International Journal of Machine Learning and Cybernetics
https://doi.org/10.1007/s13042-017-0760-y

ORIGINAL ARTICLE

Unsupervised feature selection based on self-representation sparse regression and local similarity preserving

Ronghua Shang[1] · Jiangwei Chang[1] · Licheng Jiao[1] · Yu Xue[2]

Received: 25 December 2016 / Accepted: 5 December 2017

Abstract
Feature selection, as an indispensable method of data preprocessing, has attracted the attention of researchers. In this paper, we propose a new feature selection model called unsupervised feature selection based on self-representation sparse regression and local similarity preserving, i.e., UFSRL. Specifically, UFSRL is sparse reconstruction of the original data itself, rather than fitting low-dimensional embedding, and the manifold learning exerted on UFSRL model to preserve the local similarity of the data. Moreover, the $l_{2,1/2}$-matrix norm has been imposed on the coefficient matrix, which make the proposed model sparse and robust to noise. In order to solve the proposed model, we design an effective iterative algorithm, and present the analysis of its convergence. Extensive experiments on eight synthetic and real-world data-sets are conducted, and the results of UFSRL compared with six corresponding feature selection algorithms. The experimental results show that UFSRL can effectively identify the feature subset with discriminative while reconstructing the data sparsely, and it is superior to some unsupervised feature selection algorithms in clustering performance.

Keywords Unsupervised feature selection · Sparse reconstruction · Similarity preserving · $L_{2,1/2}$-matrix norm

1 Introduction

In many practical applications, such as pattern recognition [1, 2], data mining [3] and computer vision [4], the data they face are usually represented by high dimensional features [5]. The high-dimensional features make some challenges posed mainly in time and space on the higher requirements to data analysis tasks. In general, the dimensionality of data is far more than the number of samples, so it is easy to fall into the "dimension disaster" by directly using such data for correlation analysis. In data analysis applications [6], not all feature attributes are equally important or discriminatory. Among the high-dimensional features, there are redundant features [7], irrelevant features or even bad features [8] belonging to noise. In addition, for machine learning applications [9], such as classification, clustering and regression tasks in low-dimensional space is relatively easy to achieve, but there are great difficulties in high-dimensional space to complete the task of learning and classification [10, 11]. At a consequence, it is necessary to select a discriminative feature subset using feature selection methods [12]. Feature selection can not only achieve the compress representation of original data by effectively eliminating redundant and irrelevant features, but also improve the quality of learning [13, 14]. In contrast to other reduction techniques based on transformation techniques [15, 16], such as PCA [17, 18], feature selection techniques can preserve the semantics of the original variables [19]. The goal of feature selection is to select a small subset of features with high discriminative power from high dimensional data. And it does not change the variable representation of the original data. Therefore, feature selection techniques are widely used as an effective dimensionality reduction method.

Feature selection algorithm can be roughly divided into three classes, i.e., filter model [20, 21], wrapper model [22], and embedded model [23]. The filter model, such as Variance [24] and Fisher score [25], analyzes the relationship between features to assess their quality. The computational efficiency of filter methods is high, but the selected feature

✉ Ronghua Shang
rhshang@mail.xidian.edu.cn

[1] Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education of China, School of Artificial Intelligence, Xidian University, Xi'an 710071, China

[2] School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, China

Published online: 18 December 2017

subsets usually do not have a good classification performance. Wrapper model can be regarded as a classifier, and the selected feature subset using Wrapper methods is more discriminative than Filter methods usually. However, Wrapper has a high computational cost and is not suitable for large-scale data feature selection. Embedded model, such as SVM-RFE [26], integrates the feature search part with the learning part, and its computational efficiency is lower than that of Filter, and higher than Wrapper. According to whether or not the label information of the data set is adopted, the feature selection algorithm can also be divided into supervised feature selection [27–29], semi-supervised feature selection [30–32] and unsupervised feature selection [33–35]. The supervised feature selection algorithms rely on the category information to select the most discriminative features, according to the correlation between the feature and the category labels. The semi-supervised feature selection algorithms utilize unlabeled data-sets and a small number of labeled. Both the supervised and semi-supervised algorithms require discriminant information in the category label. But in today's large data age, since to get the corresponding label information requires expensive resources, the class label of the data set is usually unknown in many practical applications. In the absence of label information, unsupervised feature selection can determine the importance of features according to the potential characteristics between them. Therefore, the research of unsupervised feature selection algorithm is very important. This paper focuses on the research of unsupervised feature selection algorithms. Its purpose is to reduce the dimensionality of data while make the data information fully utilized.

Recently, some feature selection methods with sparse regularization constraints [36–38] have been proposed. Unlike traditional feature selection algorithms such as Laplacian Score [39], the sparse feature selection algorithms select features jointly by optimizing the sparse model, at the same time, the sparse representation of the data is obtained. In [33], Cai et al. proposed an unsupervised feature selection model based on l_1-norm [40, 41]. extend l_1-norm to l_p-norm $(0<p<1)$, which makes the learning model more sparse. However, many models based on l_1-norm can only solve the problem of binary classification [19], and ignore the relevant information between features. In order to solve this problem, Nie et al. [42] proposed a new model using $l_{2,1}$-norm instead of l_1-norm. However, $l_{2,1}$-norm is also based on l_1-norm. In order to obtain better sparsity, Wang et al. [43] extends $l_{2,1}$-norm to $l_{2,p}$-matrix norm $(0<p<1)$, which can select more sparse feature subsets jointly. When p equals 1/2, the model has the best sparsity and robustness [5, 43, 44]. In this paper, we focus on the feature weighting algorithm with sparse term constraints and use $l_{2,1/2}$-matrix norm.

In addition, the feature selection algorithms achieve dimensionality reduction by removing the redundancy and irrelevant features. Moreover, the classification or clustering performance remain unchanged even improved after feature selection. Redundant features have self-representation properties, i.e., each feature property can be approximated by a linear combination of its associated features [45]. It is easy to understand that a high degree of self-similarity is widespread in natural phenomena. There are similarities between the different blocks of the same image, the time series of climate monitoring are very similar, and the different segments of the coastline are similar, etc. This self-similarity property is widespread in high-dimensional data. And as sparsity results in sparse representation, self-similarity means that each feature attribute can be represented linearly by all other feature attributes.

