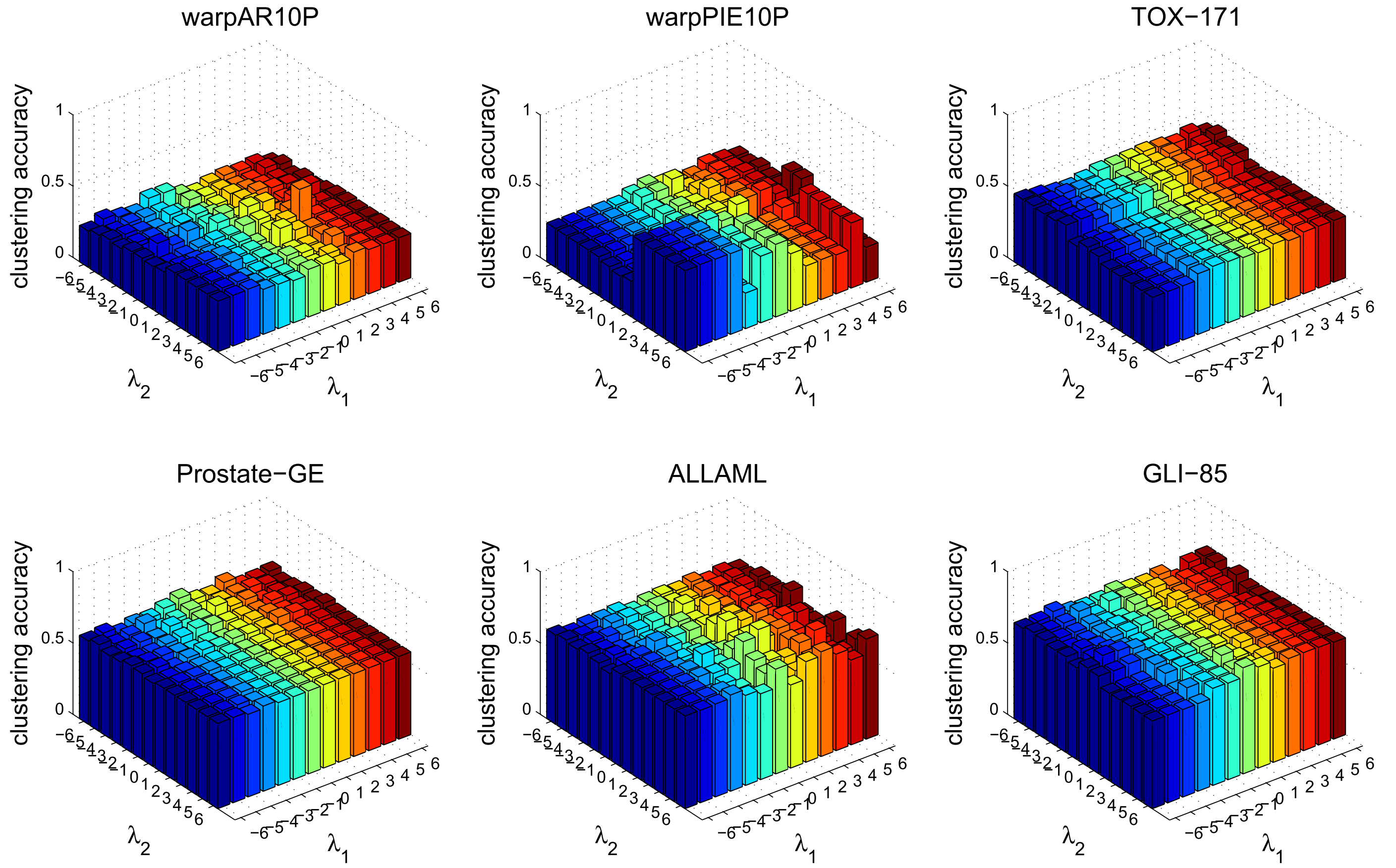
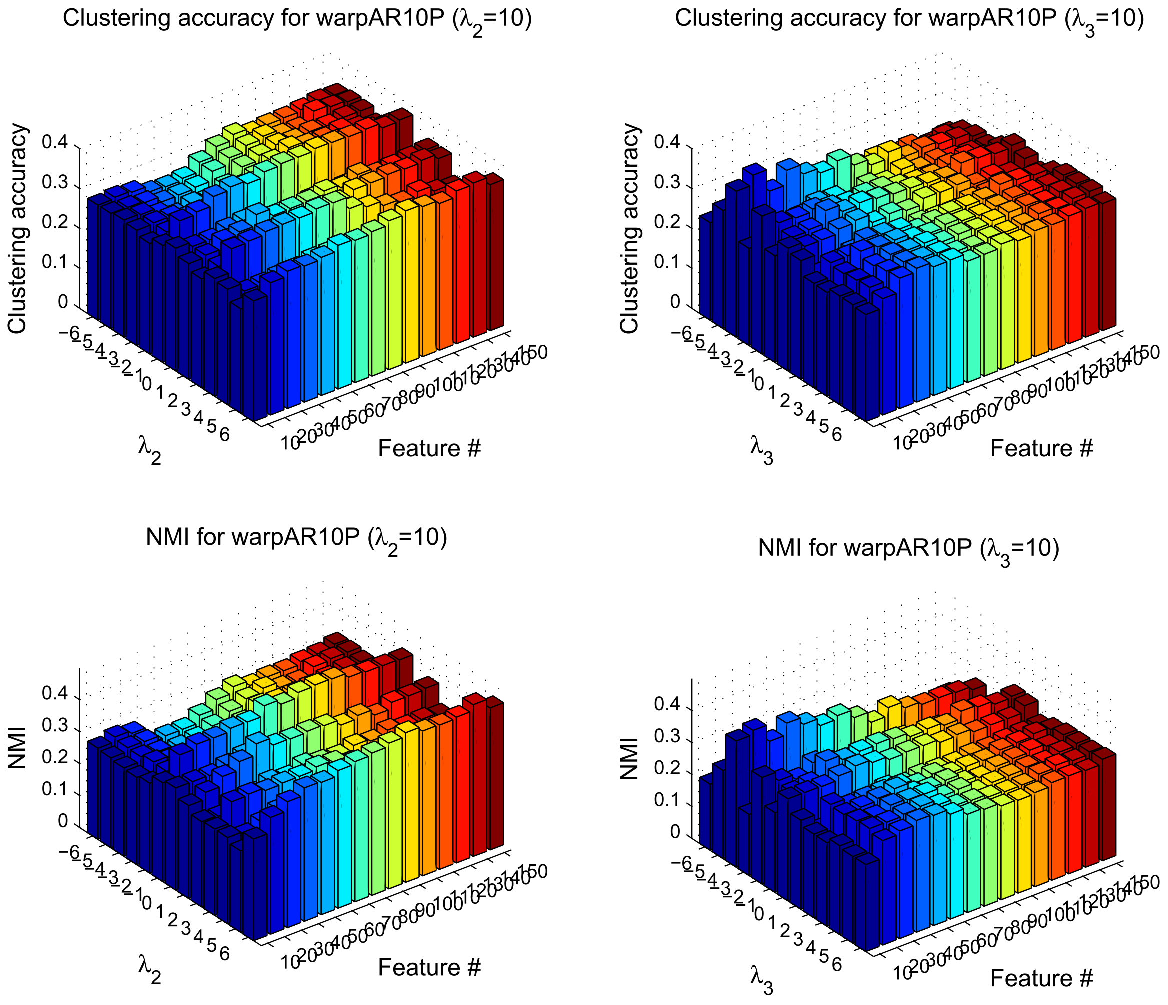


