Neurocomputing 290 (2018) 87–99

Contents lists available at ScienceDirect

Neurocomputing

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

Feature selection based dual-graph sparse non-negative matrix factorization for local discriminative clustering

Yang Meng[a], Ronghua Shang[a,*], Licheng Jiao[a], Wenya Zhang[b], Yijing Yuan[a], Shuyuan Yang[a]

[a] Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, Xidian University, Xi'an, Shaanxi 710071, China
[b] School of Computer Sciences, Xidian University, Xi'an, Shaanxi 710071, China

ARTICLE INFO

Article history:
Received 20 November 2016
Revised 10 March 2017
Accepted 6 February 2018
Available online 15 February 2018

Communicated by Feiping Nie

keywords:
Dual-graph
Non-negative matrix factorization
Local discriminative
Feature selection
Clustering

ABSTRACT

Non-negative matrix factorization (NMF) can map high-dimensional data into a low-dimensional data space. Feature selection can eliminate the redundant and irrelevant features from the alternative features. In this paper, we propose a feature selection based dual-graph sparse non-negative matrix factorization (DSNMF) which can find an appropriate low dimensional representation of data by NMF and then select more discriminative features to further reduce the dimension of the low dimensional space by feature selection rather than reduce the dimension by only NMF or feature selection in many previous methods. DSNMF combines dual-graph model with non-negative matrix factorization, which can not only simultaneously preserve the geometric structures in both the data space and the feature space, but also make the two non-negative matrix factors update iteratively and interactively. In addition, DSNMF exerts $L_{2,1}$-*norm* constraint on the non-negative matrix factor of the feature space to make full use of the sparse self-representation information. What's more, we propose a new local discriminative feature selection clustering called feature selection based dual-graph sparse non-negative matrix factorization for local discriminative clustering (DSNMF-LDC) whose clustering effects are better. We give the objective function, the iterative updating rules and the convergence proof. Our empirical study shows that DSNMF-LDC is robust and excellent in comparison to 9 feature selection algorithms and 7 clustering algorithms in clustering accuracy (ACC) and normalized mutual information (NMI).

1. Introduction

With the rapid development of information technology and computer, the size of the collected data in many fields has increased. The massive high-dimensional data have put forward severe challenge to the traditional machine learning and the statistical analysis [1]. Feature selection aims to eliminate the redundant and irrelevant features, identify and preserve the discriminative features from the high-dimensional data [2–5]. It can reduce the feature dimensions, simplify the calculation model and improve the model accuracy, running efficiency and learning performance [6]. The discriminative features got from feature selection can be used in clustering to improve the quality of clustering. How to explore the inherent rules and the essential structure in the high-dimensional data, to efficiently obtain the useful features and represent them in low dimensions have become hot issues in the machine learning, pattern recognition, data mining and statistical analysis. It has wide applications in people's life, such as text classification [7], medical diagnosis [8,9], video event detection [10], intrusion detection [11] and some other fields.

Many studies about feature selection algorithms have shown that the data information generally distributes in the nonlinear low-dimensional submanifold of the high-dimensional space, so the researchers put forward a lot of manifold learning methods to discover the potential geometric structure [12–17]. The main idea of Laplacian score (LapScore) [12] and spectral feature selection (SPEC) [13] is to evaluate each feature according to the local preserving strategy and remove the poor features. However, they do not use learning mechanism. Multi-cluster feature selection (MCFS) [14] and minimum redundancy spectral feature selection (MRSF) [15] combine the embedded learning with the sparse constraints, and their difference lies in the different sparse constraints. MCFS uses L_1-*norm* constraint and MRSF uses advanced $L_{2,1}$-*norm* constraint. However, MCFS and MRSF both belong to the step-by-step feature selection, which cannot take the effect of the manifold information on the following feature selection into account. Joint embedding learning and sparse regression feature selection (JELSR) [16] combines the embedded learning with spectral regression, which can effectively preserve the discriminative features. Locality and similarity preserving embedding feature

* Corresponding author.
E-mail addresses: rhshang@mail.xidian.edu.cn, 280331025@qq.com (R. Shang).

https://doi.org/10.1016/j.neucom.2018.02.044
0925-2312/

selection (LSPE) [17] combines the embedded learning with feature selection, which can preserve the discriminative features and excavate the geometric information of the data space. The aforementioned methods can only preserve the local manifold information of the data space and cannot make full use of the local manifold information of the feature space, which cannot completely excavate the potential information. In Ref. [18], Chang et al. have proposed a convex sparse principal component analysis (CSPCA) algorithm which adopts the recent advances of sparsity and robust PCA into a joint framework to leverage the mutual benefit. CSPCA is the first convex sparse and robust PCA algorithm, which can always ensure the algorithm achieves the global optimum. In Ref. [1], an advanced self-representation based dual-graph regularized feature selection clustering (DFSC) has been proposed to solve this problem. DFSC utilizes the dual-graph (the data graph and the feature graph) model which considers the manifold information both of the data space and the feature space to make full use of the geometric structure of the data. DFSC obviously outperforms the previous algorithms at the clustering effects. However, from the updating rules in DFSC, we can see the self-representation coefficients matrices in feature space and data space can only update by themselves rather than affect each other for the self-representation model, which cannot give full play to the dual-graph model.

Clustering is divided into several categories according to the preset clustering number, so as to make the similarities of elements in the same class as large as possible, and make the similarities of elements in the different classes as small as possible [19–21]. To evaluate the performance of feature selection algorithms, we often need to cluster according to the selected features. There are some common clustering algorithms, such as K-means, NMF [22,23], dual regularized co-clustering (DRCC) [24], concept factorization (CF) [25], locally consistent concept factorization (LCCF) [26] and dual-graph regularized concept factorization clustering (GCF) [27]. K-means is one of the commonest clustering algorithms, which is simple and easy to understand, but its performance will significantly decrease in dealing with high-dimensional problems. To solve this problem, linear discriminant analysis (LDA) has been combined with K-means, which can effectively use the discriminative information and achieve good results in data clustering. The purpose of NMF [22,23] is to decompose the input data into two low-dimensional matrices of the data space and the feature space. Inspired by NMF, DRCC [24] does tri-factorization on the input data. In addition, DRCC adopts the dual-graph model to effectively utilize the potential information. CF [25] is an extension of NMF and applies the idea of the kernel method [26], which can be used in the datasets containing negative values. Based on CF, Cai et al. have proposed LCCF [27] which can preserve the geometrical manifold structures. Ye and Jin have proposed GCF [28] which adds dual-graph model into LCCF. GCF can preserve the geometrical manifold information both of the data graph and the feature graph. However, the aforementioned clustering algorithms do not apply the feature selection in advance when dealing with high-dimensional data, so there are some redundant and irrelevant features in the original features.

To solve the problem in the aforementioned feature selection algorithms [12–17], we propose a feature selection based dual-graph sparse non-negative matrix factorization (DSNMF). Dual-graph model is added in DSNMF, which can preserve the local geometric information of both the data space and the feature space, and fully excavate the potential data information. To solve the problem in the recently proposed DFSC [1], DSNMF adopts non-negative matrix factorization rather than self-representation matrices. Therefore, it can make the two non-negative matrix factors of the data space and the feature space update iteratively and interactively, which can give full play to the dual-graph model.

Considering that the previous clustering algorithms [22–28] are lack of discrimination, we combine K-means with LDA [29–31] and propose a feature selection based dual-graph sparse non-negative matrix factorization for local discriminative clustering (DSNMF-LDC). DSNMF-LDC not only has the advantages of the dual-graph model in the co-clustering algorithms, but also utilizes the feature selection which can remove some redundant and irrelevant features in the original features and select the discriminative features. Therefore, DSNMF-LDC can not only be robust to the noises, but also reduce the dimension of the data, save computing and storage resources. What's more, we utilize the local discriminative feature selection clustering after feature selection, which can greatly improve the clustering effectiveness and robustness.

Our main contributions are the following four aspects:

- We integrate NMF into feature selection rather than reduce the dimension by only NMF or feature selection, which can reduce the dimension of data as much as possible and efficiently deal with high-dimensional data.
- We combine the non-negative matrix factorization with the dual-graph (the data graph and the feature graph) model for feature selection, which can reduce the dimension of data as much as possible. The two non-negative matrix factors of the data space and the feature space can update iteratively and interactively, which can give full play to the dual-graph model.
- We exert $L_{2,1}$-*norm* constraint on the non-negative matrix factor of the feature space, which reflects the sparse self-representation information and the importance of selected features. What's more, it ensures the sparsity of the non-negative matrix factor of the feature space, which can simplify the calculation.
- We utilize the local discriminative feature selection clustering after feature selection, which can greatly improve the clustering effectiveness and robustness.

The rest of this paper is organized as follows: in Section 2, we introduce our feature selection based dual-graph sparse non-negative matrix factorization for local discriminative clustering (DSNMF-LDC) in detail. Extensive experiments and corresponding analyses are done in Section 3. In Section 4, we provide some conclusions and suggestions for the future work.

2. Feature selection based dual-graph sparse non-negative matrix factorization for local discriminative clustering (DSNMF-LDC)

2.1. Feature selection based dual-graph sparse non-negative matrix factorization (DSNMF)

2.1.1. Objective function

An advanced DFSC has been proposed in [1], which exerts the self-representation matrices of the data space and the feature space. However, it can only update by themselves and cannot affect each other, so that it cannot give full play to the dual-graph model. In order to solve this problem, we propose a feature selection based dual-graph sparse non-negative matrix factorization (DSNMF).

Non-negative matrix factorization can obtain the potential data information by decomposing the data matrix into non-negative matrix factors, which is a very effective method of the matrix approximation. We have a dataset $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \in \Re^{m \times n}$, where $\mathbf{x}_i = [x_{i1}, x_{i2}, \ldots, x_{im}]^T \in \Re^m$ is the ith vector, m is the number of the feature dimensions, n is the number of the simples. The purpose of non-negative matrix factorization is to decompose the data matrix into two non-negative matrix factors $\boldsymbol{P} = [\boldsymbol{P}_1, \boldsymbol{P}_2, \ldots, \boldsymbol{P}_m]^T \in \Re^{m \times c}$ and $\boldsymbol{S} = [\boldsymbol{S}_1, \boldsymbol{S}_2, \ldots, \boldsymbol{S}_n]^T \in \Re^{n \times c}$, where c is the clustering number, $\boldsymbol{P}$ is the non-negative matrix factor of the feature space

and $\boldsymbol{S}$ is the non-negative matrix factor of the data space. Each element in the non-negative matrix can be calculated as follows:

$$x_{ij} = \sum_{a=1}^{k} P_{ia} S_{aj}^{T} + f_{ij} \tag{1}$$

where f_{ij} is the residual error term of each element after the non-negative matrix factorization. The formula (1) can be expanded into a matrix form:

$$\boldsymbol{X} = \mathbf{P}\mathbf{S}^{T} + \mathbf{F} \tag{2}$$

where $\boldsymbol{F}$ is the residual error matrix. We aim to minimize the residual error matrix $\boldsymbol{F}$:

$$\min ||\mathbf{X} - \mathbf{P}\mathbf{S}^{T}||_{F}^{2} \tag{3}$$

where $||.||_F$ is the Frobenius norm (F-norm) of the matrix. We can get the Euclidean distance of the two matrices by calculating the square of the F-norm.

The recent study [24,32,33] have shown that the data information distributes in the nonlinear low-dimensional submanifold of the high-dimensional space, namely the data manifold, and the feature information distributes also in a low-dimensional submanifold, namely the feature manifold. Therefore, we construct a dual-graph model (the data graph and the feature graph) to effectively simulate the geometric structure of the data manifold and the feature manifold, which can preserve the local geometric information of the data as completely as possible. There are n points in the graph and each point represents a sample data. For each point $\boldsymbol{x}_i$, we find its k nearest neighbors and establish edges among them, on which we construct edge weights to represent the similarities between $\boldsymbol{x}_i$ and its k nearest neighbors. Thus, the weight matrix $\boldsymbol{W}$ is also called similarity matrix. There are many methods to construct edge weights, such as some common methods in [22]: heat kernel weighting, binary (0–1) weighting and dot-product weighting. We can choose the most appropriate similarity matrix according to the specific situation, which can improve the learning accuracy. For example, we usually choose the heat kernel weighting to measure the similarities among the points in the image data.