Traditional feature selection algorithm such as *laplacian score for feature selection* (LapScore) proposed by He et al. [39], constructs the nearest neighbor graph to evaluate each feature to rank all the features, and then one selects features one by one which protect the local data structure. As an extension of the LS method, Zhao et al. proposed *spectral feature selection for supervised and unsupervised learning* (SPEC) in [46]. SPEC evaluates features based on spectral regression. However, both of them neglect the redundant information between features in the process of feature selection. Recently, a series of feature selection algorithms based on spectral clustering have been proposed. Such algorithms usually use the spectral theory to obtain a low dimensional embedding matrix, and then the feature selection problem can be converted to a regression fitting problem. In an algorithm named *unsupervised feature selection for multi-cluster data* (MCFS) [33], a low-dimensional embedding matrix is calculated first, and then each sample is fitted to the embedding matrix by regularized sparse learning. In [47], Zhao et al. proposed an algorithm named *efficient spectral feature selection with minimum redundancy* (MRSF). Both MRSF and MCFS use the same two-step framework, and the difference between them is that the former use $l_{2,1}$-norm constraints instead of l_1-norm in the sparse learning. Since MCFS and MRSF use a two-step strategy, the effect is degraded [42]. In order to overcome such shortcomings, Hou et al. proposed *joint embedding learning and sparse regression: a framework for unsupervised feature selection* (JELSR) in [48] and Fang et al. proposed *locality and similarity preserving embedding for feature selection* (LSPE) in [49]. JELSR and LSPE adopt the joint framework of embedded learning and sparse regression. The feature subset chosen by JELSR can protect locality, while LSPE can protect locality and similarity. However, both JELSR and LSPE use a low-dimensional embedding learning model, which is sensitive to the embedding dimensionality [49]. In addition, different from the above methods which only consider the spatial structure of the data, considering both

the feature space and the data space to protect the local geometrical structure of the original data, Shang et al. proposed *self-representation based dual-graph regularized feature selection clustering* (DFSC) in [50]. However, JELSR, LSPE, and DFSC all use a sparse framework based on $l_{2,1}$-norm.

In order to overcome the shortcomings of low dimension embedding learning, which is very sensitive to embedding dimensionality, and to make the coefficient matrix has good sparsity, we use the sparse self-representation model based on $l_{2,1/2}$-matrix norm. We reconstruct the original data by learning a coefficient matrix, and then fit the data itself. In addition, we also use the manifold learning method to preserve the local similarity of the data.

The main contributions of this paper are summarized as follows:

1. We propose a new feature selection model using sparse self-representation. The model realizes the compression reconstruction of the data by learning a nonnegative self-representation matrix. At the same time, the local geometrical structure information of the original space is protected by constructing the neighborhood graph. In our model, both self-representation sparse reconstruction, similarity preserving and feature selection are considered.
2. To be able to select the sparse and discriminative subset of features, we use $l_{2,p}$-norm $(0<p<1)$ for the regularization term of the self-representation matrix. The sparsity constraint of the feature selection matrix makes the proposed model to reduce the redundancy and noises effectively.
3. In addition, we design an iterative algorithm to solve the objective function of the proposed model, and prove its convergence. In contrast to other feature selection algorithms that require matrix inversion in the solution process, the proposed algorithm is simple and effective.

The rest of this paper is organized as follows. In the second part, the sparse reconstruction and graph embedding are introduced first, and then our UFSRL framework is proposed. In the third part, the solution process of the algorithm and convergence analysis are given in detail. The extensive experiments are described in the fourth part. Finally, we summarize our work with future work.

2 The proposed algorithm

In this part, we first introduce the $L_{2,p}$-matrix norm model involved in the UFSRL framework, and then give the proposed algorithm model.

2.1 $L_{2,p}$-matrix norm sparse model

In general, sparse feature selection methods are able to obtain the most discriminative feature subset, which is the sparse solution, because of a variety of sparse constraints are used in joint model. As in [42, 47–51], $l_{2,1}$-norm has been widely used and achieved good results. Compared to the $l_{2,1}$ norm framework, The framework based on $l_{2,p}$ $(0<p<1)$ matrix norm is proposed by Wang et al. [43]. The $l_{2,p}$-matrix norm has the best performance with p equal to1/2 [44].

For any arbitrary matrix $\boldsymbol{A}\in \mathrm{R}^{s\times t}$, the $L_{l,p}$-norm is defined as follows:

$$||A||_{l,p}=\left(\sum_{i=1}^{s}\left(\sum_{j=1}^{t}|A_{i,j}|^{l}\right)^{p/l}\right)^{1/p} \tag{1}$$

When $l=p=1$, it is the l_1-norm of $\boldsymbol{A}$ and we denote it as $\|\boldsymbol{A}\|_1$.

When $l=p=2$, it is the l_2-norm of $\boldsymbol{A}$ and we denote it as $\|\boldsymbol{A}\|_2$.

When $l=2$, $p=1$, it is the $l_{2,1}$-norm of $\boldsymbol{A}$ and we denote it as $\|\boldsymbol{A}\|_{2,1}$.

When $l=2$, $p=1/2$, it is the $l_{2,1/2}$-norm of $\boldsymbol{A}$ and we denote it as $\|\boldsymbol{A}\|_{2,1/2}$.

It is worth noting that the $l_{2,1/2}$-matrix norm is non-convex, so the objective function is also non-convex. Because of its good sparseness and robustness, the regularization constraint based on the $l_{2,1/2}$-matrix norm is adopted in this paper.

2.2 Objective function

Unsupervised feature selection algorithms can remove redundant and irrelevant features from a given dataset without label information, so as to select the most discriminative feature subsets. Therefore, the unsupervised feature selection method is suitable for large-scale data analysis. Given a dataset $\boldsymbol{X}\in R^{n\times m}$, where n and m are the number of samples and features respectively. In the matrix form, $\boldsymbol{X}=[\boldsymbol{x}_1;\boldsymbol{x}_2;\ldots;\boldsymbol{x}_n]$ or $\boldsymbol{X}=[\boldsymbol{f}_1,\boldsymbol{f}_2,\ldots,\boldsymbol{f}_m]$, where $\boldsymbol{x}_i\in R^m$ $(1\le i\le n)$ and $\boldsymbol{f}_j\in R^n(1\le j\le m)$ is the i-th sample and the j-th feature respectively. Assuming that the number of features to be chosen is r, the unsupervised feature selection algorithms is to select the feature subset $\boldsymbol{X'}\in R^{m\times r}$ from $\boldsymbol{X}\in R^{n\times m}$.

Feature selection aim to select the optimal feature subset $\boldsymbol{X'}$ by searching in the m-dimensional feature space, where $\boldsymbol{X'}=[\boldsymbol{f}_1,\boldsymbol{f}_2,\ldots,\boldsymbol{f}_r]$, $(r<<m)$. Denote $\boldsymbol{U}=[U_1,U_2,\ldots,U_m]$, where U_k $(1\le k\le m)$ represents whether the k-th feature is selected. And $U_k=1$ if and only if the k-th feature is selected,

or $U_k=0$ otherwise. The optimal solution can be obtained by minimizing the objective function as follows:

$$\begin{aligned} &\min_{\mathbf{f}} l(\mathbf{X}-\mathbf{X}Diag(\mathbf{U})) \\ &s.t.\ \ U_k \in \{0,1\} \end{aligned} \tag{2}$$

where $l(\bullet)$ represent a loss function. $Diag(\boldsymbol{U})$ is a diagonal matrix, and the k-th diagonal element $Diag(\boldsymbol{U})_{kk}=U_k$. $\boldsymbol{X}Diag(\boldsymbol{U})$ is a sparsity matrix. The low dimension matrix $\boldsymbol{X'}$ can be obtained by deleting the zero element in $\boldsymbol{X}Diag(\boldsymbol{U})$.