Fig. 6. Clustering accuracy on six data sets with different parameters for SCUFS.


Clustering accuracy for warpAR10P (λ_2=10)

Clustering accuracy for warpAR10P (λ_3=10)

NMI for warpAR10P (λ_2=10)

NMI for warpAR10P (λ_3=10)

Fig. 7. Clustering results on warpAR10P with different parameters for SCUFS.

We can use the following updating method:

$$\mathbf{F}_{ij} = \frac{(\frac{1}{\lambda_1}\mathbf{QF} + 2\gamma\mathbf{F})_{ij}}{(\mathbf{LF} + \frac{1}{\lambda_1}\mathbf{QX}^T\mathbf{W} + 2\gamma\mathbf{FF}^T\mathbf{F})_{ij}}\mathbf{F}_{ij} \tag{19}$$

After updating $\mathbf{F}$, we need normalize $\mathbf{F}$ to meet the constraint $\mathbf{F}^T\mathbf{F} = \mathbf{I}$.

Update W

To update feature selection matrix $\mathbf{W}$, we fix $\mathbf{Z}$ and $\mathbf{F}$, and omit irrelevant items. We get the following optimization problem:

$$\min_{\mathbf{W}} \frac{1}{\lambda_2}\|\mathbf{X}^T\mathbf{W} - \mathbf{F}\|_{2,1} + \|\mathbf{W}\|_{2,1} \tag{20}$$

Note that the equation in (20) is equivalent to the following optimization problem:

$$\min_{\mathbf{W}} \left\{ \begin{array}{l} \frac{1}{\lambda_2}Tr((\mathbf{X}^T\mathbf{W} - \mathbf{F})^T\mathbf{G}(\mathbf{X}^T\mathbf{W} - \mathbf{F})) \\ + Tr(\mathbf{W}^T\mathbf{HW}) \end{array} \right\} \tag{21}$$

where $\mathbf{G}$ and $\mathbf{H}$ are diagonal matrices, their i^{th} diagonal elements are $G_{ii} = \frac{1}{2\|(\mathbf{X}^T\mathbf{W} - \mathbf{F})_{i:}\|_2}$ and $H_{ii} = \frac{1}{2\|\mathbf{W}_{i:}\|_2}$.