We define that the similarity matrix of the feature graph is $\boldsymbol{W}^{P}$. Based on the non-negative matrix factor $\boldsymbol{P}$ of the feature space, the low-dimensional representation smoothness of the feature space is measured as follows [24]:

$$\begin{aligned}
&\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}\left\|\mathbf{P}_i - \mathbf{P}_j\right\|^{2} W_{ij}^{P} \\
&= \sum_{i=1}^{m} \mathbf{P}_i \mathbf{P}_i^{T} D_{ii}^{P} - \sum_{i=1}^{m}\sum_{j=1}^{m} \mathbf{P}_i \mathbf{P}_i^{T} W_{ij}^{P} \\
&= Tr(\mathbf{P}^{T}\mathbf{D}^{P}\mathbf{P}) - Tr(\mathbf{P}^{T}\mathbf{W}^{P}\mathbf{P}) \\
&= Tr(\mathbf{P}^{T}\mathbf{L}^{P}\mathbf{P})
\end{aligned} \tag{4}$$

From the above formula, we can know that the Laplacian matrix of the feature graph is $\boldsymbol{L}^{P} = \boldsymbol{D}^{P} - \boldsymbol{W}^{P}$, where $\boldsymbol{D}^{P}$ is a diagonal matrix and $[\mathbf{D}^{P}]_{ii} = \sum_j [\mathbf{W}^{P}]_{ij}$. Therefore, the diagonal elements of the diagonal matrix $\boldsymbol{D}^{P}$ are the sum of the row elements of the matrix $\boldsymbol{W}^{P}$.

Similarly, we define that the similarity matrix of the data graph is $\boldsymbol{W}^{S}$. Based on the non-negative matrix factor $\boldsymbol{S}$ of the data space, the representation smoothness of the data space is measured as follows [24]:

$$\begin{aligned}
&\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n}\left\|\mathbf{S}_i - \mathbf{S}_j\right\|^{2} W_{ij}^{S} \\
&= \sum_{i=1}^{n} \mathbf{S}_i \mathbf{S}_i^{T} D_{ii}^{S} - \sum_{i=1}^{n}\sum_{j=1}^{n} \mathbf{S}_i \mathbf{S}_i^{T} W_{ij}^{S} \\
&= Tr(\mathbf{S}^{T}\mathbf{D}^{S}\mathbf{S}) - Tr(\mathbf{S}^{T}\mathbf{W}^{S}\mathbf{S}) \\
&= Tr(\mathbf{S}^{T}\mathbf{L}^{S}\mathbf{S})
\end{aligned} \tag{5}$$

Considering the non-negative matrix factorization and the dual-graph model, and the $L_{2,1}$-*norm* constraint on the non-negative matrix factor $\boldsymbol{P}$ of the feature space to ensure that the row sparsity of the matrix $\boldsymbol{P}$, we propose a novel feature selection, namely feature selection based dual-graph sparse non-negative matrix factorization (DSNMF). $||\mathbf{P}_i||_2$ contains the self-representation information of the ith feature, which reflects its importance in all the features. We can take it as an important criterion of the feature selection and add it into the following objective function:

$$\begin{aligned}
J(\mathbf{S},\mathbf{P}) = &\left\|\mathbf{X} - \mathbf{P}\mathbf{S}^{T}\right\|_{F}^{2} + \alpha Tr\left(\mathbf{S}^{T}\mathbf{L}^{S}\mathbf{S}\right) \\
&+\beta Tr\left(\mathbf{P}^{T}\mathbf{L}^{P}\mathbf{P}\right) + \theta\|\mathbf{P}\|_{2,1} \\
&s.t. \quad \mathbf{S} \geq 0, \mathbf{P} \geq 0
\end{aligned} \tag{6}$$

where α,β are the non-negative regular parameters. The regular parameters α,β can balance the weight of the first reconstruction error term and the next several terms. When $\beta = \theta = 0$, the formula (6) will degenerate as Cai et al. proposed graph regularized non-negative matrix factorization (GNMF) in [22]. When $\alpha = \beta = \theta = 0$, the formula (6) will degenerate as the traditional non-negative matrix factorization (NMF).

Based on the final non-negative matrix factor $\boldsymbol{P}$ of the feature space, we select the former q features according to the value of $||\mathbf{P}_i||_2$ that is feature selection based dual-graph sparse non-negative matrix factorization (DSNMF).

2.1.2. Iterative updating rules

The iterative updating method is commonly used to solve the non-convex problem. The iterative updating method fixes the newest value of a variable and solves the convex optimization problem of another variable at each iteration, in which we can obtain the stable solution or the local optimal solution of the non-convex problem.

The objective function in the formula (6) is a non-convex problem on the non-negative matrix factors $\boldsymbol{P}$ and $\boldsymbol{S}$, so the iterative updating rules are used to solve this problem. We fix the non-negative matrix factor $\boldsymbol{P}$ when solving the non-negative matrix factor $\boldsymbol{S}$ of the data space and fix the non-negative matrix factor $\boldsymbol{S}$ when solving the non-negative matrix factor $\boldsymbol{P}$ of the feature space. We rewrite the formula (6) as:

$$\begin{aligned}
J(\mathbf{S},\mathbf{P}) = &\left\|\mathbf{X} - \mathbf{P}\mathbf{S}^{T}\right\|_{F}^{2} + \alpha Tr\left(\mathbf{S}^{T}\mathbf{L}^{S}\mathbf{S}\right) \\
&+\beta Tr\left(\mathbf{P}^{T}\mathbf{L}^{P}\mathbf{P}\right) + \theta\|\mathbf{P}\|_{2,1} \\
&= Tr\left(\mathbf{X}^{T}\mathbf{X} - 2\mathbf{X}^{T}\mathbf{P}\mathbf{S}^{T} + \mathbf{S}\mathbf{P}^{T}\mathbf{P}\mathbf{S}^{T}\right) \\
&+\alpha Tr\left(\mathbf{S}^{T}\mathbf{L}^{S}\mathbf{S}\right) + \beta Tr\left(\mathbf{P}^{T}\mathbf{L}^{P}\mathbf{P}\right) + \theta\|\mathbf{P}\|_{2,1} \\
&= Tr\left(\mathbf{X}^{T}\mathbf{X}\right) - 2Tr\left(\mathbf{X}^{T}\mathbf{P}\mathbf{S}^{T}\right) + Tr\left(\mathbf{S}\mathbf{P}^{T}\mathbf{P}\mathbf{S}^{T}\right) \\
&+\alpha Tr\left(\mathbf{S}^{T}\mathbf{L}^{S}\mathbf{S}\right) + \beta Tr\left(\mathbf{P}^{T}\mathbf{L}^{P}\mathbf{P}\right) + \theta\|\mathbf{P}\|_{2,1} \\
&s.t. \quad \mathbf{S} \geq 0, \mathbf{P} \geq 0
\end{aligned} \tag{7}$$

Let Ψ_{ij} and Φ_{kj} be the corresponding Lagrange multipliers for the constraint on $S_{ij} \geq 0$ and $P_{kj} \geq 0$, respectively. Then we have the Lagrange function of the formula (7) as follows:

$$\begin{aligned}
L(\mathbf{S},\mathbf{P}) = &Tr\left(\mathbf{X}^{T}\mathbf{X}\right) - 2Tr\left(\mathbf{X}^{T}\mathbf{P}\mathbf{S}^{T}\right) + Tr\left(\mathbf{S}\mathbf{P}^{T}\mathbf{P}\mathbf{S}^{T}\right) \\
&+\alpha Tr\left(\mathbf{S}^{T}\mathbf{L}^{S}\mathbf{S}\right) + \beta Tr\left(\mathbf{P}^{T}\mathbf{L}^{P}\mathbf{P}\right) + \theta Tr\left(\mathbf{P}^{T}\mathbf{V}\mathbf{P}\right) \\
&+Tr\left(\Psi\mathbf{S}^{T}\right) + Tr\left(\Phi\mathbf{P}^{T}\right)
\end{aligned} \tag{8}$$

where the ith diagonal element V_{ii} of the diagonal matrix $\mathbf{V} \in \Re^{m \times m}$ can be calculated as follows:

$$V_{ii} = \frac{1}{2\|\mathbf{P}_i\|_2} \tag{9}$$

In order to avoid overflow, we add a small enough constant ε in the definition of the matrix $\boldsymbol{V}$, so the formula (9) can be rewritten as follows:

$$V_{ii} = \frac{1}{2\max(\|\boldsymbol{P}_i\|_2, \varepsilon)} \tag{10}$$

When updating the variable $\boldsymbol{P}$, the partial derivative of $L(\boldsymbol{S},\boldsymbol{P})$ with respect to $\boldsymbol{P}$ is:

$$\frac{\partial L(\mathbf{S},\mathbf{P})}{\partial \mathbf{P}} = -2\mathbf{X}\mathbf{S} + 2\mathbf{P}\mathbf{S}^T\mathbf{S} + 2\beta\mathbf{L}^P\mathbf{P} + 2\theta\mathbf{V}\mathbf{P} + \Phi \tag{11}$$

Considering the KKT conditions $\Phi_{kj}P_{kj} = 0$, we have:

$$\left[-2\mathbf{X}\mathbf{S} + 2\mathbf{P}\mathbf{S}^T\mathbf{S} + 2\beta\mathbf{L}^P\mathbf{P} + 2\theta\mathbf{V}\mathbf{P}\right]_{ij}P_{ij} = 0 \tag{12}$$

Since $\boldsymbol{L}^P = \boldsymbol{D}^P - \boldsymbol{W}^P$ and the elements of $\boldsymbol{D}^P$ and $\boldsymbol{W}^P$ are non-negative, the formula (12) can be rewritten as follows:

$$\left[-2\mathbf{X}\mathbf{S} + 2\mathbf{P}\mathbf{S}^T\mathbf{S} + 2\beta\mathbf{D}^P\mathbf{P} - 2\beta\mathbf{W}^P\mathbf{P} + 2\theta\mathbf{V}\mathbf{P}\right]_{ij}P_{ij} = 0 \tag{13}$$

So that we can get the following updating formula of the variable $\boldsymbol{P}$:

$$P_{ij} \leftarrow P_{ij}\frac{\left[\mathbf{X}\mathbf{S}+\beta\mathbf{W}^P\mathbf{P}\right]_{ij}}{[\mathbf{P}\mathbf{S}^T\mathbf{S} + \beta\mathbf{D}^P\mathbf{P} + \theta\mathbf{V}\mathbf{P}]_{ij}} \tag{14}$$

Similarly, when updating the variable $\boldsymbol{S}$, the partial derivative of $L(\boldsymbol{S}, \boldsymbol{P})$ with respect to $\boldsymbol{S}$ is:

$$\frac{\partial L(\mathbf{S},\mathbf{P})}{\partial \mathbf{S}} = -2\mathbf{X}^T\mathbf{P} + 2\mathbf{S}\mathbf{P}^T\mathbf{P} + 2\alpha\mathbf{L}^S\mathbf{S} + \Psi \tag{15}$$

Considering the KKT conditions $\Psi_{ij}S_{ij} = 0$, we have:

$$\left[-2\mathbf{X}^T\mathbf{P} + 2\mathbf{S}\mathbf{P}^T\mathbf{P} + 2\alpha\mathbf{L}^S\mathbf{S}\right]_{ij}S_{ij} = 0 \tag{16}$$

Since $\boldsymbol{L}^S = \boldsymbol{D}^S - \boldsymbol{W}^S$ and the elements of $\boldsymbol{D}^S$ and $\boldsymbol{W}^S$ are non-negative, the formula (16) can be written as follows:

$$\left[-2\mathbf{X}^T\mathbf{P} + 2\mathbf{S}\mathbf{P}^T\mathbf{P} + 2\alpha\mathbf{D}^S\mathbf{S} - 2\alpha\mathbf{W}^S\mathbf{S}\right]_{ij}S_{ij} = 0 \tag{17}$$

So that we can get the following updating formula of the variable $\boldsymbol{S}$:

$$S_{ij} \leftarrow S_{ij}\frac{\left[\mathbf{X}^T\mathbf{P}+\alpha\mathbf{W}^S\mathbf{S}\right]_{ij}}{[\mathbf{S}\mathbf{P}^T\mathbf{P} + \alpha\mathbf{D}^S\mathbf{S}]_{ij}} \tag{18}$$

2.2. Local discriminative feature selection clustering

Clustering is divided into several categories according to the pre-set clustering number, so as to make the similarities of elements in the same class as large as possible, and make the similarities of elements in the different classes as small as possible [19–21]. In DSNMF, we select the former q features of the dataset $\boldsymbol{X}$ according to the value of $\|\mathbf{P}_i\|_2$ in a descending order, so that we can obtain the matrix $\mathbf{X}_f = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n] \in \Re^{q\times n}$ after the feature selection. Then, we cluster the dataset $\boldsymbol{X}_f$. The dataset $\boldsymbol{X}_f$ is divided into c clusters $\{C_i\}_{i=1}^c$, so as to obtain a mark matrix $\mathbf{M} = [\mathbf{m}_1, \mathbf{m}_2, \ldots, \mathbf{m}_n]^T \in \{0,1\}^{n\times c}$. The jth element of $\boldsymbol{m}_i$ is m_{ij} which is 1 if $\mathbf{x}_i \in C_j$ and 0 otherwise. We use the indirect method like it in [35] to solve the mark matrix instead of the direct method.