It can be seen that Eq. (2) is an NP-hard problem. In order to solve this problem, we introduce a nonnegative coefficient matrix $\boldsymbol{F}\in R^{m\times m}$, and the following objective function is obtained:

$$\begin{aligned} &\min_{\mathbf{f},\,\mathbf{A}} l(\mathbf{X}-\mathbf{X}Diag(\mathbf{U})\mathbf{F}) \\ &s.\,t.\ \ U_k \in \{0,1\},\ \mathbf{F}\geqslant 0 \end{aligned} \tag{3}$$

We observe that $Diag(\boldsymbol{U})\boldsymbol{F}$ is a row-sparse matrix, we can redefine $\boldsymbol{A}=Diag(\boldsymbol{U})\boldsymbol{F}$ and a sparsity constraints imposed on $\boldsymbol{A}$ to guarantee row sparseness. Finally, we obtain the following objective function:

$$\begin{aligned} &\min_{\mathbf{A}} l(\mathbf{X}-\mathbf{X}\mathbf{A})+||\mathbf{A}||_{l,p}^{p} \\ &s.t.\ \ \mathbf{A}\geqslant 0 \end{aligned} \tag{4}$$

The commonly least-squares error loss function is used in our framework. To holds the equation $\boldsymbol{x_i}=\boldsymbol{x_i}\boldsymbol{A}$ as much as possible, i.e., fit data matrix $\boldsymbol{XA}$, which is sparse, with the original data. In addition, the $l_{2,1/2}$-matrix norm is exerted on the regularization term. Equation (4) can be rewritten as follows:

$$\begin{aligned} &\arg\ \min_{A} ||\mathbf{X}-\mathbf{X}\mathbf{A}||_F^2+\beta||\mathbf{A}||_{2,1/2}^{1/2} \\ &\quad s.t.\ \mathbf{A}\geqslant 0 \end{aligned} \tag{5}$$

where β is a nonnegative constant, the parameter balances the two terms. β should be large enough to ensure that the coefficient matrix is sparse.

In recent years, some feature selection algorithms using manifold learning have been proposed [49], which can preserve the local structure of the data at the same time as the feature selection. Given a dataset $\boldsymbol{X}$, we construct a undirected k-nn graph $G=(V, E)$. V includes all the sample points in dataset $\boldsymbol{X}$, and E includes all the connection edges between points in V. For each point x_i, we can find its nearest k nearest neighborhood set, i.e., $N(\boldsymbol{x}_i)$, and edges is set up if and only if two points $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are neighbors. The weighted matrix $\boldsymbol{W}$ in the constructed graph is defined as follows:

$$\mathbf{W}_{i,j}=\begin{cases} d(\mathbf{x}_i,\mathbf{x}_j), & if\ \mathbf{x}_i\in N_k(\mathbf{x}_j)\ or\ \mathbf{x}_j\in N_k(\mathbf{x}_i) \\ 0, & otherwise. \end{cases} \tag{6}$$

where $N_k(\boldsymbol{x}_i)$ denotes the set of k neighbors of $\boldsymbol{x}_i$. $d(\boldsymbol{x}_i, \boldsymbol{x}_j)$ measures the similarity between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, we can use Gaussian kernel $e^{-\frac{||\mathbf{x}_i-\mathbf{x}_j||^2}{2\sigma^2}}$ or 0–1 weighting.

In the proposed model, we want to select the subset of features which preserve the original structure of the original data. The low dimension representation of the adjacent points ($\boldsymbol{x}_i$ and $\boldsymbol{x}_j$) in the high-dimensional data, i.e. ($\boldsymbol{x}_i\boldsymbol{A}$ and $\boldsymbol{x}_j\boldsymbol{A}$) should also be adjacent to each other [52]. In order to protect the original geometrical structure of the original data, the method of graph embedding is introduced. By minimizing the Eq. (7), the local structure information of the original data can be best preserved.

$$\begin{aligned} &\frac{1}{2}\sum_{i,j=1}^{n}||\mathbf{x}_i\mathbf{A}-\mathbf{x}_j\mathbf{A}||^2\mathbf{W}_{ij} \\ &=\sum_{i,j=1}^{n}(\mathbf{x}_i\mathbf{A})\,\mathbf{W}_{ij}(\mathbf{x}_i\mathbf{A})^T-\sum_{i,j=1}^{n}(\mathbf{x}_i\mathbf{A})\mathbf{W}_{ij}(\mathbf{x}_j\mathbf{A})^T \\ &=\sum_{i=1}^{n}(\mathbf{x}_i\mathbf{A})\,\mathbf{D}_{ij}(\mathbf{x}_i\mathbf{A})^T-\sum_{i,j=1}^{n}(\mathbf{x}_i\mathbf{A})\mathbf{W}_{ij}(\mathbf{x}_j\mathbf{A})^T \\ &=\mathbf{Tr}\left((\mathbf{X}\mathbf{A})^T\mathbf{D}(\mathbf{X}\mathbf{A})\right)-\mathbf{Tr}\left((\mathbf{X}\mathbf{A})^T\mathbf{W}(\mathbf{X}\mathbf{A})\right) \\ &=\mathbf{Tr}\left(\mathbf{A}^T\mathbf{X}^T\mathbf{L}\mathbf{X}\mathbf{A}\right) \end{aligned} \tag{7}$$

The weighted matrix of the graph $G=(V, E)$ is $\boldsymbol{W}$, $\boldsymbol{D}$ is a diagonal matrix, and $\mathbf{D}_{ii}=\sum_j\mathbf{W}_{ij}$. The Laplacian matrix embedded in graph G is $\boldsymbol{L}=\boldsymbol{D}-\boldsymbol{W}$. Note that $\boldsymbol{W}$ is a symmetric matrix in the undirected graph. The Laplacian matrix $\boldsymbol{L}$ contains the local neighbor information of the original data, so it can protect the essential structure of the data.

In UFSRL, self-representation sparse reconstruction model, local similarity protection and feature selection are integrated to an objective function, which makes our model more suitable for feature selection. We update the objective function in Eq. (5) to obtain the final objective function:

$$\begin{aligned} &\arg\ \min_{A} ||\mathbf{X}-\mathbf{X}\mathbf{A}||_F^2+\alpha Tr\left(\mathbf{A}^T\mathbf{X}^T\mathbf{L}\mathbf{X}\mathbf{A}\right)+\beta||\mathbf{A}||_{2,1/2}^{1/2} \\ &\quad s.t.\ \ \mathbf{A}\geqslant 0 \end{aligned} \tag{8}$$

where $Tr(\boldsymbol{B})$ denotes the trace of matrix $\boldsymbol{B}$. The balance parameter $\alpha>0, \beta>0$.