We set the derivative of Eq. (21) with respect to $\mathbf{W}$ to zero. Then, we have:

$$\left\{ \begin{array}{l} \frac{1}{\lambda_2}\frac{\partial Tr((\mathbf{X}^T\mathbf{W} - \mathbf{F})^T\mathbf{G}(\mathbf{X}^T\mathbf{W} - \mathbf{F}))}{\partial\mathbf{W}} \\ + \frac{\partial Tr(\mathbf{W}^T\mathbf{HW})}{\partial\mathbf{W}} \end{array} \right\} = 0 \tag{22}$$

After solving the equation in (22), we get:

$$\mathbf{W} = (\mathbf{XGX}^T + \lambda_2\mathbf{H})^{-1}(\mathbf{XGF}) \tag{23}$$

We can iteratively update $\mathbf{W}$, $\mathbf{G}$ and $\mathbf{H}$ until the objective function converges.

In our algorithm, We update $\mathbf{Z}$, $\mathbf{F}$ and $\mathbf{W}$ by solving above three subproblems iteratively until the objective function converges. As for the stopping criterion, we utilize the loss function in Eq. (5). When the loss variation ratio is below 10^{-6}, we break the loop. Empirically, we set the maximum iteration number as 100. We summarize our algorithm in Algorithm 1.

Algorithm 1. Subspace Clustering guided Unsupervised Feature Selection.

```
Input:
The data matrix $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$;
The parameters $\lambda_1$ and $\lambda_2$;
Output:
The feature selection matrix $\mathbf{W} \in \mathbb{R}^{d\times c}$;
Initialize W,F and Z.
while
    Update Z by Eq. (12);
```