We firstly define a cluster assignment matrix $\boldsymbol{Z}$ as follows:

$$\mathbf{Z} = [\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_n]^T = \mathbf{M}\left(\mathbf{M}^T\mathbf{M}\right)^{-1/2} \tag{19}$$

where $\boldsymbol{z}_i$ is the cluster assignment vector for the data vector $\boldsymbol{x}_i$. Since $\boldsymbol{M}^T\boldsymbol{M}$ is a diagonal matrix, we can easily conclude as follows:

$$\mathbf{Z}^T\mathbf{Z} = \left(\mathbf{M}^T\mathbf{M}\right)^{-1/2}\mathbf{M}^T\mathbf{M}\left(\mathbf{M}^T\mathbf{M}\right)^{-1/2} = \mathbf{I}_n \tag{20}$$

We define the between-cluster scatter matrix $\boldsymbol{S}_b$, the within-cluster scatter matrix $\boldsymbol{S}_w$ and the total scatter matrix $\boldsymbol{S}_t$ like [31] as follows:

$$\mathbf{S}_b = \tilde{\mathbf{X}}_f\mathbf{Z}\mathbf{Z}^T\tilde{\mathbf{X}}_f^T \tag{21}$$

$$\mathbf{S}_w = \tilde{\mathbf{X}}_f\tilde{\mathbf{X}}_f^T - \tilde{\mathbf{X}}_f\mathbf{Z}\mathbf{Z}^T\tilde{\mathbf{X}}_f^T \tag{22}$$

$$\mathbf{S}_t = \mathbf{S}_b + \mathbf{S}_w = \tilde{\mathbf{X}}_f\tilde{\mathbf{X}}_f^T \tag{23}$$

where $\tilde{\mathbf{X}}_f = \mathbf{X}_f\mathbf{H}_n$. We define a centering matrix $\mathbf{H}_n = \mathbf{I}_n - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^T \in \Re^{n\times n}$, where $\boldsymbol{I}_n$ is the n-dimensional unit matrix and all elements of the column vector $\mathbf{1}_n$ is 1. In order to better cluster that makes the distance between the data in different clusters as far as possible and the data in same cluster as close as possible, we utilize the Fisher criterion and obtain the best cluster assignment matrix $\boldsymbol{Z}_{best}$ by maximizing the following objective function:

$$\begin{aligned}\mathbf{Z}_{best} &= \arg\max_{\mathbf{Z}} Tr\left[(\mathbf{S}_t + \mu\mathbf{I})^{-1}\mathbf{S}_b\right] \\ &= \arg\max_{\mathbf{Z}} Tr\left[\left(\tilde{\mathbf{X}}_f\tilde{\mathbf{X}}_f^T + \mu\mathbf{I}\right)^{-1}\tilde{\mathbf{X}}_f\mathbf{Z}\mathbf{Z}^T\tilde{\mathbf{X}}_f^T\right] \\ &= \arg\max_{\mathbf{Z}} Tr\left[\mathbf{Z}^T\tilde{\mathbf{X}}_f^T\left(\tilde{\mathbf{X}}_f\tilde{\mathbf{X}}_f^T + \mu\mathbf{I}\right)^{-1}\tilde{\mathbf{X}}_f\mathbf{Z}\right] \\ &\quad s.t. \quad \mu > 0\end{aligned} \tag{24}$$

Since $\boldsymbol{Z}^T\boldsymbol{Z} = \boldsymbol{I}_n$ in the formula (20), we can obtain the best cluster assignment matrix $\boldsymbol{Z}_{best}$ by minimizing the following objective function:

$$\begin{aligned}\mathbf{Z}_{best} &= \arg\min_{\mathbf{Z}} Tr\left[\mathbf{Z}^T\mathbf{Z} - \mathbf{Z}^T\tilde{\mathbf{X}}_f^T\left(\tilde{\mathbf{X}}_f\tilde{\mathbf{X}}_f^T + \mu\mathbf{I}\right)^{-1}\tilde{\mathbf{X}}_f\mathbf{Z}\right] \\ &\quad s.t. \quad \mu > 0\end{aligned} \tag{25}$$

Since the local manifold information of the data is similar to be linear [36], we adopt a local clique $N_k(\boldsymbol{x}_i)$ comprising the k nearest neighbors of a data point $\boldsymbol{x}_i$, which contains the data point $\boldsymbol{x}_i$ itself and its $(k-1)$ nearest neighbors. Therefore, we establish such a local linear discriminative model to evaluate the clustering results.

We define a matrix $\mathbf{X}_i = [x_{i_1}, x_{i_2}, \ldots, x_{i_k}]$ which consists of all data points in the local clique $N_k(\boldsymbol{x}_i)$ and define the corresponding index set as $D_i = \{i_1, i_2, \ldots, i_k\}$. Similarly, according to the local linear discriminative model, we define $\boldsymbol{Z}_i = [\mathbf{z}_{i_1}, \mathbf{z}_{i_2}, \ldots, \mathbf{z}_{i_k}]^T \in \Re^{k\times c}$ which is derived from the assignment matrix cluster $\boldsymbol{Z}$ as follows:

$$\mathbf{Z}_i = \mathbf{A}_i^T\mathbf{Z} \tag{26}$$

where $\boldsymbol{A}_i$ is a selection matrix, $(\boldsymbol{A}_i)_{xy}$ is 1 if $x = D_i\{y\}$ and 0 otherwise.

As the formula (24), we utilize the Fisher criterion and obtain the best cluster assignment matrix $(\boldsymbol{Z}_i)_{best}$ of the local linear discriminative model by maximizing the following objective function:

$$\begin{aligned}(\mathbf{Z}_i)_{best} &= \arg\max_{\mathbf{Z}_i} Tr\left[\mathbf{Z}_i^T\tilde{\mathbf{X}}_i^T\left(\tilde{\mathbf{X}}_i\tilde{\mathbf{X}}_i^T + \mu\mathbf{I}\right)^{-1}\tilde{\mathbf{X}}_i\mathbf{Z}_i\right] \\ &\quad s.t. \quad \mu > 0\end{aligned} \tag{27}$$

where $\tilde{\mathbf{X}}_i = \mathbf{X}_i\mathbf{H}_k$.

As the formula (25), we can also obtain the best cluster assignment matrix $(\boldsymbol{Z}_i)_{best}$ of the local linear discriminative model by minimizing the following objective function:

$$\begin{aligned}(\mathbf{Z}_i)_{best} &= \arg\min_{\mathbf{Z}_i} Tr\left[\mathbf{Z}_i^T\mathbf{H}_k\mathbf{Z}_i - \mathbf{Z}_i^T\tilde{\mathbf{X}}_i^T\left(\tilde{\mathbf{X}}_i\tilde{\mathbf{X}}_i^T + \mu I\right)^{-1}\tilde{\mathbf{X}}_i\mathbf{Z}_i\right] \\ &\quad s.t. \quad \mu > 0\end{aligned} \tag{28}$$

Since $\underset{i}{\tilde{\mathbf{X}}}\mathbf{H}_k = \underset{i}{\tilde{\mathbf{X}}}$, we have

$$\begin{aligned}&\mathbf{Z}_i^T\tilde{\mathbf{X}}_i^T\left(\tilde{\mathbf{X}}_i\tilde{\mathbf{X}}_i^T + \mu\mathbf{I}_m\right)^{-1}\tilde{\mathbf{X}}_i\mathbf{Z}_i \\ &= \mathbf{Z}_i^T\mathbf{H}_k^T\tilde{\mathbf{X}}_i^T\left(\tilde{\mathbf{X}}_i\tilde{\mathbf{X}}_i^T + \mu\mathbf{I}\right)^{-1}\tilde{\mathbf{X}}_i\mathbf{H}_k\mathbf{Z}_i \\ &= \mathbf{Z}_i^T\mathbf{H}_k\tilde{\mathbf{X}}_i^T\left(\tilde{\mathbf{X}}_i\tilde{\mathbf{X}}_i^T + \mu\mathbf{I}\right)^{-1}\tilde{\mathbf{X}}_i\end{aligned}$$

$$\times\left(\tilde{\mathbf{X}}_i^T\tilde{\mathbf{X}}_i+\mu\mathbf{I}\right)\left(\tilde{\mathbf{X}}_i^T\tilde{\mathbf{X}}_i+\mu\mathbf{I}\right)^{-1}\mathbf{H}_k\mathbf{Z}_i$$
$$=\mathbf{Z}_i^T\mathbf{H}_k\tilde{\mathbf{X}}_i^T\left(\tilde{\mathbf{X}}_i\tilde{\mathbf{X}}_i^T+\mu\mathbf{I}\right)^{-1}\left(\tilde{\mathbf{X}}_i\tilde{\mathbf{X}}_i^T+\mu\mathbf{I}\right)$$
$$\times\tilde{\mathbf{X}}_i\left(\tilde{\mathbf{X}}_i^T\tilde{\mathbf{X}}_i+\mu\mathbf{I}\right)^{-1}\mathbf{H}_k\mathbf{Z}_i$$
$$=\mathbf{Z}_i^T\mathbf{H}_k\tilde{\mathbf{X}}_i^T\tilde{\mathbf{X}}_i\left(\tilde{\mathbf{X}}_i^T\tilde{\mathbf{X}}_i+\mu\mathbf{I}\right)^{-1}\mathbf{H}_k\mathbf{Z}_i$$
$$=\mathbf{Z}_i^T\mathbf{H}_k\left(\tilde{\mathbf{X}}_i^T\tilde{\mathbf{X}}_i+\mu\mathbf{I}-\mu\mathbf{I}\right)\left(\tilde{\mathbf{X}}_i^T\tilde{\mathbf{X}}_i+\mu\mathbf{I}\right)^{-1}\mathbf{H}_k\mathbf{Z}_i$$
$$=\mathbf{Z}_i^T\left[\mathbf{H}_k-\mu\mathbf{H}_k\left(\tilde{\mathbf{X}}_i^T\tilde{\mathbf{X}}_i+\mu\mathbf{I}\right)^{-1}\mathbf{H}_k\right]\mathbf{Z}_i \tag{29}$$

Therefore, the formula (28) can be converted to:

$$(\mathbf{Z}_i)_{best}=\arg\min_{\mathbf{Z}_i}Tr\left[\mathbf{Z}_i^T\mathbf{H}_k\mathbf{Z}_i-\mathbf{Z}_i^T\tilde{\mathbf{X}}_i^T\left(\tilde{\mathbf{X}}_i\tilde{\mathbf{X}}_i^T+\mu\mathbf{I}\right)^{-1}\tilde{\mathbf{X}}_i\mathbf{Z}_i\right]$$
$$=\arg\min_{\mathbf{Z}_i}Tr\{\mathbf{Z}_i^T\mathbf{H}_k\mathbf{Z}_i-\mathbf{Z}_i^T\left[\mathbf{H}_k-\mu\mathbf{H}_k\left(\tilde{\mathbf{X}}_i^T\tilde{\mathbf{X}}_i+\mu\mathbf{I}\right)^{-1}\mathbf{H}_k\right]\mathbf{Z}_i\}$$
$$=\arg\min_{\mathbf{Z}_i}Tr\left[\mathbf{Z}_i^T\mathbf{H}_k\left(\tilde{\mathbf{X}}_i^T\tilde{\mathbf{X}}_i+\mu\mathbf{I}\right)^{-1}\mathbf{H}_k\mathbf{Z}_i\right]$$
$$s.t.\quad \mu>0 \tag{30}$$

We can rewrite the formula (30) as follows:

$$(\mathbf{Z}_i)_{best}=\arg\min_{\mathbf{Z}_i}Tr\left[\mathbf{Z}_i^T\mathbf{L}_i\mathbf{Z}_i\right] \tag{31}$$

where

$$\mathbf{L}_i=\mathbf{H}_k\left(\tilde{\mathbf{X}}_i^T\tilde{\mathbf{X}}_i+\mu\mathbf{I}\right)^{-1}\mathbf{H}_k \tag{32}$$

In order to get the global best cluster assignment matrix $\boldsymbol{Z}_{best}$, we need to globally integrate each best cluster assignment matrix $(\boldsymbol{Z}_i)_{best}$ of the local linear discriminative model. We can take the formula (26) into the formula (31) and get the following formula:

$$\mathbf{Z}_{best}=\sum_{i=1}^{n}(\mathbf{Z}_i)_{best}=\sum_{i=1}^{n}\arg\min_{\mathbf{Z}_i}Tr\left[\mathbf{Z}_i^T\mathbf{L}_i\mathbf{Z}_i\right]$$
$$=\arg\min_{\mathbf{Z}_i}\sum_{i=1}^{n}Tr\left[\mathbf{Z}_i^T\mathbf{A}_i\mathbf{L}_i\mathbf{A}_i^T\mathbf{Z}_i\right]$$
$$=\arg\min_{\mathbf{Z}_i}Tr\left[\mathbf{Z}^T\left(\sum_{i=1}^{n}\mathbf{A}_i\mathbf{L}_i\mathbf{A}_i^T\right)\mathbf{Z}^T\right] \tag{33}$$