For the self-representation coefficient matrix $\boldsymbol{A}$, we impose the nonnegative constraint on $\boldsymbol{A}$. The regularization term constraint must be introduced not only to guarantee the sparseness of $\boldsymbol{A}$, but also to avoid the trivial solution, i.e., $\boldsymbol{A}=\boldsymbol{I}_m$. Let $\boldsymbol{A}=[\boldsymbol{A}_1;\boldsymbol{A}_2;\ldots;\boldsymbol{A}_m]$, $\boldsymbol{A}_i$ is the i-th row of the matrix $\boldsymbol{A}$. The value of $\|\boldsymbol{A}_i\|_2$ reflects the importance of the i-th feature relative to whole features. So we can sort all the m features according to $\|\boldsymbol{A}_i\|_2$, and then select the first discriminative r features.

2.3 Comparison with other feature selection methods

Obviously, sparse reconstruction, local geometrical structure similarity and feature selection are considered in the proposed UFSRL framework. We know that both SPEC and LapScore are graph-based algorithms. In order to protect the original structure, UFSRL also uses the method of graph embedding to protect the local similarity. LapScore selects the feature subset that can best preserves the locality of data, however, UFSRL selects the feature subset that can minimize the sparse reconstruction error while preserving the similarity structure of the data. SPEC can be viewed as an extension of LapScore. MCFS uses a spectral regression framework, which divides spectral learning and sparse learning into two steps. We use graph embedding and sparse learning jointly in the proposed UFSRL. JELSR [48] and LSPE [49] solve embedding learning and sparse learning jointly. The features chosen by JELSR preserve the locality well, while LSPE can preserve locality and similarity simultaneously. However, since both JELSR and LSPE are the regression model based on the low-dimensional embedding, the two algorithms are very sensitive to the dimensionality of embedding. UFSRL is the sparse regression model based on the self-representation, which can preserve data local similarity in low dimensional space and high dimensional space. In addition, different from the above methods which only consider the structure of data space, DFSC [50] considers both feature space and data space to integrate self-representation structure, local manifold learning and the sparse learning into an objective function. However, same as JELSR and LSPE, DFSC uses a sparse framework based on $l_{2,1}$-norm. To make the coefficient matrix learned more sparse and robust, the regularization term constraint based on $l_{2,1/2}$-matrix norm is used in the proposed UFSRL.

3 Optimization for solving UFSRL

3.1 Iterative algorithm

From [53], an efficient iterative algorithm is developed to optimize the objective function as Eq. (8) and then the feature weight matrix $\boldsymbol{A}$ is obtained.

Given a square matrix $\boldsymbol{S}=[\boldsymbol{s}_1; \boldsymbol{s}_2; \ldots; \boldsymbol{s}_m]$, we can obtain $||\mathbf{S}||_{2,1/2}^{1/2} = 4Tr\left(\mathbf{S}^T\mathbf{\Lambda}\mathbf{S}\right)$, and $\mathbf{\Lambda}$ is a diagonal matrix. So we rewrite Eq. (8) as follows:

$$\begin{aligned} &\min_{\mathbf{A}} \ Tr[(\mathbf{X}-\mathbf{X}\mathbf{A})^T(\mathbf{X}-\mathbf{X}\mathbf{A})] + \alpha Tr(\mathbf{A}^T\mathbf{X}^T\mathbf{L}\mathbf{X}\mathbf{A}) + 4\beta Tr(\mathbf{A}^T\mathbf{D}\mathbf{A}) \\ &s.t. \quad \mathbf{A} \geqslant 0 \end{aligned} \tag{9}$$

where $\mathbf{D}_{ii} = {}^1\!/_4||\mathbf{a}_i||_2^{3/2}$. We set $\beta=4\beta$, and define:

$$P(\mathbf{A}) = \ Tr[(\mathbf{X}-\mathbf{X}\mathbf{A})^T(\mathbf{X}-\mathbf{X}\mathbf{A})] + \alpha Tr(\mathbf{A}^T\mathbf{X}^T\mathbf{L}\mathbf{X}\mathbf{A}) + \beta Tr(\mathbf{A}^T\mathbf{D}\mathbf{A}) \tag{10}$$

For constraint $\boldsymbol{A} \geq 0$, a Lagrange multiplier is introduced, i.e., $\psi = [\varphi_{ij}]$. Then the Lagrange function of the Eq. (9) obtained:

$$L(\mathbf{A}) = P(\mathbf{A}) + Tr(\phi\mathbf{A}) \tag{11}$$

Let $\frac{dL(\mathbf{A})}{d\mathbf{A}} = 0$, the Eq. (12) is obtained:

$$2(\mathbf{X}^T\mathbf{X} + \alpha\mathbf{X}^T\mathbf{L}\mathbf{X} + \beta\mathbf{D})\mathbf{A} - 2\mathbf{X}^T\mathbf{X} + \phi = 0 \tag{12}$$

Because the KKT condition [47] $\phi_{ij}\mathbf{A}_{ij} = 0$ is used, we get the following updating rules:

$$[(\mathbf{X}^T\mathbf{X} + \alpha\mathbf{X}^T\mathbf{L}\mathbf{X} + \beta\mathbf{D})\mathbf{A} - \mathbf{X}^T\mathbf{X}]_{ij}\mathbf{A}_{ij} = 0 \tag{13}$$

We define $\boldsymbol{M}=\boldsymbol{X}^T\boldsymbol{X}+\alpha\boldsymbol{X}^T\boldsymbol{L}\boldsymbol{X}+\beta\boldsymbol{D}$, and $\boldsymbol{M}=\boldsymbol{M}^+-\boldsymbol{M}^-$. The updating rule of Eq. (10) is obtained:

$$\mathbf{A}_{ij} \leftarrow \mathbf{A}_{ij}\frac{[\mathbf{M}^-\mathbf{A} + \mathbf{X}^T\mathbf{X}]_{ij}}{[\mathbf{M}^+\mathbf{A}]_{ij}} \tag{14}$$

The detail of UFSRL algorithm is shown in Algorithm 1.