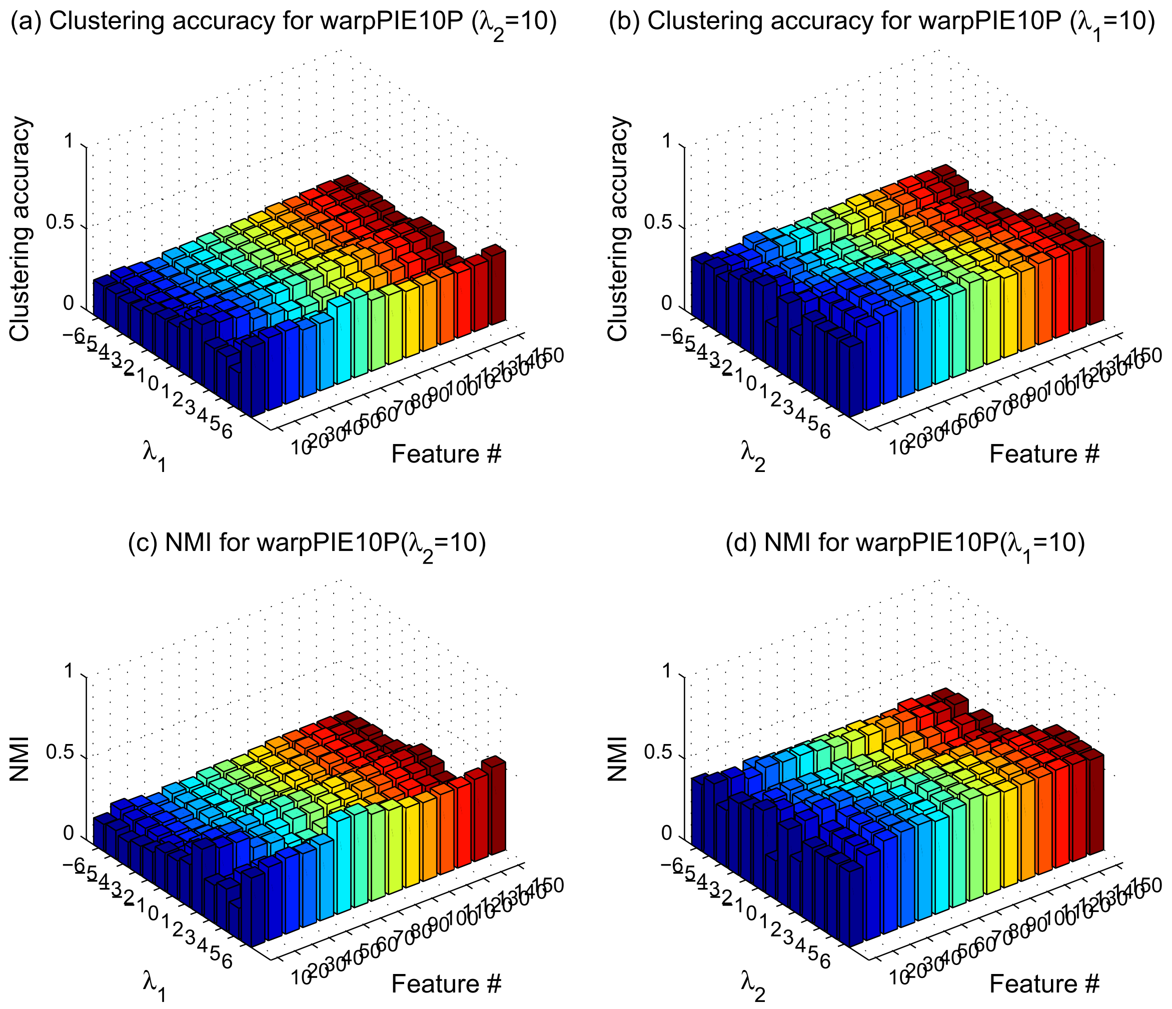


Fig. 8. Clustering results on six data sets with different parameters for SCUFS.

```
        Update F by Eq. (19);
        Update W by Eq. (23);
        until It converges.
```

3.3. Time complexity

In this section, we analyze the time complexity of our optimization problem. There are three subproblems in our optimization method: subproblem **Z**, subproblem **F** and subproblem **W**. In subproblem **Z**, matrix multiplication is needed for each row of **Z** with the time complexity of $O(n^2d)$. Therefore, the total time complexity of subproblem **Z** is $O(n^3d)$. In subproblem **F**, each iteration to update **F** costs $O(cn^2)$, c is the number of pseudo classes $(c < < , d)$. We use the efficient algorithm to solve the subproblem **W** [13]. When $n > d$, the time complexity of subproblem **W** is $O(nd^2 + d^3 + ndc + d^2c)$. When $d > n$, the time complexity of subproblem **W** is $O(d^3 + n^2d + ndc + n^2c)$. The time complexity of subproblem **W** is $O(max(n, d) \times d^2)$. Thus, the total time complexity of SCUFS in each iteration is as follows: if $n > d$, the time complexity for solving Eq. (5) is $O(n^3d)$ and if $d > n$, the time complexity for solving Eq. (5) is $O(n^3d + d^3)$.

3.4. Discussions

In this part, we visually show the variations of the learned similarity matrix $\mathbf{S} = \frac{|\mathbf{Z}| + |\mathbf{Z}^T|}{2}$ in different iterations. The learned similarity matrices in 1, 5 and 51 iterations are shown in Fig. 3(a), (b) and (c), respectively. We can see that a better **S** is learned with the number of iterations increasing. A better similarity graph matrix can induce a better pseudo label matrix **F** in Eq. (5), and therefore can lead to a better unsupervised feature selection result.

4. Experiments

In this section, we conduct experiments on several benchmark datasets to evaluate the performance of our algorithm, and we compare SCUFS with state-of-the-art algorithms.

4.1. Datasets

In our experiment, we use six benchmark datasets, including two face image datasets (warpAR10P and warpPIE10P) and four microarray datasets (TOX-171, Prostate-GE and ALLAML, GLI-85). In Table 1, we summarize the detailed information of these six benchmark datasets.

4.2. Comparison methods

We compare SCUFS algorithm with six state-of-the-art unsupervised feature selection algorithms: Laplacian Score [8], MCFS [30], UDFS [43], SPEC [44], RUFS [33] and EUFS [9]. We obtain the codes of these comparison methods from the original authors and the following is the summary of the comparison methods:

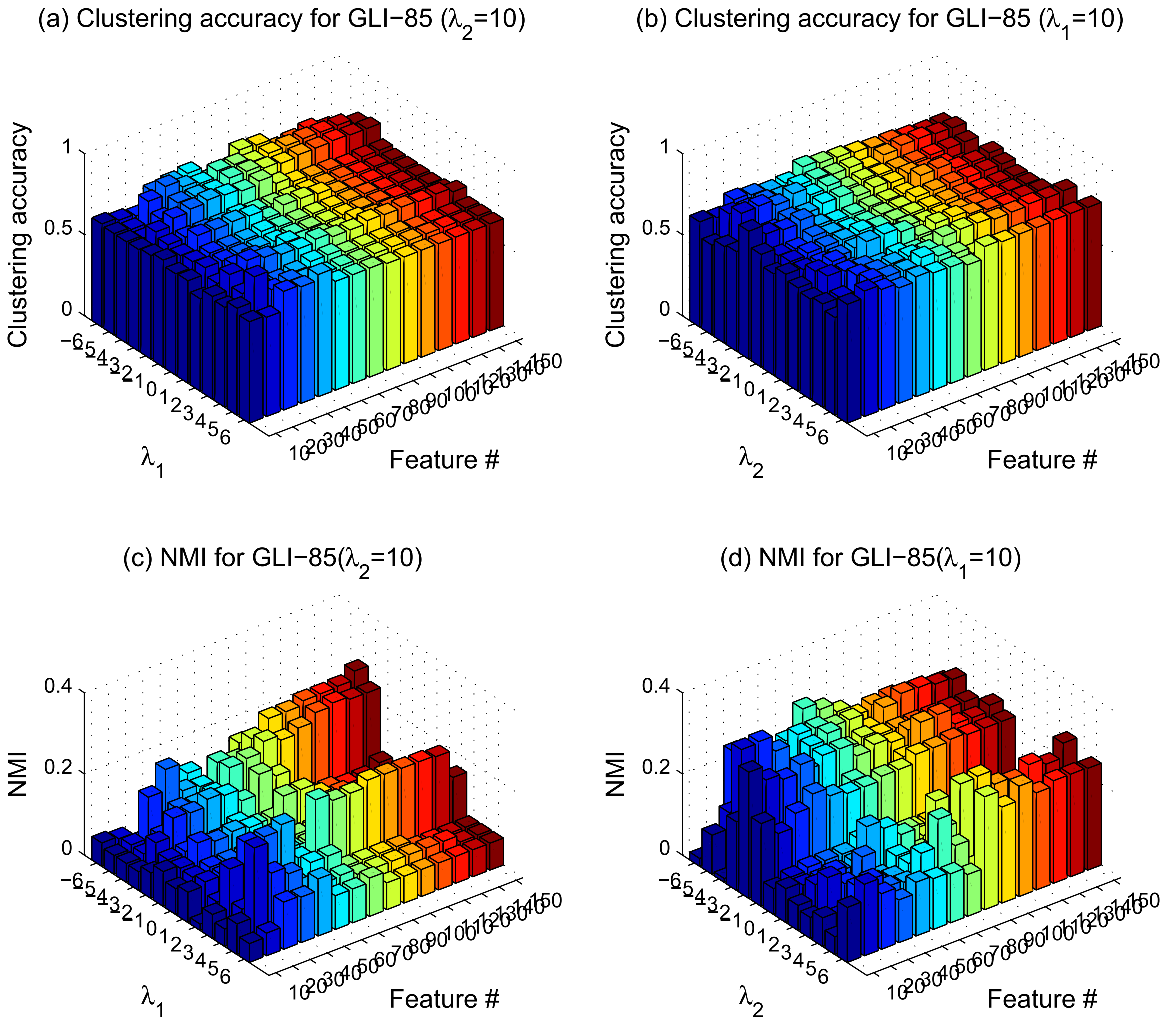


Fig. 9. Clustering results on six data sets with different parameters for SCUFS.

1. Laplacian Score [8]: Laplacian Score is a filter method and evaluates features through its power of preserving local manifold structure.
2. MCFS [30]: multi-cluster unsupervised feature selection. MCFS is a filter method, which detects data distribution by spectral clustering techniques and selects features using sparse regression.
3. UDFS [43]: unsupervised discriminant feature selection. UDFS selects discriminative features by exploiting discriminative information and group sparsity.
4. SPEC [44]: spectral feature selection. SPEC is also a filter method and uses spectral analysis to select features. Unlike other feature selection methods, SPEC is an algorithm unifying supervised and unsupervised feature selection.
5. RUFS [33]: robust unsupervised feature selection. RUFS performs nonnegative matrix factorization to do robust label learning and robust feature selection with group sparsity in the meantime.
6. EUFS [9]: embedded unsupervised feature selection. Like RUFS, EUFS applies matrix factorization to obtain the cluster bases and the pseudo class labels and selects features with cluster bases.

4.3. Parameter setting

Following the previous works ([9,43,33]), we use classification accuracy, Normalized Mutual Information(NMI) and clustering accuracy to evaluate our method. The range of parameters is from $\{10^{-6}, 10^{-5}, \ldots, 10^{5}, 10^{6}\}$ and we record the best classification and clustering results. We set the number of selected features as $\{10, 20, 30, \ldots, 150\}$ and take average results of different dimensions as the final results for all methods. As for Laplacian Score, MCFS, UDFS, RUFS and EUFS, the neighborhood size is fixed to be 5. We use nearest neighbor classifier to evaluate classification performance and K-means to perform clustering algorithm. As K-means algorithm depends on initialization, we repeat clustering algorithm 20 times with random initializations, and the average results are adopted.