We define:

$$\mathbf{L}=\sum_{i=1}^{n}\mathbf{A}_i\mathbf{L}_i\mathbf{A}_i^T \tag{34}$$

Then, we can rewrite the formula (33) as follows:

$$\mathbf{Z}_{best}=\arg\min_{\mathbf{Z}_i}Tr\left[\mathbf{Z}^T\mathbf{L}\mathbf{Z}^T\right]$$
$$s.t.\quad \mathbf{Z}^T\mathbf{Z}=\mathbf{I}_n,\mathbf{Z}=\mathbf{M}\left(\mathbf{M}^T\mathbf{M}\right)^{-1/2} \tag{35}$$

The objective function in the formula (35) is a NP hard problem because of the constraint on $\mathbf{Z}=\mathbf{M}(\mathbf{M}^T\mathbf{M})^{-1/2}$. Following [37], we can firstly ignore this constraint, so the formula (35) can be converted to the following formula:

$$\mathbf{Z}_{best}=\arg\min_{\mathbf{Z}_i}Tr\left[\mathbf{Z}^T\mathbf{L}\mathbf{Z}^T\right]$$
$$s.t.\quad \mathbf{Z}^T\mathbf{Z}=\mathbf{I}_n \tag{36}$$

We relax $\boldsymbol{Z}$ to the continuous-valued domain, so that eigenvalue decomposition method can be used to solve the objective function as follows:

$$\boldsymbol{L}\mathbf{u}_i=\lambda_i\mathbf{u}_i \tag{37}$$

where $\lambda_1\le\lambda_2\le\ldots\ldots\le\lambda_n$ are the eigenvalues of the matrix $\boldsymbol{L}$ and $\boldsymbol{u}_1$, $\boldsymbol{u}_2$,..., $\boldsymbol{u}_n$ are the corresponding eigenvectors. We should remove the trivial solution $\lambda_1=0$ and $\boldsymbol{u}_1=\mathbf{1}_n$, and then we get the optimal solution that consists of the former c eigenvalues according to the eigenvalues as follows:

$$\mathbf{Z}_{best}=[\mathbf{u}_1,\mathbf{u}_2,\ldots,\mathbf{u}_c] \tag{38}$$

We discretize $\boldsymbol{Z}_{best}$ using spectral rotation [34]. However, since any rotation matrix is orthogonal matrix and $\tilde{\mathbf{Z}}_{best}=\mathbf{Z}_{best}\mathbf{R}$ is the solution of the formula (36), so the solution of the objective function is not unique. Therefore, we define a mapping function to solve the optimal binary valued cluster assignment matrix $\boldsymbol{M}_{best}$ rather than the best cluster assignment matrix $\boldsymbol{Z}_{best}$:

$$\mathbf{M}_{best}=f^{-1}(\mathbf{Z}_{best})=Diag\left(\mathbf{Z}_{best}\mathbf{Z}_{best}{}^T\right)^{-1/2}\mathbf{Z}_{best} \tag{39}$$

where $\mathrm{Diag}(\boldsymbol{Z}_{best}\boldsymbol{Z}_{best}{}^T)$ is a diagonal matrix that consists of the diagonal elements of $\boldsymbol{Z}_{best}\boldsymbol{Z}_{best}{}^T$. According to the evidence available in [35]: $f^{-1}(\boldsymbol{Z}_{best}\boldsymbol{R})=\boldsymbol{M}_{best}\boldsymbol{R}$, and $\boldsymbol{M}_{best}\boldsymbol{R}$ is the optimal solution of the formula (35). We can obtain the binary valued cluster assignment matrix $\boldsymbol{M}$ and the rotation matrix $\boldsymbol{R}$ at the same time by continuous iterative optimization as follows:

$$\arg\min_{\mathbf{M},\mathbf{R}}||\mathbf{M}-\mathbf{M}_{best}\mathbf{R}||^2$$
$$s.t.\quad \mathbf{M}\mathbf{1}_c=\mathbf{1}_n,\mathbf{R}^T\mathbf{R}=\mathbf{I} \tag{40}$$

2.3. Procedure of DSNMF-LDC

The single graph model in some existing algorithms [12–17] can only preserve the local manifold structure of the data space. The self-representation matrices in DFSC can only update by themselves and cannot affect each other. In view of these existing problems, we propose a feature selection based dual-graph sparse non-negative matrix factorization for local discriminative clustering (DSNMF-LDC) which mainly consists of the feature selection based dual-graph sparse non-negative matrix factorization (DSNMF) and local discriminative feature selection clustering. The procedure of DSNMF-LDC is shown in Table 1.

2.4. Convergence analysis

We analyze the convergence of our algorithm and prove the objective function in the formula (6) decreases monotonically using the iterative updating rules (14) and (18) when giving $\boldsymbol{X}\in\Re^{m\times n}$ and any initial non-negative matrix factors $\mathbf{P}\in\Re^{m\times c}$ and $\boldsymbol{S}\in\Re^{n\times c}$.

We firstly analyze the convergence of the formula (18).

Definition 1. If the following conditions are satisfied:

$$M\left(x,x'\right)\ge N(x)\text{ and }M(x,x)=N(x) \tag{41}$$

$M(x, x')$ is an auxiliary function for $N(x)$.

Assuming that for the $(t+1)$th generation of the updating formula is as follows:

$$x^{t+1}=\arg\min_{x}M(x,x^t) \tag{42}$$

It can clearly prove $N(x^{t+1})\le M(x^{t+1},x^t)\le M(x^t,x^t)=N(x^{t+1})$.

Lemma 1.

$$M\left(S_{ij},S_{ij}^t\right)=N_{ij}\left(S_{ij}^t\right)+N_{ij}'\left(S_{ij}^t\right)\left(S_{ij}-S_{ij}^t\right)$$
$$+\frac{\left[\mathbf{S}\mathbf{P}^T\mathbf{P}+\alpha\mathbf{D}^S\mathbf{S}\right]_{ij}}{S_{ij}^t}\left(S_{ij}-S_{ij}^t\right)^2 \tag{43}$$

is the auxiliary function for N_{ij}, where $N(\mathbf{S})=\|\mathbf{X}-\mathbf{P}\mathbf{S}^T\|_F^2+\alpha Tr(\mathbf{S}^T\mathbf{L}^S\mathbf{S})$.

Proof. . Since $N_{ij}'(\mathbf{S})=[-2\mathbf{X}^T\mathbf{P}+2\mathbf{S}\mathbf{P}^T\mathbf{P}+2\alpha\mathbf{L}^S\mathbf{S}]_{ij}$ and $N_{ij}''(\mathbf{S})=2[\mathbf{P}^T\mathbf{P}]_{ij}+2\alpha\mathbf{L}_{ii}^S$, we can get the Taylor expansion of $N_{ij}(S_{ij})$:

Table 1
Procedure of DSNMF-LDC.

Input: the dataset $\mathbf{X}=[\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_n]^T\in\Re^{m\times n}$, the clustering number c, the maximum iteration number *Niter*, the regular parameters α, β, λ, the number of selected features q.
Output: the non-negative matrix factor of the feature space $\boldsymbol{P}$, the non-negative matrix factor of the data space $\boldsymbol{S}$, the clustering *label*.
1. Initialize matrix $\boldsymbol{P}$, $\boldsymbol{S}$, $\boldsymbol{V}$.
2. Update the matrices $\boldsymbol{P}$ and $\boldsymbol{S}$ according to the iterative updating rules (14) and (18). Update the matrix $\boldsymbol{V}$ according to the formula (10) and the matrix $\boldsymbol{P}$ of the moment until the convergence conditions are satisfied.
3. Select the former q features according to the value of $\|\mathbf{P}_i\|_2$ in a descending order and obtain the matrix $\mathbf{X}_f=[\mathbf{x}_1,\mathbf{x}_2,\ldots,\mathbf{x}_n]\in\Re^{q\times n}$ after the feature selection.
4. Construct the local clique of each data point.
5. Compute $\boldsymbol{L}_i$ according to the formula (32) and solve the Laplacian matrix $\boldsymbol{L}$ in the formula (34) by the global integration.
6. Solve the eigenvectors using the eigenvalue decomposition method in (37) and get the optimal solution $\mathbf{Z}_{best}=[\mathbf{u}_1,\mathbf{u}_2,\ldots,\mathbf{u}_c]$. In addition, we should pay attention to remove trivial solution $\lambda_1=0$ and $\boldsymbol{u}_1=\mathbf{1}_n$.
7. Discretize the optimal solution $\mathbf{Z}_{best}$ by the spectral rotation and then solve the optimal binary valued cluster assignment matrix $\boldsymbol{M}_{best}$ using the formula (39).
8. Finally, obtain the binary valued cluster assignment matrix $\boldsymbol{M}$ by continuous iterative optimization.

$$N(S_{ij})=N_{ij}(S_{ij}^t)+N'_{ij}(S_{ij}^t)(S_{ij}-S_{ij}^t)+\left\{[\mathbf{P}^T\mathbf{P}]_{jj}+\alpha L_{ii}^S\right\}(S_{ij}-S_{ij}^t)^2 \tag{44}$$

Since $\begin{cases}[\mathbf{S}\mathbf{P}^T\mathbf{P}]_{ij}=\sum_{r=1}^{k}S_{ir}^t[\mathbf{P}^T\mathbf{P}]_{rj}\geq S_{ij}^t[\mathbf{P}^T\mathbf{P}]_{jj}\\ \alpha[\mathbf{D}^S\mathbf{S}]_{ij}=\alpha\sum_{r=1}^{n}D_{ir}^SS_{rj}^t\geq\alpha D_{ii}^SS_{ij}^t\geq\alpha(D_{ii}^S-W_{ii}^S)S_{ij}^t=\alpha L_{ii}^SS_{ij}^t\end{cases}$,
we have $\frac{[\mathbf{S}\mathbf{P}^T\mathbf{P}+\alpha\mathbf{D}^S\mathbf{S}]_{ij}}{S_{ij}^{(t)}}\geq[\mathbf{P}^T\mathbf{P}]_{jj}+\alpha L_{ii}^S$, so that $M(S_{ij},S_{ij}^t)\geq N(S_{ij})$.

According to the simultaneous Eqs. (42) and (43), we know $M(S_{ij}^{t+1},S_{ij}^t)$ is the local minimum of (43) and S_{ij}^{t+1} is the corresponding local minimum point:

$$S_{ij}^{t+1}=S_{ij}^t-\frac{S_{ij}^tN'_{ij}(S_{ij}^t)}{2[\mathbf{S}\mathbf{P}^T\mathbf{P}+\alpha\mathbf{D}^S\mathbf{S}]_{ij}}=S_{ij}^t\frac{[\mathbf{X}^T\mathbf{P}+\alpha\mathbf{W}^S\mathbf{S}]_{ij}}{[\mathbf{S}\mathbf{P}^T\mathbf{P}+\alpha\mathbf{D}^S\mathbf{S}]_{ij}} \tag{45}$$

Since (43) is the auxiliary function for N_{ij}, so N_{ij}decreases monotonically using the formula (18).