Algorithm 1: The UFSRL algorithm

Input: The Data matrix $\boldsymbol{X}\in R^{n\times m}$, the parameters α, β, k, T;

Initial: Set $\boldsymbol{A}\in R^{m\times m}$ as a ones matrix, $\boldsymbol{D}\in R^{m\times m}$as an identity matrix;

1. Construct k-nn graph and compute $\boldsymbol{L}$;
2. For t=1:T
3. Compute $\boldsymbol{M}_t=\boldsymbol{X}^T\boldsymbol{X}+\alpha\boldsymbol{X}^T\boldsymbol{L}\boldsymbol{X}+\beta\boldsymbol{D}$;
4. Compute $\boldsymbol{M}_t^+=(|\boldsymbol{M}_t|+\boldsymbol{M}_t)/2$, $\boldsymbol{M}_t^-=(|\boldsymbol{M}_t|-\boldsymbol{M}_t)/2$;
5. Update $\boldsymbol{A}$ as $\boldsymbol{A}_{ij}^{t+1} \leftarrow \boldsymbol{A}_{ij}^t\frac{[\boldsymbol{M}^-\boldsymbol{A}+\boldsymbol{X}^T\boldsymbol{X}]_{ij}}{[\boldsymbol{M}^+\boldsymbol{A}]_{ij}}$;
6. Update the diagonal matrix $\boldsymbol{D}$ as $\boldsymbol{D}_{ii}^{t+1} = {}^1\!/_4||\boldsymbol{a}_i^{t+1}||_2^{3/2}$;

Output: Rank all m features in descending order according to $||\boldsymbol{a}_i^t||_2$, and then select top r features.

3.2 Convergence analysis

In this section, we analyze the convergence of the proposed UFSRL algorithm.

Theorem 1 *The value of the objective function as Eq.* (10) *decreases monotonically under the updating rules in Eq.* (14).

In the next, we prove the Theorem 1 is true.

Definition 1 If two conditions $\Omega(\boldsymbol{B},\boldsymbol{B}') \geq F(\boldsymbol{B})$ and $\Omega(\boldsymbol{B},\boldsymbol{B})=F(\boldsymbol{B})$ are satisfied, $\Omega(\boldsymbol{B},\boldsymbol{B}')$ is an auxiliary function for $P(\boldsymbol{B})$. Then F is non-increasing under the following updating formula:

$$\mathbf{B}^{t+1} = \arg\min_{W} \Omega\left(\mathbf{B}, \mathbf{B}^t\right) \tag{15}$$

Lemma 1 *The* $\Omega(\boldsymbol{A}_{ij}, \boldsymbol{A}_{ij}^t)$ *as Eq.* (16) *is an auxiliary function of* $P(\boldsymbol{A}_{ij})$ *in objective function* (10).

$$\Omega\left(\mathbf{A}_{ij}, \mathbf{A}_{ij}^{t}\right) = P_{ij}\left(\mathbf{A}_{ij}^{t}\right) + P_{ij}^{'}\left(\mathbf{A}_{ij}^{t}\right)\left(\mathbf{A}_{ij} - \mathbf{A}_{ij}^{t}\right) + \frac{[\mathbf{M}^{+}\mathbf{A}^{t}]_{ij}}{\mathbf{A}_{ij}^{t}}\left(\mathbf{A}_{ij} - \mathbf{A}_{ij}^{t}\right)^{2} \tag{16}$$

Proof The Taylor series expansion of $P_{ij}(\boldsymbol{A}_{ij})$ is:

$$P_{ij}(\mathbf{A}_{ij}) = P_{ij}\left(\mathbf{A}_{ij}^{t}\right) + P_{ij}^{'}\left(\mathbf{A}_{ij}^{t}\right)\left(\mathbf{A}_{ij} - \mathbf{A}_{ij}^{t}\right) + \frac{1}{2}P_{ij}^{''}\left(\mathbf{A}_{ij}^{t}\right)\left(\mathbf{A}_{ij} - \mathbf{A}_{ij}^{t}\right)^{2} \tag{17}$$

where $P_{ij}^{\prime}\left(A_{ij}\right)$ and $P_{ij}^{\prime\prime}\left(A_{ij}\right)$ is the first-order and second-order derivative of $P_{ij}(\boldsymbol{A}_{ij})$ with respect to $\boldsymbol{A}$ respectively. Therefore, we get

$$P_{ij}^{'}(\mathbf{A}_{ij}) = \left(2\mathbf{X}^{T}\mathbf{X}\mathbf{A} - 2\mathbf{X}^{T}\mathbf{X} + 2\alpha\mathbf{X}^{T}\mathbf{L}\mathbf{X}\mathbf{A} + 2\beta\mathbf{D}\mathbf{A}\right)_{ij}; \tag{18}$$

$$P_{ij}^{''}(\mathbf{A}_{ij}) = \left(2\mathbf{X}^{T}\mathbf{X} + 2\alpha\mathbf{X}^{T}\mathbf{L}\mathbf{X} + 2\beta\mathbf{D}\right)_{ii}. \tag{19}$$

We have $\boldsymbol{M} = \boldsymbol{X}^{T}\boldsymbol{X} + \alpha\boldsymbol{X}^{T}\boldsymbol{L}\boldsymbol{X} + \beta\boldsymbol{D}$, so $[\boldsymbol{M}^{+}\boldsymbol{A}_{ij}{}^{t}]_{ij} \geq \boldsymbol{A}_{ij}{}^{t}\boldsymbol{M}_{ii}$ is established. The inequation $\Omega(\boldsymbol{A}_{ij}, \boldsymbol{A}_{ij}{}^{t}) \geq P(\boldsymbol{A}_{ij})$ provided since $\frac{[\mathbf{M}^{+}\mathbf{A}^{t}]_{ij}}{\mathbf{A}_{ij}^{t}} \geqslant \frac{1}{2}P_{ij}^{''}\left(\mathbf{A}_{ij}^{t}\right)$ s satisfied. Therefore, $\Omega(\boldsymbol{A}_{ij}, \boldsymbol{A}_{ij}{}^{t}) \geq P(\boldsymbol{A}_{ij})$. And the equation $\Omega(\boldsymbol{A}_{ij}, \boldsymbol{A}_{ij}) = P(\boldsymbol{A}_{ij})$ is established constantly. So the Lemma 1 holds.

Proof Replacing $\Omega(\boldsymbol{W}_{ij}, \boldsymbol{W}_{ij}{}^{t})$ in Eq. (15) by Eq. (16), then we set $\frac{\partial\Omega\left(\mathbf{A}_{ij}, \mathbf{A}_{ij}^{t}\right)}{\partial\mathbf{A}_{ij}} = 0$. We can get:

$$\mathbf{A}_{ij}^{t+1} \leftarrow \mathbf{A}_{ij}^{t}\frac{[\mathbf{M}^{-}\mathbf{A}^{t} + \mathbf{X}^{T}\mathbf{X}]_{ij}}{[\mathbf{M}^{+}\mathbf{A}^{t}]_{ij}} \tag{20}$$

We can see that the Eq. (20) is the thame as Eq. (14). So the Theorem 1 holds.

According to Theorem 1, the local optimal $\boldsymbol{A}$ obtained using the iterative approach in Algorithm 1.