4.4. Feature selection results

We present the classification accuracy, NMI and clustering accuracy of different feature selection methods in Table 2–4. The experimental results illustrate that SCUFS performs well in classification accuracy, NMI and clustering accuracy and also demonstrate the effectiveness of our algorithm. There are two reasons why our method achieves superior performances: (1) our method utilizes subspace clustering to mine underlying cluster structure of data; (2) the similarity between data is not specified in advance, we learn the similarity matrix by iteratively updating **F**, **Z** and **W**. Thus, it reflects real similarity between the data properly.

4.5. Parameters sensitivity

In Figs. 4–6, we show the experimental results on different λ_1 and λ_2. The logarithms (base 10) of parameters are taken.

The results of classification accuracy are shown in Fig. 4, and we can see that our algorithm is not sensitive to λ_1 with the same λ_2. As for image data, warpAR10P and warpPIE10P, the classification accuracy of

$\lambda_2 < 1$ is higher than $\lambda_2 > 1$. Thus, we know that the information of cluster structure is important for image data. It also can be seen that the biological data are not so sensitive to λ_1 and λ_2. However, our method has higher classification accuracy with $\lambda_2 = 10^{-1}$ for TOX-171 and the classification accuracy changes rapidly with different parameters for Prostate-GE and ALLAML. As for GLI-85, the data is not sensitive to λ_1 and λ_2.

The results of NMI are presented in Fig. 5. The experimental results are sensitive to parameters using this evaluation criterion. For warpAR10P, the results suddenly increase to a peak when $\lambda_1 = 10^3$ and $\lambda_2 = 10$ but are stable with other parameters. As for warpPIE10P, our method has high NMI with $\lambda_1 < 10^{-2}$ and $\lambda_2 > 10$ and the NMI changes slightly when $10^{-6} < = \lambda_1 < = 10^{-2}$ and $10^{-6} < = \lambda_2 < = 10^{-2}$. Our method has good performance with $\lambda_1 < 10^{-1}$ for TOX-171 and is insensitive when $10^{-3} < = \lambda_1 < = 10^6$ and $10 < = \lambda_2 < = 10^6$. For Prostate-GE, ALLAML and GLI-85, the results change frequently. However, the results of Prostate-GE are not sensitive to parameters when $10^{-2} < = \lambda_1 < = 10^6$ and $10^3 < = \lambda_2 < = 10^6$. ALLAML has the same NMI when $10^{-6} < = \lambda_1 < = 10^{-4}$ and $1 < = \lambda_2 < = 10^6$. The NMI of GLI-85 changes slightly when $10^3 < = \lambda_1 < = 10^5$ and $10^2 < = \lambda_2 < = 10^6$.

From Fig. 6, we can see that the results of clustering accuracy are not sensitive to parameters on six datasets. For warpAR10P, the results have a peak when $\lambda_1 = 10^3$ and $\lambda_2 = 10$. For warpPIE10P, our method has high clustering accuracy with $\lambda_1 < 10^{-2}$ and $\lambda_2 > 10$. The results of clustering accuracy are relatively smooth comparing with the results of clustering NMI for the biological data.

The sensitivity of the feature dimension is a very challenging and unsolved problem in feature selection. We analyze the sensitiveness of λ_1, λ_2 and the number of selected features in Fig. 7, 8 and 9. The logarithms (base 10) of parameters are taken. The results show that our method is sensitive to the number of selected features. Fig. 9 shows the clustering accuracy and clustering NMI drop as the number of selected feature increases.

5. Conclusions and future work

In this paper, we proposed a subspace clustering guided unsupervised feature selection (SCUFS) method. Compared with the existing UFS methods that conduct spectral clustering on the kernel similarity matrix, we learn a similarity graph matrix by self-representation of samples. The self-representation can well uncover the multi-subspace structure of data, and therefore we can learn a more accurate data distribution in unsupervised settings. We proposed a joint framework that considers self-representation of samples, spectral clustering and feature selection simultaneously. Experiments on benchmark datasets show that the performance of the proposed method is superior to the state-of-the-art unsupervised feature selection methods.

In the future work, we will extend the proposed model to deal with more complex data, e.g., data with noises, corruptions or missing entries. Sparse and low rank representation will be introduced to SCUFS to improve the robustness of the proposed model.

Conflict of interest

None declared.

Acknowledgements

This work was supported by the National Program on Key Basic Research Project under Grant 2013CB329304, the National Natural Science Foundation of China under Grants 61502332, 61432011, 61222210.

References

[1] Y. Zhai, Y.S. Ong, I.W. Tsang, The emerging big dimensionality, Comput. Intell. Mag. IEEE 9 (2014) 14–26.