We know Lemma 2 from [15]:

Lemma 2. For any non-zero vector $\mathbf{a}\in\Re^m$, $\mathbf{b}\in\Re^m$,

$$\|\mathbf{a}\|_2-\frac{\|\mathbf{a}\|_2^2}{2\|\mathbf{b}\|_2}\leq\|\mathbf{b}\|_2-\frac{\|\mathbf{b}\|_2^2}{2\|\mathbf{b}\|_2} \tag{46}$$

Proof. From Lemma 2, we have:

$$\frac{\|\mathbf{P}_i^{t+1}\|_2^2}{2\|\mathbf{P}_i^t\|_2}-\|\mathbf{P}_i^{t+1}\|_2\geq\frac{\|\mathbf{P}_i^t\|_2^2}{2\|\mathbf{P}_i^t\|_2}-\|\mathbf{P}_i^t\|_2 \tag{47}$$

In the ith generation, we fix $\boldsymbol{V}$ as $\boldsymbol{V}^t$ to solve $\boldsymbol{S}^{t+1}$ and $\boldsymbol{P}^{t+1}$. We have:

$$\begin{aligned}&-2Tr\left(\mathbf{X}^T\mathbf{P}^{t+1}(\mathbf{S}^{t+1})^T\right)+Tr\left(\mathbf{S}^{t+1}(\mathbf{P}^{t+1})^T\mathbf{P}^{t+1}(\mathbf{S}^{t+1})^T\right)\\&+\alpha Tr\left((\mathbf{S}^{t+1})^T\mathbf{L}^S\mathbf{S}^{t+1}\right)+\beta Tr\left((\mathbf{P}^{t+1})^T\mathbf{L}^P\mathbf{P}^{t+1}\right)\\&+\theta Tr\left((\mathbf{P}^{t+1})^T\mathbf{V}^t\mathbf{P}^{t+1}\right)\leq-2Tr\left(\mathbf{X}^T\mathbf{P}^t(\mathbf{S}^t)^T\right)\\&+Tr\left(\mathbf{S}^t(\mathbf{P}^t)^T\mathbf{P}^t(\mathbf{S}^t)^T\right)+\alpha Tr\left((\mathbf{S}^t)^T\mathbf{L}^S\mathbf{S}^t\right)\\&+\beta Tr\left((\mathbf{P}^t)^T\mathbf{L}^P\mathbf{P}^t\right)+\theta Tr\left((\mathbf{P}^t)^T\mathbf{V}^t\mathbf{P}^t\right)\end{aligned} \tag{48}$$

Since $\|\mathbf{P}\|_{2,1}=\sum_{i=1}^{m}\|\mathbf{P}_i\|_2$, we can rewrite (48) as follows:

$$\begin{aligned}&-2Tr\left(\mathbf{X}^T\mathbf{P}^{t+1}(\mathbf{S}^{t+1})^T\right)+Tr\left(\mathbf{S}^{t+1}(\mathbf{P}^{t+1})^T\mathbf{P}^{t+1}(\mathbf{S}^{t+1})^T\right)\\&+\alpha Tr\left((\mathbf{S}^{t+1})^T\mathbf{L}^S\mathbf{S}^{t+1}\right)+\beta Tr\left((\mathbf{P}^{t+1})^T\mathbf{L}^P\mathbf{P}^{t+1}\right)+\theta\|\mathbf{P}^{t+1}\|_{2,1}\\&+\theta\sum_{i=1}^{m}\left(\frac{\|\mathbf{P}_i^{t+1}\|_2^2}{2\|\mathbf{P}_i^t\|_2}-\|\mathbf{P}_i^{t+1}\|_2\right)\leq-2Tr\left(\mathbf{X}^T\mathbf{P}^t(\mathbf{S}^t)^T\right)\\&+Tr\left(\mathbf{S}^t(\mathbf{P}^t)^T\mathbf{P}^t(\mathbf{S}^t)^T\right)+\alpha Tr\left((\mathbf{S}^t)^T\mathbf{L}^S\mathbf{S}^t\right)\\&+\beta Tr\left((\mathbf{P}^t)^T\mathbf{L}^P\mathbf{P}^t\right)+\theta\|\mathbf{P}^t\|_{2,1}+\theta\sum_{i=1}^{m}\left(\frac{\|\mathbf{P}_i^t\|_2^2}{2\|\mathbf{P}_i^t\|_2}-\|\mathbf{P}_i^t\|_2\right)\end{aligned} \tag{49}$$

Combining (47) with (49), we can get the following in equation:

$$\begin{aligned}&-2Tr\left(\mathbf{X}^T\mathbf{P}^{t+1}(\mathbf{S}^{t+1})^T\right)+Tr\left(\mathbf{S}^{t+1}(\mathbf{P}^{t+1})^T\mathbf{P}^{t+1}(\mathbf{S}^{t+1})^T\right)\\&+\alpha Tr\left((\mathbf{S}^{t+1})^T\mathbf{L}^S\mathbf{S}^{t+1}\right)+\beta Tr\left((\mathbf{P}^{t+1})^T\mathbf{L}^P\mathbf{P}^{t+1}\right)+\theta\|\mathbf{P}^{t+1}\|_{2,1}\\&\leq-2Tr\left(\mathbf{X}^T\mathbf{P}^t(\mathbf{S}^t)^T\right)+Tr\left(\mathbf{S}^t(\mathbf{P}^t)^T\mathbf{P}^t(\mathbf{S}^t)^T\right)\\&+\alpha Tr\left((\mathbf{S}^t)^T\mathbf{L}^S\mathbf{S}^t\right)+\beta Tr\left((\mathbf{P}^t)^T\mathbf{L}^P\mathbf{P}^t\right)+\theta\|\mathbf{P}^t\|_{2,1}\end{aligned} \tag{50}$$

In summary, based on the above convergence analysis, we can know the formula (6) decreases monotonically using the iterative updating rules (14) and (18).

2.5. Computational complexity analysis

In this section, we discuss the computational complexities of the proposed algorithm. Through the computational complexity analysis, we can intuitively see the computational efficiency of it. The common method to express the computational complexity is using big O notation [38].

Different from the standard NMF algorithm, a dual-graph model is adopted in DSNMF, which can make full use of the geometric information of the data space and the feature space to further excavate the potential information of the data, but the calculation of DSNMF is relatively complex. Assuming that the objective function converges after t iterations, the overall computational cost for NMF is $O(tmnc)$ [22]. Compared with NMF, the extra computational complexity of our algorithm is to construct the data graph and the feature graph. The computational cost for the data graph is $O(mn^2)$ and the computational cost for the feature graph is $O(m^2n)$, so the overall computational cost for DSNMF is $O(tmnc+mn^2+m^2n)$.

3. Experiments and analysis

In this section, we show the clustering experiments and the comparison of the clustering effects in different algorithms on 6 datasets. Our experiments mainly consist of two parts: the first part is the comparative experiments and analysis in our DSNMF, DSNMF-LDC and other 9 feature selection algorithms on 6 datasets. The second part is the comparative experiments and analysis in

Table 2
Test datasets.

Dataset	Dimensionality	Size	Class
Umist	644	575	20
Isolet	617	1560	26
ORL	1024	400	40
Sonar	60	208	2
BC	30	569	2
Dbworld_bodies	4702	64	2

our DSNMF-LDC and other 7 clustering algorithms on the dataset COIL20 [39].

3.1. Comparison with other feature selection algorithms

We carry out the comparative experiments on 6 datasets in our DSNMF, DSNMF-LDC and other 9 feature selection algorithms that are all features selected algorithms and other 7 feature selection algorithms mentioned in the introduction part: LapScore [12], SPEC [13], MCFS [14], MRSF [15], JELSR [16], LSPE [17], CSPCA [18] and DFSC [1].

3.1.1. Datasets

We compare the clustering results in aforementioned 10 feature selection algorithms on the following 6 datasets. The test datasets used are similar to those in [1], which is shown in Table 2.

3.1.2. Evaluation metrics

In the experiment, we take the datasets categories as the clusters. Usually, the clustering effects are represented by comparing the clustering labels with the ground truth labels. Two evaluation metrics are used to measure the clustering effects, that are clustering accuracy (ACC) [22,38] and normalized mutual information (NMI) [22,40,41].

3.1.3. Parameter settings

We compare our DSNMF and DSNMF-LDC with other 9 feature selection algorithms on 6 datasets in Table 2 using clustering ACC and NMI. In DFSC, LSPE, LapScore, SPEC and MCFS, the neighbor number k is selected from {3, 5, 10, 15}, the bandwidth σ of heat kernel weighting is selected from $\{1, 10^3, 10^5\}$. In DFSC, the regular parameter α is selected from {0.01, 0.1, 0.5, 1.0, 5.0, 9.0, 13.0, 17.0}, β is selected from {10, 100, 1000}, λ is selected from {300, 800, 2000, 4000, 6000, 8000}. The parameter settings in DSNMF and DSNMF-LDC are similar to those in DFSC, the regular parameter α is selected from {0.01, 0.1, 0.5, 1.0, 5.0, 9.0, 13.0, 17.0}, β and θ are selected from {300, 800, 2000, 4000, 6000, 8000}, μ is selected from $\{10^{-8}, 10^{-6}, 10^{-4}, 10^{-2}, 10^{\circ}, 10^2, 10^4, 10^6, 10^8\}$.

We adjust these parameters to maximize clustering ACC and NMI of the algorithms, so we may select different features on different datasets. For the sake of fairness, the same clustering algorithm K-means is used after feature selection LapScore, SPEC, MCFS, MRSF, JELSR, LSPE, DFSC and DSNMF. We carry out the experiment 100 times, and calculate the means respectively.

3.1.4. Experimental results and analysis

Tables 3 and 4 show clustering ACC and NMI in DSNMF, DSNMF-LDC and other 9 feature selection algorithms on 6 datasets, including the means (MEAN) and standard deviation (STD) of 100 times tests. We bold mark the best result in each dataset.

We can see in Table 3 that DSNMF can achieve almost all the best results in clustering ACC except that it is inferior to JELSR and LSPE on the dataset Sonar. On all the datasets, DSNMF-LDC can achieve the best clustering ACC, and it can reach more than 63%.

We can see in Table 4 that DSNMF and DSNMF-LDC can achieve almost all the best results in clustering NMI except that they are inferior to LSPE on the dataset Dbworld_bodies. DSNMF and DSNMF-LDC can achieve good clustering results and clustering NMI can reach more than 67% except on the datasets Sonar and BC.

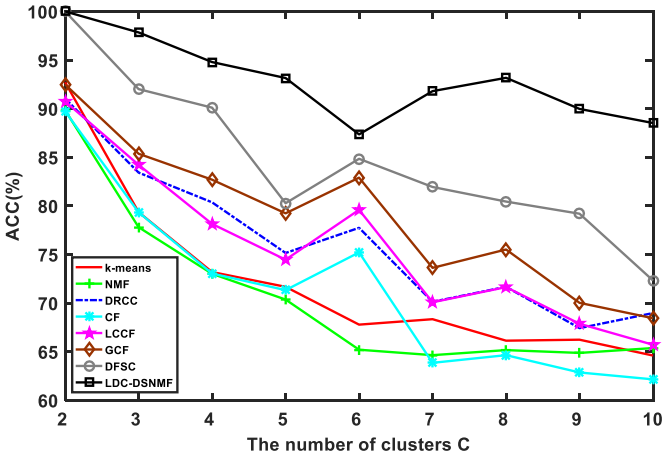


Fig. 1. Clustering ACC on the dataset COIL20.

Compared with the recently proposed DFSC, our DSNMF and DSNMF-LDC are superior to DFSC in clustering ACC and NMI on all the datasets. Moreover, in view of STDs of clustering ACC and NMI, they are smaller in DSNMF-LDC than in DFSC on all the datasets, so DSNMF-LDC is relatively stable and robust. It is mainly due to the non-negative matrix factorization in DSNMF, so that it can make the two non-negative matrix factors of the data space and the feature space update iteratively and interactively, which can give full play to the dual-graph model. In addition, the local discriminative feature selection clustering added after DSNMF can greatly improve the clustering effectiveness and robustness.

3.2. Comparison with other clustering algorithms

We carry out the comparative experiments on the dataset COIL20 in our DSNMF-LDC and other 7 clustering algorithms that are K-means and other 6 clustering algorithms mentioned in the introduction part: NMF [22,23], DRCC [24], CF [25], LCCF [27], GCF [28] and DFSC [1].

3.2.1. Parameter settings

In the clustering algorithms LCCF, DRCC, GCF, DFSC and DSNMF-LDC, we choose the binary (0–1) weighting to construct the neighbor graph and set the neighbor number $k = 5$. In LCCF, DRCC and GCF, we set $\lambda = \mu = 100$. In DFSC, we set $\alpha = 100$, $\lambda = 10^8$, β is selected from $\{10^{-1}, 1, 10, 10^2\}$. For a fair comparison, we set $\alpha = \beta = 100$, $\theta = 10^8$ in DSNMF-LDC.

3.2.2. Experimental results and analysis

The dataset COIL20 contains 1440 images of 20 objects viewed from different angles and each image is scaled to 32×32 pixel which represented by a 1024-dimensional vector. We carry out the comparative experiments on the dataset COIL20 in our DSNMF-LDC and other clustering algorithms including K-means, matrix factorization clustering algorithms (NMF, CF, LCCF), co-clustering algorithms (DRCC, GCF) and feature selection clustering algorithm (DFSC). For each particular clustering number c, we carry out the experiment 20 times, and calculate the means, respectively. The clustering ACC and NMI on COIL20 in different algorithms are shown in Tables 5 and 6, as well as shown in Figs. 1 and 2.

As we can see from Table 5, overall the clustering ACC shows a downward trend in the increasing clustering number c. The clustering effects of K-means, NMF and CF are not ideal. LCCF is slightly

Table 3
Clustering ACC in DSNMF, DSNMF-LDC and other 9 feature selection algorithms on 6 datasets (MEAN ± STD%).