4 Experiments and analysis

In this section, we validate the effectiveness of the proposed UFSRL algorithm through experiments. We compare the proposed UFSRL with the six feature selection algorithms in one synthetic dataset and seven real-world data-sets. At first, k features are selected by those algorithms, and then k-means clustering is used to get the clustering results. We then compare the clustering performance of UFSRL with those of LapScore, SPEC, MCFS, JELSR, LSPE and DFSC using the clustering results. Finally, we discuss the sensitive of parameters on the UFSRL algorithm.

4.1 Data sets

We conduct extension experiments, 7 real-world data-sets including *Umist, ORL, Isolet, Ionosphere, BC, Dbworld_bodies* and *Prostate-GE* are used in total. Table 1 show the information about the data-sets in detail.

4.2 Compared methods

In order to verify the effectiveness of the proposed UFSRL as a feature selection algorithm, we use all the features as a benchmark, and with several unsupervised feature selection algorithm were compared. The compared algorithms are summarized as follows.

1. Baseline: all features of data-sets are used.
2. LapScore [39]: feature subset preserving the local manifold structure is selected.
3. MCFS [33]: feature subset is selected by spectral analysis and sparse regression based on l_1-norm regularization.
4. SPEC [46]: feature subset is selected using spectral clustering.
5. JELSR [48]: feature subset best protect the locality of data is selected using embedding leaning and sparse regression jointly.
6. LSPE [49]: feature subset best protect the similarity and locality of original data is selected.
7. DFSC [50]: feature subset preserving the local manifold structure of both data space and feature space is selected.

4.3 Evaluation metrics

After the feature selection is completed, we use the k-means algorithm to cluster the reduced dimension data. For the evaluation of clustering effect, two most commonly evaluation criterion are used, i.e., clustering Accuracy (ACC) [54] and Normalized Mutual Information (NMI). Here, the

Table 1 Datasets in detail

Dataset	Instances	Features	Classes
Umist	575	644	20
ORL	400	1024	40
Isolet	1560	617	26
Ionosphere	351	34	2
BC	569	30	2
Prostate-GE	102	5699	2
Dbworld_bodies	64	4702	2

larger of ACC and NMI show the better performance of the algorithms.

Given a sample $\boldsymbol{x}_i$, p_i and r_i are the clustering labels and the ground truth labels of $\boldsymbol{x}_i$, respectively. The ACC is defined as follows:

$$ACC = \frac{1}{n}\sum_{i=1}^{n}\delta\left(p_i, map\left(r_i\right)\right) \tag{21}$$

where n is the number of samples, and map(•) is the optimal mapping function to permute clustering labels and the truth labels which can be obtained by using *Hungarian* algorithm [55]. $\delta(\bullet)$ is an indicator function, and $\delta(x, y) = 1$ if $x = y$, or $\delta(x, y) = 0$ otherwise.

The NMI is defined as follows:

$$NMI = \frac{MI(P, Q)}{\max(H(P), H(Q))} \tag{22}$$

Here, P and Q are the clustering labels and truth labels of data-sets respectively, and $MI(P,Q)$ is the information entropy between P and Q.

We know that the values of ACC and NMI are between 0 and 1. It is worth noting that, ACC and NMI are two different evaluation criteria. ACC matches the clustering labels and the real labels one by one, while NMI reflects the clustering labels and the real label consistency. For a dataset, it is possible that the best ACC and NMI are not obtained at the same time.

4.4 Experimental settings

In this paper, UFSRL proposed and other algorithms compared need to set some parameters. For the feature selection algorithm based graph embedding, we need to set the bandwidth σ of Gaussian function and the parameter k to construct the k-nn graph. For LSPE, JELSR, LapScore, MCFS, SPEC, DFSC and UFSRL, the value of k is chosen from {5, 10, 15} and the value of σ is chosen from $\{10^0, 10^3, 10^5\}$ respectively. For DFSC, we tune α from {0.01, 0.1, 0.5, 1.0, 3.0, 5.0, 7.0, 9.0, 11.0, 13.0, 15.0, 17.0}, the value of λ is searched from {300, 500, 800, 1000, 2000, 3000, 4000, 5000, 6000, 7000, 8000}. And we set β as $\{10, 10^2, 10^3\}$ in DFSC. We tune other parameters in the proposed UFSRL algorithm by a greedy-search way. And we set α as {0.1, 1.0, 3.0, 5.0, 7.0, 10, 13, 15, 17, 20}, β is searched from $\{10^2, 5\times10^2, 10^3, 5\times10^3, 10^4, 5\times10^4, 10^5, 5\times10^5, 10^6, 5\times10^6\}$. In addition, the number of selected features r does not exceed a half of the number of features. In particular, r is no more than 1000 on *Prostate-GE* and *Dbworld_bodies*. In experiments, the k-means algorithm is used for clustering and we set the number of clusters equal to the number of real classes in data-sets.

For all algorithms, we adjust the parameters involved and then list the best clustering results in the Tables 2 and 4. Normally, different data-sets can achieve the best ACC and NMI under different parameters, and the best ACC and NMI may require different parameters. We adopt k-means clustering for the data after feature reduction, i.e., dimensionality reduce, while the performance of k-means depends on initialization. So we repeat the clustering 100 times with random initialization for each step, and then the average results with the standard deviation (SD) are output.

4.5 The effectiveness of the UFSRL

In this part, we use a synthetic data set to validate the effectiveness of the proposed feature selection algorithm. The "*iris*" dataset with noises is adopted in our experiment. The original "*iris*" dataset is a matrix with 150×4, consisting of 150 samples and 4 feature attributes. In order to verify the validity of the UFSRL, we add Gaussian noise to the "*iris*" dataset, and the "*iris-noisy*" dataset obtained. The size of "*iris-noisy*" is 150×50, i.e., increasing the feature attributes to 50. We use "*iris-noisy*" as the synthetic dataset, in which the first four features are the original features.

In our experiment, the proposed UFSRL adopted in "*iris-noisy*" and we get the coefficient matrix $\boldsymbol{A}$. The coefficient

Table 2 The clustering ACC of some FS algorithms on seven datasets (mean ± SD%)