[2] V.D. Mil'Man, New proof of the theorem of a. dvoretzky on intersections of convex bodies, Funct. Anal. Appl. 5 (1971) 288–295.

[3] X. Lu, X. Li, Multiresolution imaging, IEEE Trans. Cybern. 44 (2014) 149–160.

[4] X. Lu, Y. Wang, Y. Yuan, Sparse coding from a bayesian perspective, IEEE Trans. Neural Netw. Learn. Syst. 24 (2013) 929–939.

[5] L. Shao, L. Liu, X. Li, Feature learning for image classification via multiobjective genetic programming, IEEE Trans. Neural Netw. Learn. Syst. 25 (2014) 1359–1371.

[6] X. Zhu, S. Zhang, Z. Jin, Z. Zhang, Z. Xu, Missing value estimation for mixed-attribute data sets, IEEE Trans. Knowl. Data Eng. 23 (2011) 110–121.

[7] Y. Yang, Z. Ma, F. Nie, X. Chang, A.G. Hauptmann, Multi-class active learning by uncertainty sampling with diversity maximization, Int. J. Comput. Vis. 113 (2015) 113–127.

[8] X. He, D. Cai, P. Niyogi, Laplacian score for feature selection, in: Advances in neural information processing systems, pp. 507–514.

[9] S. Wang, J. Tang, H. Liu, Embedded unsupervised feature selection, in: Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, pp. 470–476.

[10] L. Du, Y.-D. Shen, Unsupervised feature selection with adaptive structure learning, in: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '15, ACM, New York, NY, USA, 2015, pp. 209–218.

[11] X. Zhu, X. Li, S. Zhang, C. Ju, X. Wu, Robust joint graph sparse coding for unsupervised spectral feature selection, IEEE Trans. Neural Netw. Learn. Syst. (2016) 1–13.

[12] X. Chang, F. Nie, Y. Yang, H. Huang, A convex formulation for semisupervised multi-label feature selection, in: Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence.

[13] F. Nie, H. Huang, X. Cai, C. H. Ding, Efficient and robust feature selection via joint $l_{2,1}$-norms minimization, in: Advances in neural information processing systems, pp. 1813–1821.

[14] R. He, T. Tan, L. Wang, W.-S. Zheng, $l_{2,1}$ regularized correntropy for robust feature selection, in: Computer Vision and Pattern Recognition (CVPR), pp. 2504–2511.

[15] Y. Han, Y. Yang, X. Zhou, Co-regularized ensemble for feature selection, in: Proceedings of the 23rd International Joint Conference on Artificial Intelligence, IJCAI 2013, pp. 1380–1386.

[16] H. Gao, F. Nie, X. Li, H. Huang, Multi-view subspace clustering, in: Proceedings of the IEEE International Conference on Computer Vision, pp. 4238–4246.

[17] M. Qian, C. Zhai, Unsupervised feature selection for multi-view clustering on text-image web news data, in: Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, pp. 1963–1966.

[18] Y. Feng, J. Xiao, Y. Zhuang, X. Liu, Adaptive unsupervised multi-view feature selection for visual concept recognition, in: Computer Vision-ACCV 2012, Springer, 2012, pp. 343–357.

[19] J. Tang, X. Hu, H. Gao, H. Liu, Unsupervised feature selection for multi-view data in social media., in: SDM, SIAM, pp. 270–278.

[20] Y. Lei, L. Jun, Y. Jieping, Efficient methods for overlapping group lasso, IEEE Trans. Pattern Anal. Mach. Intell. 35 (2013) 2104–2116.

[21] J. Wang, J. Ye, Multi-layer feature reduction for tree structured group lasso via hierarchical projection, in: Advances in Neural Information Processing Systems, pp. 1279–1287.

[22] T. Gao, Z. Wang, Q. Ji, Structured feature selection, in: Proceedings of the IEEE International Conference on Computer Vision, pp. 4256–4264.

[23] X. Wu, K. Yu, W. Ding, H. Wang, X. Zhu, Online feature selection with streaming features, IEEE Trans. Pattern Anal. Mach. Intell. 35 (2013) 1178–1192.

[24] J. Wang, M. Wang, P. Li, L. Liu, Z. Zhao, X. Hu, X. Wu, Online feature selection with group structure analysis, IEEE Trans. Knowl. Data Eng. 27 (2015) 3029–3041.

[25] K. Yu, W. Ding, X. Wu, Lofs: Library of online streaming feature selection, arXiv:1603.00531(2016)

[26] J. Tang, H. Liu, Unsupervised feature selection for linked social media data, in: Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 904–912.

[27] F. Nie, S. Xiang, Y. Jia, C. Zhang, S. Yan, Trace ratio criterion for feature selection., in: Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence.

[28] R. Kohavi, G.H. John, Wrappers for feature subset selection, Artif. Intell. 97 (1997) 273–324.