Algorithms	Umist	Isolet	ORL	Sonar	BC	Dbworld_bodies
All features	44.23 ± 1.02	50.58 ± 0.85	50.00 ± 0.43	54.32 ± 1.20	72.27 ± 0.20	73.81 ± 0.00
LapScore	37.30 ± 0.93	48.79 ± 0.56	44.50 ± 0.73	58.80 ± 1.14	70.17 ± 0.36	73.47 ± 1.16
SPEC	42.56 ± 1.20	49.50 ± 0.63	49.88 ± 0.23	61.00 ± 1.26	74.00 ± 0.23	77.94 ± 1.85
MCFS	46.55 ± 1.00	54.48 ± 0.84	49.40 ± 0.93	54.20 ± 0.84	71.00 ± 0.58	91.13 ± 1.04
MRSF	48.38 ± 1.05	50.80 ± 0.69	49.78 ± 0.69	60.33 ± 1.40	72.79 ± 0.22	85.02 ± 1.59
JELSR	48.90 ± 1.03	55.08 ± 0.45	50.02 ± 0.56	64.20 ± 0.94	74.20 ± 0.30	90.63 ± 0.00
LSPE	49.26 ± 1.12	56.11 ± 0.63	50.25 ± 0.80	66.25 ± 1.67	75.86 ± 0.24	**93.75 ± 0.00**
CSPCA	40.00 ± 0.95	55.85 ± 1.65	51.90 ± 2.21	58.37 ± 2.53	83.69 ± 1.91	77.29 ± 1.80
DFSC	50.12 ± 2.79	60.14 ± 3.51	51.71 ± 2.61	58.57 ± 2.31	85.41 ± 0.00	91.75 ± 1.09
DSNMF	53.39 ± 2.71	63.56 ± 2.22	57.63 ± 1.83	63.70 ± 0.34	85.41 ± 0.00	92.97 ± 1.10
DSNMF-LDC	**79.48 ± 0.25**	**63.97 ± 0.06**	**64.25 ± 0.71**	**72.60 ± 0.00**	**88.75 ± 0.00**	**93.75 ± 0.00**

Table 4
Clustering NMI in DSNMF, DSNMF-LDC and other 9 feature selection algorithms on 6 datasets (MEAN ± STD%).

Algorithms	Umist	Isolet	ORL	Sonar	BC	Dbworld_bodies
All features	60.30 ± 1.45	73.02 ± 0.92	70.36 ± 1.17	0.88 ± 0.00	17.61 ± 0.00	24.00 ± 0.00
LapScore	56.32 ± 1.52	66.80 ± 1.20	67.80 ± 1.76	1.68 ± 0.00	16.79 ± 0.00	23.82 ± 1.01
SPEC	57.04 ± 1.24	66.90 ± 1.49	70.26 ± 1.65	5.97 ± 0.42	18.83 ± 0.00	25.20 ± 1.62
MCFS	69.20 ± 1.31	70.43 ± 1.93	70.98 ± 1.78	1.87 ± 2.85	17.32 ± 0.00	67.88 ± 1.62
MRSF	66.67 ± 1.43	68.35 ± 1.67	70.50 ± 1.81	2.96 ± 1.04	17.32 ± 0.00	56.79 ± 2.39
JELSR	70.18 ± 1.64	70.50 ± 1.34	70.20 ± 1.72	6.24 ± 0.00	18.86 ± 0.00	54.89 ± 0.00
LSPE	70.91 ± 1.50	71.01 ± 1.85	71.04 ± 1.11	7.24 ± 0.38	18.83 ± 0.00	**68.09 ± 0.00**
CSPCA	63.26 ± 1.09	71.44 ± 0.46	71.85 ± 1.22	3.39 ± 1.52	38.39 ± 1.32	23.08 ± 2.18
DFSC	65.85 ± 1.76	73.98 ± 1.33	73.27 ± 1.25	2.22 ± 1.03	42.23 ± 0.00	58.93 ± 3.67
DSNMF	70.02 ± 0.82	77.11 ± 0.35	75.73 ± 0.97	8.48 ± 0.00	42.23 ± 0.00	65.63 ± 3.49
DSNMF-LDC	**90.39 ± 0.57**	**78.03 ± 0.03**	**77.71 ± 0.20**	**15.01 ± 0.00**	**47.39 ± 0.00**	67.09 ± 0.00

Table 5
Clustering ACC on the datasets COIL20.

c	2	3	4	5	6	7	8	9	10	AVG	STD
K-means	92.71	79.35	73.19	71.67	67.78	68.34	66.13	66.23	64.60	72.22	8.93
NMF	89.84	77.80	73.01	70.36	65.20	64.64	65.16	64.87	65.37	70.69	8.53
DRCC	91.04	83.42	80.36	75.15	77.74	70.13	71.67	67.42	68.97	76.21	7.74
CF	89.72	79.34	73.04	71.33	75.21	63.85	64.64	62.86	62.15	71.34	9.20
LCCF	90.74	84.22	78.14	74.46	79.59	70.08	71.64	67.87	65.71	75.82	8.14
GCF	92.48	85.36	82.69	79.23	82.90	73.62	75.51	70.02	68.44	78.91	7.78
DFSC	**100.00**	92.01	90.10	80.27	84.84	81.94	80.44	79.19	72.32	84.56	8.26
DSNMF-LDC	**100.00**	**97.84**	**94.79**	**93.15**	**87.35**	**91.80**	**93.17**	**89.97**	**88.52**	**92.95**	**4.16**

Table 6
Clustering NMI on the datasets COIL20.

c	2	3	4	5	6	7	8	9	10	AVG	STD
K-means	79.64	66.11	67.56	68.95	71.51	72.17	71.32	72.39	70.57	71.13	3.85
NMF	71.25	63.42	67.87	66.07	68.34	70.14	70.40	71.65	71.89	69.00	2.85
DRCC	77.29	74.57	75.14	72.26	72.86	73.42	73.89	70.38	69.40	73.25	2.40
CF	71.13	63.21	66.38	67.67	65.33	66.67	67.28	66.40	66.27	66.70	2.10
LCCF	74.51	68.69	70.63	72.22	68.81	70.57	70.67	69.86	68.69	70.52	**1.90**
GCF	80.40	76.35	77.43	78.56	74.89	75.31	76.45	72.71	70.63	75.86	2.95
DFSC	**100.00**	90.97	93.97	82.77	86.09	83.28	84.91	74.17	76.43	85.84	8.17
DSNMF-LDC	**100.00**	**94.25**	**94.78**	**89.68**	**86.45**	**89.07**	**92.46**	**91.27**	**90.14**	**92.01**	3.96

better than them in most of the clustering number *c*. DRCC and GCF have the higher clustering ACC than aforementioned four algorithms. DFSC has the higher clustering ACC than aforementioned six algorithms. Our DSNMF-LDC is the most effective algorithm. Comparing to the best algorithm other than our proposed DSNMF-LDC algorithms, i.e., DFSC, DSNMF-LDC achieves 8.39 percent improvement in accuracy.

Similar to Table 5 of the clustering ACC, we can see from the clustering NMI in the Table 6, overall it shows a downward trend in the increasing clustering number *c*. Comparing to the best algorithm other than our proposed DSNMF-LDC algorithms, i.e., DFSC, DSNMF-LDC achieves 6.17% improvement in normalized mutual information. The clustering effects of K-means, NMF and CF are not ideal. LCCF is slightly better than them in most of the clustering number *c*, due to the local geometric information used in LCCF. DRCC and GCF have the higher clustering NMI than aforementioned four algorithms, because DRCC and GCF can make full use of the local geometric structure and preserve the manifold information in the data space and the feature space at the same time. DFSC has the higher clustering NMI than aforementioned six algorithms, because DFSC eliminates the redundant and irrelevant features, and then selects the more discriminative and effective features from the original features using feature selection. Our DSNMF-LDC is the most effective algorithm, because DSNMF-LDC synthesizes the advantages of the previous algorithms. DSNMF-LDC adopts the non-negative matrix factorization, so the two non-negative matrix factors of the data space and the feature space can update iteratively and interactively, which can give full play to the

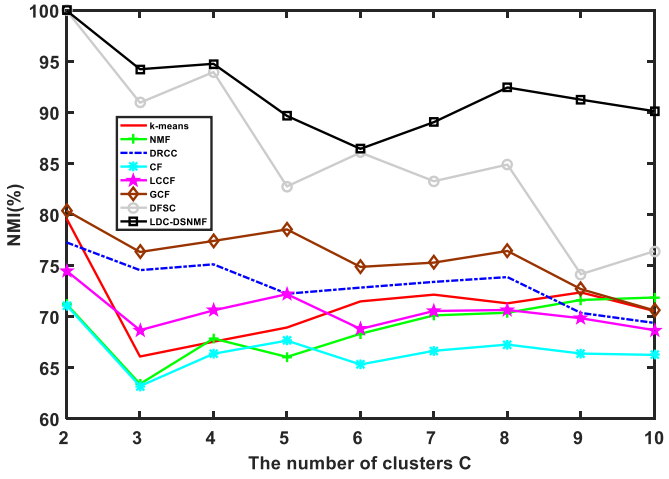


Fig. 2. Clustering NMI on the dataset COIL20.

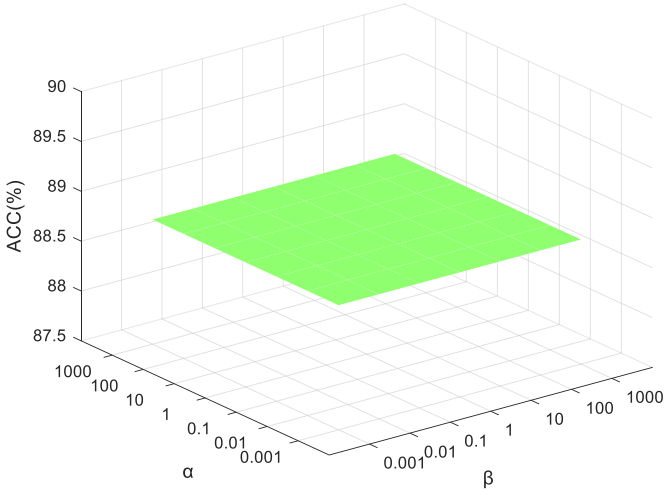


Fig. 3. Clustering ACC with different α and β on BC.

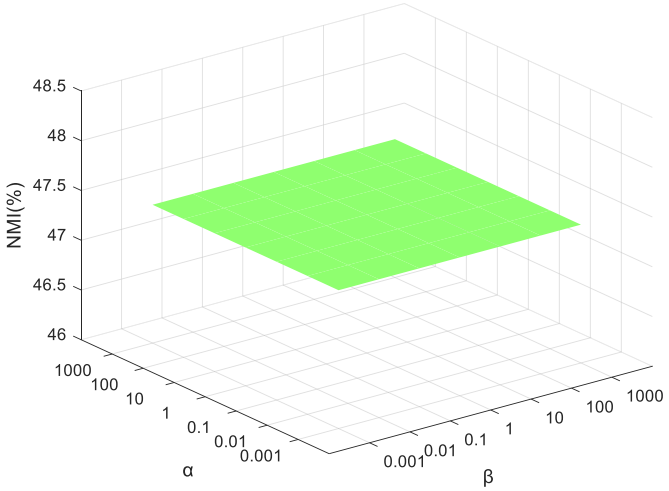


Fig. 4. Clustering NMI with different α and β on BC.

dual-graph model. In addition, DSNMF-LDC imposes the $L_{2,1}$-*norm* constraint on the non-negative matrix factor $\boldsymbol{P}$ of the feature space, which can make full use of the sparse self-representation information. Therefore, DSNMF-LDC can not only ensure the sparsity, but also select the more effective discriminative features to enhance the robustness to noise. What's more, the local discriminative feature selection clustering is added in DSNMF-LDC, so it is more discriminative and has better clustering results than other algorithms.

In order to show the clustering effectiveness of each clustering algorithm more vividly, we draw Figs. 1 and 2 according to Tables 5 and 6.

As we can vividly see in Figs. 1 and 2, the clustering ACC and NMI show a downward trend in the increasing clustering number c. In addition, our DSNMF-LDC can achieve better clustering results than other clustering algorithms in all clustering number c.

3.3. Parameter sensitivity analysis

There are some parameters in our DSNMF-LDC, such as the heat kernel bandwidth parameter σ, the neighbor number k, the number of selected features q, the sparse self-representation parameter θ, the regular parameters α, β and μ.

In the first experiment, we analyze the sensitivity of the regular parameter α and β on three representative datasets i.e., BC, Isolet and Dbworld_bodies, since they are very different in the data dimensionality. We fix other parameters, and then set α and β selected from $\{10^{-3}, 10^{-2}, 10^{-1}, 10^{\circ}, 10^{1}, 10^{2}, 10^{3}\}$. For each parameter setting, we carry out the experiment 20 times, calculate the means respectively and plot the three-dimensional map in Figs. 3–8.

From Figs. 3 and 4, we can see that the clustering results of DSNMF-LDC on BC are not sensitive to the regular parameters α, β. ACC is always equal to 88.75% and NMI is always equal to 47.39%.

From Figs. 5 and 6, we can see that the clustering results of DSNMF-LDC are not sensitive to the regular parameters α, β. ACC = 63.97% is the maximum when $\alpha = 1000$, $\beta = 0.01$. ACC = 62.76% is the minimum when $\alpha = 10$, $\beta = 100$. NMI = 78.03% is the maximum when $\alpha = 0.1$, $\beta = 100$. NMI = 77.58% is the minimum when $\alpha = 0.001$, $\beta = 0.001$.