Algorithms	Umist	ORL	Isolet	Ionosphere	BC	Prostate-GE	Dbworld-bodies
Baseline	44.47 ± 2.10	51.45 ± 3.12	58.08 ± 3.84	70.69 ± 1.72	85.41 ± 0.00	58.51 ± 0.46	62.70 ± 12.02
LapScore	38.55 ± 1.99	44.24 ± 2.13	48.96 ± 2.76	68.38 ± 0.00	85.24 ± 0.00	57.42 ± 0.49	71.86 ± 16.35
SPEC	41.49 ± 3.65	48.77 ± 3.14	50.32 ± 3.38	68.03 ± 2.82	76.78 ± 6.51	61.35 ± 2.30	72.17 ± 16.58
MCFS	48.98 ± 3.68	50.49 ± 2.82	55.49 ± 3.80	69.64 ± 0.55	83.13 ± 0.00	61.37 ± 1.39	91.14 ± 0..74
JELSR	51.11 ± 2.44	51.19 ± 2.51	59.38 ± 3.45	69.62 ± 0.22	85.32 ± 0.00	61.76 ± 0.00	91.34 ± 1.26
LSPE	49.76 ± 2.62	50.77 ± 2.87	58.80 ± 3.41	71.84 ± 0.11	85.24 ± 0.00	62.39 ± 0.47	**93.75 ± 0.00**
DFSC	45.28 ± 2.99	53.36 ± 2.77	61.30 ± 2.87	82.62 ± 0.00	86.99 ± 0.00	60.72 ± 6.98	89.72 ± 6.33
UFSRL	**52.97 ± 4.06**	**54.29 ± 2.88**	**64.29 ± 2.82**	**84.30 ± 0.12**	**89.46 ± 0.00**	**63.73 ± 0.00**	91.47 ± 1.10

The best results are marked in bold

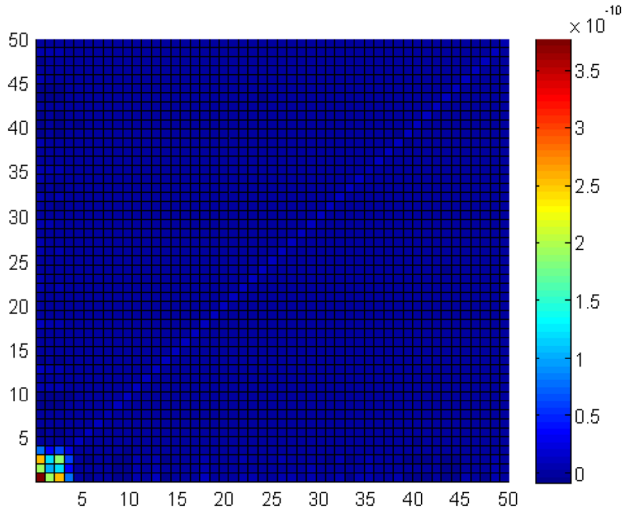

Fig. 1 The sparse matrix $\boldsymbol{A}$

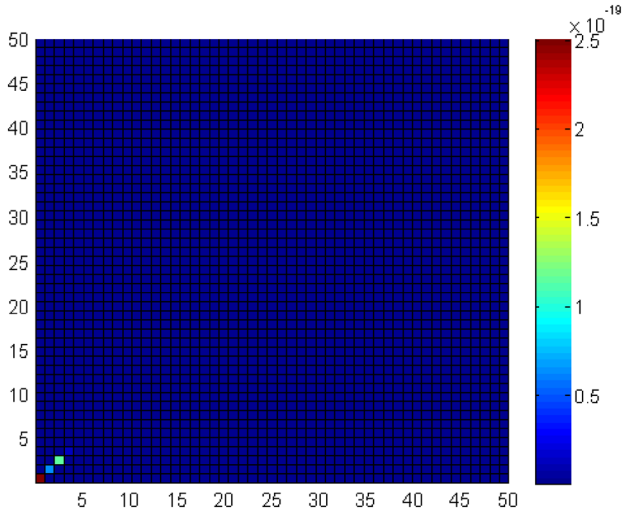

Fig. 2 The feature weighting $\|\boldsymbol{A}_i\|_2$

matrix $\boldsymbol{A}$ has the property of row sparse. In order to make it easier to see the sparse nature of $\boldsymbol{A}$, we plot the values of the elements in the matrix (Best viewed in color).

Figure 1 shows the coefficient matrix $\boldsymbol{A}$ obtained by UFSRL algorithm. We can clearly see that the coefficient matrix $\boldsymbol{A}$ is sparse and only the first four elements are greater than zero. It is also proven that the $l_{2,1/2}$-matrix norm is suitable for sparse constraints in the feature selection algorithm. After the matrix $\boldsymbol{A}$ obtained, the feature weighting, i.e. the value of $\|\boldsymbol{A}_i\|_2$ is calculated. And $\|\boldsymbol{A}_i\|_2$ indicates the importance of the i-th feature in the "*iris-noisy*" dataset. As shown in Fig. 2, we plot $\|\boldsymbol{A}_i\|_2$. It is obvious that the UFSRL selected the first four features which are the most discriminative features. The fact that UFSRL can effectively identify the feature subset with discriminant power has been proven. The experiment results on synthetic data fully demonstrate the effectiveness of the proposed algorithm as feature selection.

4.6 Experimental results on real-world data

We tune the parameters for each dataset and record the best clustering results. Table 2 shows the mean ACC with standard deviation of these algorithms on seven data-sets. The average NMI with standard deviation is presented in Table 4. In Tables 2 and 4, the best results are marked in bold. In addition, in order to prove the validity of UFSRL from the point of view of statistical test, the Wilcoxon signed rank test [56] is used to test between UFSRL and other feature selection algorithms. The significance level is 0.05. The statistical test results for clustering ACC and NMI are listed in Tables 3 and 5. For the Wilcoxon signed rank test, p is the probability of a hypothesis of equal median for two pairs of samples, and h is the test result. If there is no significant difference for the median between UFSRL and another compared algorithms, then $h=0$; otherwise, a significant difference, then $h=1$.

From Tables 2 and 3, it is easy to get following observation. Overall, the proposed UFSRL performs best on almost all data-sets. From Table 2 on the results of clustering ACC can be seen, the proposed the UFSRL algorithm has the best results in six of seven algorithms. From Table 3, we can see that the corresponding to the value of h is equal to 1, p is approximately equal to 0. Note UFSRL is superior to other feature selection algorithms in data-sets except

Table 3 The statistic results of clustering ACC on seven datasets after 100 independent runs

Datasets	LapScore		SPEC		MCFS		JELSR		LSPE		DFSC	
	p	h	p	h	p	h	p	h	p	h	p	h
Umist	3.9e−18	1	4.4e−18	1	6.7e−9	1	7.2e−5	1	1.2e−8	1	7.6e−17	1
ORL	3.9e−18	1	1.0e−16	1	7.4e−13	1	3.1e−12	1	5.7e−12	1	0.0170	1
Isolet	3.9e−18	1	3.9e−18	1	9.6e−18	1	2.6e−15	1	7.7e−16	1	7.5e−9	1
Ionosphere	1.4e−22	1	3.7e−18	1	5.6e−19	1	2.3e−20	1	1.9e−20	1	1.4e−22	1
BC	1.5e−23	1	1.9e−18	1	1.5e−23	1	3.1e−19	1	1.5e−23	1	1.5e−23	1
Prostate_GE	2.7e−19	1	2.1e−15	1	6.8e−19	1	6.8e−19	1	1.7e−19	1	1.4e−6	1
Dbworld-bodies	1.3e−14	1	2.0e−16	1	0.0174	1	0.5259	0	1.4e−17	1	0.0118	1