[29] J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio, V. Vapnik, Feature selection for svms, in: NIPS, volume 12, Citeseer, pp. 668–674.

[30] D. Cai, C. Zhang, X. He, Unsupervised feature selection for multi-cluster data, in: Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining, pp. 333–342.

[31] Z. Zhao, L. Wang, H. Liu, Efficient spectral feature selection with minimum redundancy, in: Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence.

[32] Z. Li, Y. Yang, J. Liu, X. Zhou, H. Lu, Unsupervised feature selection using nonnegative spectral analysis., in: Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence.

[33] M. Qian, C. Zhai, Robust unsupervised feature selection, in: Proceedings of the Twenty-Third international joint conference on Artificial Intelligence, pp. 1621–1627.

[34] P. Zhu, Q. Hu, C. Zhang, W. Zuo, Coupled dictionary learning for unsupervised feature selection, in: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence.

[35] H. Liu, M. Shao, Y. Fu, Consensus guided unsupervised feature selection, in: Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence.

[36] D. Cai, X. He, J. Han, T.S. Huang, Graph regularized nonnegative matrix factorization for data representation, IEEE Trans. Pattern Anal. Mach. Intell. 33 (2011) 1548–1560.

[37] J. Wu, H. Liu, H. Xiong, J. Cao, J. Chen, K-means-based consensus clustering: a unified view, IEEE Trans. Knowl. Data Eng. 27 (2015) 155–169.

[38] E. Elhamifar, R. Vidal, Sparse subspace clustering: algorithm, theory, and applications, IEEE Trans. Pattern Anal. Mach. Intell. 35 (2013) 2765–2781.

[39] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, Y. Ma, Robust recovery of subspace structures by low-rank representation, IEEE Trans. Pattern Anal. Mach. Intell. 35 (2013) 171–184.

[40] T. Zhang, A. Szlam, Y. Wang, G. Lerman, Hybrid linear modeling via local best-fit flats, Int. J. Comput. Vis. 100 (2012) 217–240.

[41] A.M. Martinez, The ar face database, CVC Technical Report 24, 1998.

[42] S. Boyd, L. Vandenberghe, Convex optimization, Cambridge university press, 2004.

[43] Y. Yang, H. T. Shen, Z. Ma, Z. Huang, X. Zhou, l2,1-norm regularized discriminative feature selection for unsupervised learning, in: IJCAI Proceedings-International Joint Conference on Artificial Intelligence, pp. 1589–1594.

[44] Z. Zhao, H. Liu, Spectral feature selection for supervised and unsupervised learning, in: Proceedings of the 24th international conference on Machine learning, pp. 1151–1157.

Pengfei Zhu (**M'15**) received the Ph.D. degree from The Hong Kong Polytechnic University, Hong Kong, China, in 2015. He received his B. S. and M. S. from Harbin Institute of Technology, Harbin, China in 2009 and 2011, respectively. Now he is an associate professor with School of Computer Science and Technology, Tianjin University. His research interests are focused on machine learning and computer vision.

Wencheng Zhu received B. S. degree from Tianjin University, Tianjin, China in 2014. Now he is a Master student with School of Computer Science and Technology, Tianjin University. His research interests are focused on machine learning and data mining.

Qinghua Hu (**M'11**) received B. S., M. S. and PhD degrees from Harbin Institute of Technology, Harbin, China in 1999, 2002 and 2008, respectively. He was an associate professor with Harbin Institute of Technology from 2008 to 2011. Now he is a full professor with School of Computer Science and Technology, Tianjin University. His research interests are focused on intelligent modeling, data mining, knowledge discovery for classification and regression. He is a PC co-chair of RSCTC 2010 and severs as referee for a great number of journals and conferences. He has published more than 70 journal and conference papers in the areas of pattern recognition and fault diagnosis.

Changqing Zhang received the B.S. and M.E. degrees in computer science from Sichuan University in 2005 and 2008, and the Ph.D. degree from Tianjin University in 2016, respectively. He is currently an Assistant Professor with the School of Computer Science and Technology, Tianjin University. His current research

Wangmeng Zuo (**M'09**) received the Ph.D. degree in computer application technology from the Harbin Institute of Technology, Harbin, China, in 2007. From July 2004 to December 2004, from November 2005 to August 2006, and from July 2007 to February 2008, he was a Research Assistant in the Department of Computing, Hong Kong Polytechnic University. From August 2009 to February 2010, he was a Visiting Professor in Microsoft Research Asia. He is currently a Professor in the School of Computer Science and Technology, Harbin Institute of Technology. His current research interests include sparse representation, biometrics, pattern recognition, and computer vision. Dr. Zuo is an Associate Editor of the *IET Biometrics*.