From Figs. 7 and 8, we can see that the clustering results of DSNMF-LDC on BC are not sensitive to the regular parameters α, β. ACC is always equal to 93.75% and NMI is always equal to 67.09%.

Therefore, from these sensitivity experiments, we can make a conclusion that the clustering results of DSNMF-LDC are not sensitive to the regular parameters α, β, and the algorithm has the strong robustness to these two parameters.

In the second experiment, we analyze the sensitivity of the regular parameter μ on the same three datasets. We fix other parameters, and then set μ selected from $\{10^{-8}, 10^{-6}, 10^{-4}, 10^{-2}, 10^{\circ}, 10^{2}, 10^{4}, 10^{6}, 10^{8}\}$. For each parameter setting, we carry out the experiment 20 times, calculate the means respectively and plot them in Figs. 9–11.

From Figs. 9 and 10, we can see that the clustering results of DSNMF-LDC on BC and Isolet are not sensitive to the regular parameters μ, but it is different in Fig. 11. Therefore, we can conclude that the clustering results of DSNMF-LDC are with different sensitivities on the different datasets to the regular parameters μ.

3.4. Effective analysis

We use the dataset Ionosphere like DFSC the effectiveness of DSNMF. There are 351 samples with 34 features in the original Ionosphere. We artificially generate 66 features with the random linear combination of the original 34 features to form a dataset with 100 features where the first 34 features are the original features.

We calculate the non-negative matrix factor $\boldsymbol{P}$ of the data space on the new generated dataset in our DSNMF, which is the coefficient matrix. We calculate $||\mathbf{P}_i||_2$ as different colors and 100 features as the coordinates in Fig. 12.

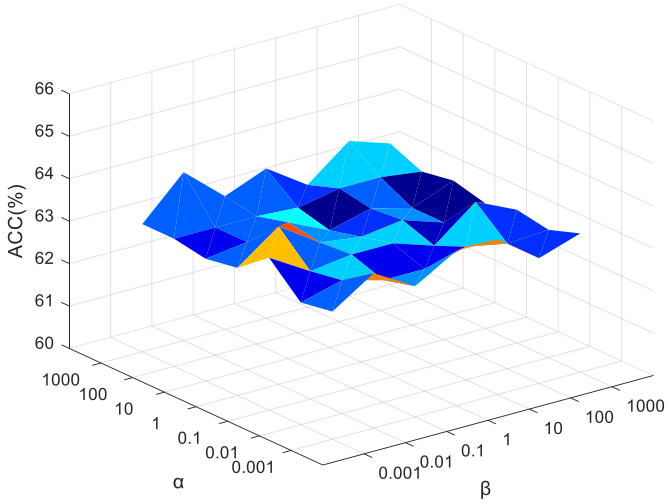


Fig. 5. Clustering ACC with different α and β on Isolet.

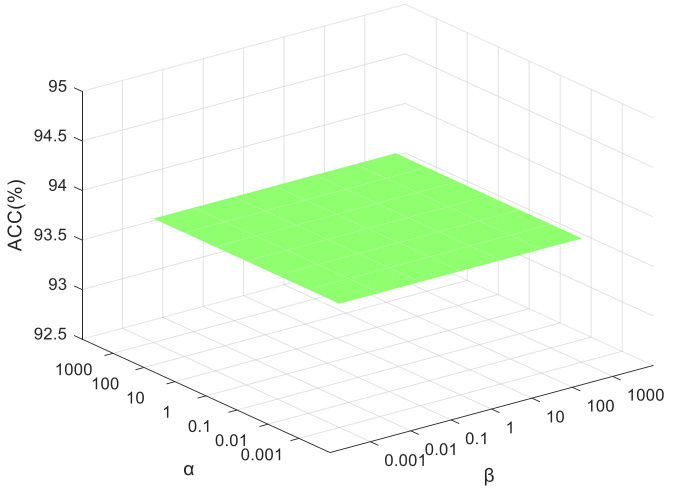


Fig. 7. Clustering ACC with different α and β on Dbworld_bodies.

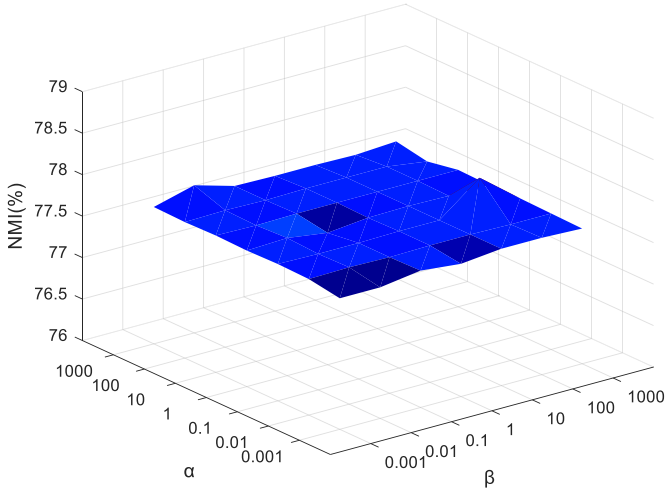


Fig. 6. Clustering NMI with different α and β on Isolet.

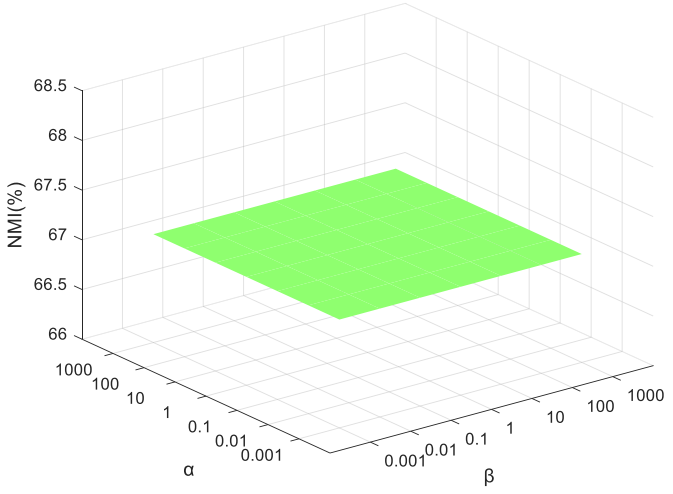


Fig. 8. Clustering NMI with different α and β on Dbworld_bodies.

From Fig. 12, we can see that the coefficients of the first 34 features are obviously larger than the coefficients of the 66 random generated features. The experiments strongly illustrate the effectiveness of our DSNMF in the feature selection.

3.5. Computational time analysis

We show the computational time experiments in feature selection algorithms with different computational complexities, i.e., SPEC [13], MCFS [14], JELSR [16], CSPCA [18], DFSC [1]and DSNMF. With a naive MATLAB R2016a implementation, the calculations are made on an Intel(R) Core(TM) i5-2450 M CPU @ 2.5 GHz Windows machine with a solid state disk (SSD) of 540 MB/s reading speed and 520 MB/s writing speed. To show the influence on the data dimensionality and data size, we select two representative datasets, i.e., Dbworld_bodies and Isolet, since they have the largest data dimensionality and the largest data size among six datasets, respectively. We select $q = 100$, $q = 300$ and $q = 500$ features of Dbworld_bodies and $q = 10$, $q = 30$ and $q = 50$ features of Isolet. For the sake of fairness, we carry out each experiment 10 times, and calculate the means of them in Tables 7 and 8.

From Tables 7 and 8, we can see that SPEC costs the least time in all the situations. For DSNMF, from the computational complexity analysis, DSNMF uses a dual-graph model and the computational cost is relatively large, but from the experimental results, the running time of DSNMF is less than many algorithms i.e., JELSR and DSFC, mainly because DSNMF converges faster and the number of iterations is smaller. In addition, the running time of the algorithm is affected by many factors, such as data dimensionality, data size, the number of selected features, etc.

Table 7
Computational time of different methods on Dbworld_bodies with different numbers of selected features.

q	SPEC	MCFS	JELSR	DSFC	CSPCA	DSNMF
100	0.067	1.248	128.226	503.470	2480.869	3.978
300	0.069	2.995	138.258	511.110	2570.522	4.021
500	0.072	6.068	158.184	518.755	5644.162	4.194

Table 8
Computational time of different methods on isolet with different numbers of selected features.

q	SPEC	MCFS	JELSR	DSFC	CSPCA	DSNMF
10	0.202	1.843	9.700	5.809	31.408	3.499
30	0.232	2.023	9.839	5.675	31.812	3.366
50	0.249	2.363	10.072	5.782	32.881	3.379

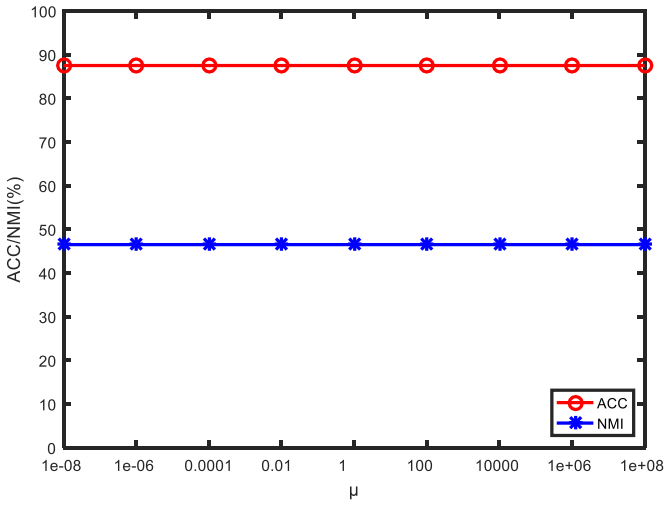


Fig. 9. ACC and NMI with different μ on BC.

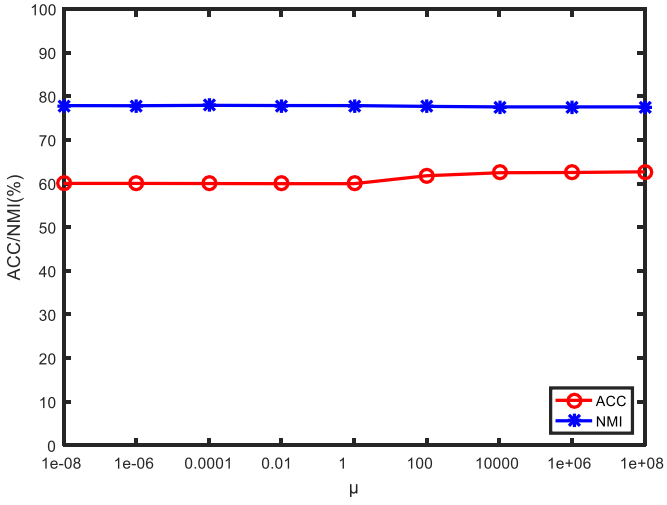


Fig. 10. ACC and NMI with different μ on Isolet.

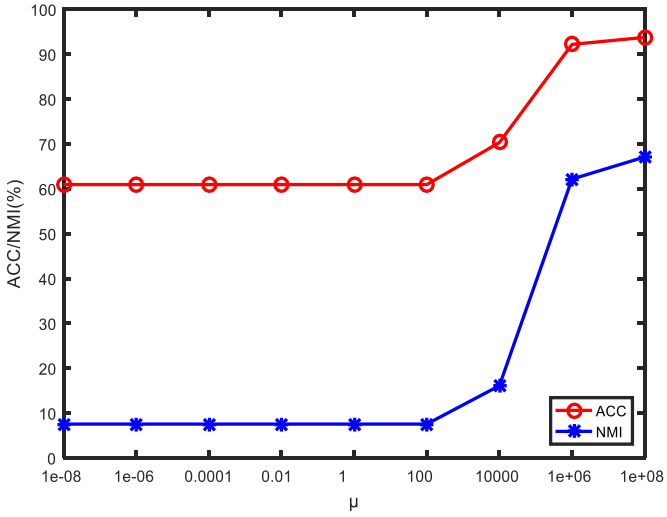


Fig. 11. ACC and NMI with different μ on Dbworld_bodies.

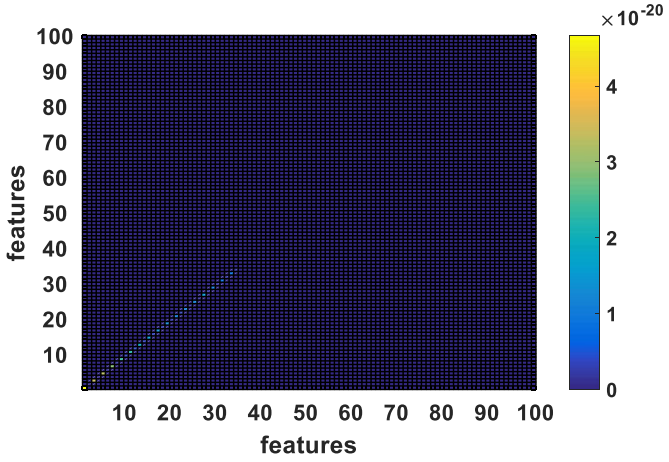


Fig. 12. Effective metric $||\mathbf{P}_i||_2$ of different features.

4. Conclusion

A new feature selection clustering algorithm is proposed in this paper, namely feature selection based dual-graph sparse non-negative matrix factorization for local discriminative clustering (DSNMF-LDC). In recent years, it is found that the data information distributes in the nonlinear low-dimensional submanifold of the high-dimensional space, namely the data manifold, and the data feature information distributes also in a low-dimensional submanifold, namely the feature manifold. Therefore, DSNMF preserves the manifold information in the data space and the feature space at the same time using the dual-graph model. DSNMF not only has the advantages of the dual-graph model in the co-clustering algorithms, but also utilizes the feature selection in advance. Therefore, it can make the two non-negative matrix factors of the data space and the feature space update iteratively and interactively, which can give full play to the dual-graph model. In addition, DSNMF-LDC imposes the sparse constraint on the non-negative matrix factor $\boldsymbol{P}$ of the data space, namely sparse self-representation information. We select the former q features according to the value of $||\mathbf{P}_i||_2$ in a descending order which can reduce the data dimensions, be robust to the noises and have better clustering results. What's more, DSNMF-LDC adds the local discriminative feature selection clustering into DSNMF, which can significantly improve the clustering effectiveness and robustness.

However, from the experiments and analysis, the feature selection DSNMF cannot achieve the best results on all the datasets, such as the dataset Dbworld_bodies. Therefore, the future work will be placed on the optimization of feature selection, such as the composition method and the similarity evaluation and so on.

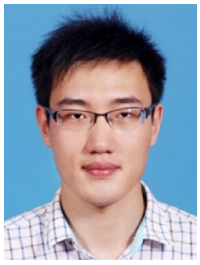
Acknowledgment

We would like to express our sincere appreciation to the editors and the anonymous reviewers for their insightful comments, which have greatly helped us in improving the quality of the paper. This work was partially supported by the National Natural Science Foundation of China, under Grant 61371201, the National Basic Research Program (973 Program) of China under Grant 2013CB329402, the National Science Foundation of China under Grant 91438103, Grant 91438201.

References

[1] R. Shang, Z. Zhang, L. Jiao, et al., Self-representation based dual-graph regularized feature selection clustering, Neurocomputing 171 (2016) 1242–1253.

[2] R. Shang, W. Wang, R. Stolkin, et al., Nonnegative spectral learning and sparse regression based dual-graph regularized feature selection, in: Proceedings of the IEEE Transactions on Cybernetics, 2017, doi:10.1109/TCYB.2017.2657007.
[3] Z. Zhao, G. Feng, J. Zhu, et al., Manifold learning: dimensionality reduction and high dimensional data reconstruction via dictionary learning, Neurocomputing 216 (2016) 268–285.
[4] S. Yang, P. Jin, B. Li, et al., Semisupervised dual-geometric subspace projection for dimensionality reduction of hyperspectral image data, IEEE Trans. Geosci. Remote Sens. 52 (6) (2014) 3587–3593.
[5] F. Nie, H. Huang, X. Cai, et al., Efficient and robust feature selection via joint L2,1-norms minimization, in: Proceedings of the Advances in neural information processing systems, 2010, pp. 1813–1821.
[6] B. Gu, V. Sheng, K. Tay, et al., Incremental support vector learning for ordinal regression, IEEE Trans. Neural Netw. Learn. Syst. 26 (7) (2015) 1403–1416.
[7] H. Ogura, H. Amano, M. Kondo, Comparison of metrics for feature selection in imbalanced text classification, Expert Syst. Appl. 38 (5) (2011) 4978–4989.
[8] Y. Saeys, I. Inza, P. Larrañaga, A review of feature selection techniques in bioinformatics, Bioinformatics 23 (19) (2007) 2507–2517.
[9] R. Cai, Z. Hao, X. Yang, et al., An efficient gene selection algorithm based on mutual information, Neurocomputing 72 (4) (2009) 991–999.
[10] Q. Zhu, L. Lin, M.L. Shyu, et al., Feature selection using correlation and reliability based scoring metric for video semantic detection, in: Proceedings of the IEEE Sixth International Conference on Semantic Computing, 2010, pp. 462–469.
[11] F. Amiri, M.M.R. Yousefi, C. Lucas, et al., Mutual information-based feature selection for intrusion detection systems, J. Netw. Comput. Appl. 34 (4) (2011) 1184–1199.
[12] X. He, D. Cai, P. Niyogi, Laplacian score for feature selection, Adv. Neural Inf. Process. Syst. 18 (2005) 507–514.
[13] Z. Zhao, H. Liu, Spectral feature selection for supervised and unsupervised learning, in: Proceedings of the Twenty Fourth International Conference on Machine Learning, 2007, pp. 1151–1157.
[14] D. Cai, C. Zhang, X. He, Unsupervised feature selection for multi-cluster data, in: Proceedings of the Sixteenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2010, pp. 333–342.
[15] Z. Zhao, L. Wang, H. Liu, Efficient spectral feature selection with minimum redundancy, in: Proceedings of the Twenty Fourth AAAI Conference on Artificial Intelligence (AAAI), 2010.
[16] C. Hou, F. Nie, X. Li, et al., Joint embedding learning and sparse regression: a framework for unsupervised feature selection, IEEE Trans. Cybern. 44 (6) (2014) 793–804.
[17] X. Fang, Y. Xu, X. Li, et al., Locality and similarity preserving embedding for feature selection, Neurocomputing 128 (5) (2013) 304–315.
[18] X. Chang, F. Nie, Y. Yang, et al., Convex sparse PCA for unsupervised feature learning, ACM Trans. Knowl. Discov. Data (TKDD) 11 (1) (2016) 1–16 3.
[19] B. Gu, V. Sheng, A robust regularization path algorithm for ν-support vector classification, in: Proceedings of the IEEE Transactions on Neural Networks and Learning Systems, 2016, doi:10.1109/TNNLS.2016.2527796.
[20] F. Nie, Z. Zeng, I.W. Tsang, et al., Spectral embedded clustering: a framework for in-sample and out-of-sample spectral clustering, IEEE Trans. Neural Netw 22 (11) (2011) 1796–1808.
[21] X. Yan, Y. Zhu, W. Zou, et al., A new approach for data clustering using hybrid artificial bee colony algorithm, Neurocomputing 97 (1) (2012) 241–250.
[22] D. Cai, X. He, J. Han, et al., Graph regularized nonnegative matrix factorization for data representation, IEEE Trans. Pattern Anal. Mach. Intell. 33 (8) (2010) 1548–1560.
[23] J. Huang, F. Nie, H. Huang, et al., Robust manifold nonnegative matrix factorization, ACM Trans. Knowl. Discov. Data (TKDD) 8 (3) (2014) 1–21 11.
[24] Q. Gu, J. Zhou, Co-clustering on manifolds, in: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2009, pp. 359–368.
[25] W. Xu, Y. Gong, Document clustering by concept factorization, in: Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, 2004, pp. 202–209.
[26] K.R. Müller, S. Mika, G. Rätsch, et al., An introduction to kernel-based learning algorithms, IEEE Trans. Neural Netw. 12 (2) (2001) 181–201.
[27] D. Cai, X. He, J. Han, Locally consistent concept factorization for document clustering, IEEE Trans. Knowl. Data Eng. 23 (6) (2010) 902–913.
[28] J. Ye, Z. Jin, Dual-graph regularized concept factorization for clustering, Neurocomputing 138 (11) (2014) 120–130.
[29] C. Ding, T. Li, Adaptive dimension reduction using discriminant analysis and k-means clustering, in: Proceedings of the Twenty Fourth International Conference on Machine Learning, 2007, pp. 521–528.
[30] F.D. l. Torre, T. Kanade, Discriminative cluster analysis, in: Proceedings of the Twenty Third International Conference on Machine Learning, 2006, pp. 241–248.
[31] R. Shang, Z. Zhang, L. Jiao, et al., Global discriminative-based nonnegative spectral clustering, Pattern Recognit. 55 (2016) 172–182.
[32] F. Shang, L. Jiao, F. Wang, Graph dual regularization non-negative matrix factorization for co-clustering, Pattern Recognit. 45 (6) (2012) 2237–2250.
[33] P. Li, J. Bu, C. Chen, et al., Relational multimanifoldcoclustering, IEEE Trans. Cybern. 43 (6) (2013) 1871–1881.
[34] C. Ding, D. Zhou, X. He, et al., R1-PCA: rotational invariantprincipal component analysis for robust subspace factorization, in: Proceedings of the Twenty Third International Conference on Machine Learning, 2006, pp. 281–288.
[35] Y. Yang, D. Xu, F. Nie, et al., Image clustering using local discriminant models and global integration, IEEE Trans. Image Process. 19 (10) (2010) 2761–2773.
[36] L.K. Saul, S.T. Roweis, Think globally, fit locally: unsupervised learning of low dimensional manifolds, J. Mach. Learn. Res. 4 (2) (2003) 119–155.
[37] J. Malik, J. Shi, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 888–905.
[38] T.H. Cormen, C.E. Leiserson, R.L. Rivest, et al., Introduction to Algorithms, MIT Press and McGraw-Hill, 2001.
[39] R. Shang, W. Wang, R. Stolkin, et al., Subspace learning-based graph regularized feature selection, Knowl. Based Syst. 112 (2016) 152–165.
[40] H. Liu, Z. Wu, D. Cai, et al., Constrained nonnegative matrix factorization for image representation, IEEE Trans. Softw. Eng. 34 (7) (2012) 1299–1311.
[41] D. Huang, J. Lai, C. Wang, Robust ensemble clustering using probability trajectories, IEEE Trans. Knowl. Data Eng. 28 (5) (2016) 1312–1326.



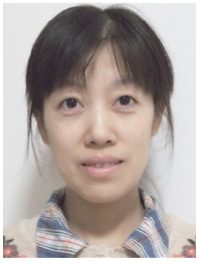
Yang Meng received the B.E. degree in electronic information engineering from Jiangsu University of Science and Technology, Zhenjiang, China. He is currently working toward master-doctor continuous study that the master's degree in electronic circuit and system and the doctor's degree in intelligent information processing from Xidian University, Xi'an, China. His current research interests include data mining and machine learning.

Ronghua Shang (M'09) received the B.S. degree in information and computation science and the Ph.D. degree in pattern recognition and intelligent systems from Xidian University in 2003 and 2008, respectively. She is currently a professor with Xidian University. Her current research interests include optimization problems, machine learning, image processing, and data mining.

Licheng Jiao (SM'89) received the B.S. degree from Shanghai Jiaotong University, Shanghai, China, in 1982, the M.S. and Ph.D. degrees from Xi'an Jiaotong University, Xi'an, China, in 1984 and 1990, respectively. From 1990 to 1991, he was a postdoctoral Fellow in the National Key Laboratory for Radar Signal Processing, Xidian University, Xi'an, China. Since 1992, Dr. Jiao has been a Professor in the School of Electronic Engineering at Xidian University. Currently, he is the Director of the Key Lab of Intelligent Perception and Image Understanding of Ministry of Education of China at Xidian University, Xi'an, China. Dr. Jiao is a Senior Member of IEEE, member of IEEE Xi'an Section Execution Committee and the Chairman of Awards and Recognition Committee, vice board chairperson of Chinese Association of Artificial Intelligence, councilor of Chinese Institute of Electronics, committee member of Chinese Committee of Neural Networks, and expert of Academic Degrees Committee of the State Council. His research interests include image processing, natural computation, machine learning, and intelligent information processing. He has charged of about 40 important scientific research projects, and published more than 20 monographs and a hundred papers in international journals and conferences.

Wenya Zhang received the B.E. degree in computer science and technology from Chongqing University of Post and Telecommunications, Chongqing, China. She is currently working toward computer technology from Xidian University, Xi'an, China. Her current research interests include data mining and machine learning.

Yijing Yuan received the B.E. degree in School of Computer Science and Control Engineering from North University of China, Taiyuan, China. She is now pursuing the M.S. degree in School of Electronic Engineering from Xidian University, Xi'an, China. Her current research interests include image processing and machine learning.



Shuyuan Yang received the B.S. degree in electrical engineering and the M.S. and Ph.D. degrees in circuit and system from Xidian University, Xi'an, China, in 2000, 2003, and 2005, respectively. She is currently a Post-Doctoral Fellow with the Institute of Intelligent Information Processing, Xidian University. Her main current research interests include intelligent signal processing, machine learning, and image processing.