Table 4 The NMI of some FS algorithms on seven datasets (mean ± SD%)

Algorithms	Umist	ORL	Isolet	Ionosphere	BC	Prostate-GE	Dbworld-bodies
Baseline	64.20 ± 1.62	71.81 ± 1.84	74.31 ± 1.72	12.19 ± 3.20	42.23 ± 0.00	2.22 ± 0.29	10.99 ± 18.08
LapScore	55.28 ± 1.45	66.30 ± 1.57	65.57 ± 1.08	8.81 ± 0.00	41.79 ± 0.00	1.58 ± 0.21	23.83 ± 27.72
SPEC	56.23 ± 3.08	70.05 ± 2.10	65.15 ± 1.81	8.50 ± 2.93	22.58 ± 12.23	5.03 ± 2.01	28.16 ± 25.60
MCFS	68.31 ± 2.42	70.68 ± 1.67	71.27 ± 1.67	10.18 ± 0.57	37.73 ± 0.00	4.49 ± 0.37	65.99 ± 4.97
JELSR	66.85 ± 1.56	71.31 ± 1.40	74.93 ± 1.43	10.49 ± 0.47	42.00 ± 0.22	5.74 ± 0.00	57.53 ± 4.24
LSPE	**69.98 ± 1.38**	71.14 ± 1.65	73.32 ± 1.69	13.41 ± 0.18	41.79 ± 0.00	5.83 ± 0.07	**68.09 ± 0.00**
DFSC	65.49 ± 2.13	72.73 ± 1.33	74.54 ± 1.40	29.31 ± 0.00	42.23 ± 0.00	5.05 ± 4.97	55.69 ± 13.24
UFSRL	67.40 ± 2.25	**73.74 ± 1.50**	**76.59 ± 1.31**	**33.54 ± 0.31**	**53.23 ± 0.00**	**7.08 ± 0.00**	65.45 ± 3.24

The best results are marked in bold

Table 5 The statistic results of clustering ACC on seven datasets after 100 independent runs

Datasets	LapScore		SPEC		MCFS		JELSR		LSPE		DFSC	
	p	h	p	h	p	h	p	h	p	h	p	h
Umist	3.9e−18	1	4.4e−18	1	6.7e−9	1	7.2e−5	1	1.2e−8	1	7.6e−17	1
ORL	3.9e−18	1	1.0e−16	1	7.4e−13	1	3.1e−12	1	5.7e−12	1	0.0170	1
Isolet	3.9e−18	1	3.9e−18	1	9.6e−18	1	2.6e−15	1	7.7e−16	1	7.5e−9	1
Ionosphere	1.4e−22	1	3.7e−18	1	5.6e−19	1	2.3e−20	1	1.9e−20	1	1.4e−22	1
BC	1.5e−23	1	1.9e−18	1	1.5e−23	1	3.1e−19	1	1.5e−23	1	1.5e−23	1
Prostate_GE	2.7e−19	1	2.1e−15	1	6.8e−19	1	6.8e−19	1	1.7e−19	1	1.4e−6	1
Dbworld-bodies	1.9e−15	1	4.5e−17	1	0.5918	0	7.8e−14	1	4.2e−14	1	7.4e−10	1

for *Dbworld-bodies*. Specifically, the UFSRL is superior to DFSC on all data-sets. and the ACC have the obvious increases on all data-sets. Compared with the LSPE algorithm, UFSRL outperforms the six data-sets except for the *Dbworld-bodies* dataset and the corresponding value of h is 1. Moreover, the proposed UFSRL has significantly improved clustering results than JELSR algorithm in terms of ACC on six data-sets, and the increase in the *Dbworld-bodies* is not significant. In short, the effectiveness of the UFSRL as a feature selection algorithm is indicated in terms of the results of clustering ACC and statistical test.

From Tables 4 and 5, we have the observation that the performance of UFSRL in terms of NMI is almost the same as that on ACC. From Table 4, the proposed UFSRL achieves the best results in five of the seven data-sets, and its increase is significant in terms of the results of Wilcoxon signed rank test. The UFSRL algorithm outperforms the DFSC on all these data-sets. The LSPE compared algorithm has the best results on the *Umist* and *Dbworld-bodies* data-sets, but its performance on all the other 5 data-sets fail to the proposed UFSRL. While the MCFS algorithm has better performance than the UFSRL algorithm only on the *Umist* dataset and *Dbworld-bodies*, and the results on the other 6 data-sets are obviously behind the UFSRL. And the value of h is equal to zero on *Dbworld-bodies* indicates that the difference between the results of UFSRL and MCFS was not significant. The effectiveness of the proposed UFSRL as a feature selection algorithm has been proven in terms of NMI and statical test on these data-sets.

In short, experimental results on real-world data-sets show that UFSRL can select more discriminative feature subset than other feature selection algorithms compared in this paper. There are four main reasons for this. (1) The self-representation regression model are used in UFSRL to effectively exploit the self-representation characteristics between features, making the selected features representative. (2) The UFSRL protects the local similarity of the data points by constructing the neighborhood graph, so that the local structure of the original data is preserved. (3) More important, the $l_{2,1/2}$-matrix norm constraints is imposed on the feature selection matrix so that the model can effectively remove redundancy and noise. (4) Multiple problems are solved by a joint framework at the same time in UFSRL model. Both self-representation sparse reconstruction and local similarity preserving are integrated into the proposed UFSRL model, and the regularization term using $l_{2,1/2}$-matrix norm makes UFSRL algorithm more suitable for feature selection. More important, the proposed UFSRL utilizes the sparse reconstruction based on self-representation instead of the low-dimension embedding learning, so there is no sensitivity to the embedding dimensionality. Meanwhile, the experimental results of UFSRL, DFSC, LSPE and JELSR also show that it is more effective to solve several problems at the same time than solve them one by one.

4.7 Computational complexity

In this section, we analyze the computational complexity of the UFSRL algorithm. First, we need construct the neighborhood matrix $\boldsymbol{L}$ with the computational complexity of $O(mn^2)$. Next, each iteration to update $\boldsymbol{A}$ need $O(cmn)$ operations. Assuming t is the number of iterations in UFSRL, the total computational complexity of UFSRL is $O(mn^2+tcmn)$.

The computational complexity of the seven feature selection algorithms is listed in Table 6. In Table 6, n is the number of samples, m is the number of features, c is the embedded dimension, t represents the number of iterations, and r is the number of selected features. Compared with the feature selection algorithms under joint framework, the computational complexity of the UFSRL algorithm is lower than that of the JELSR, LSPE, and DFSC algorithms.

In the experiment, the proposed UFSRL and other comparision algorithms is implemented in MATLAB platform on a desktop with 8 GB RAM and an Intel Core i5 CPU @ 3.2 GHz. Table 7 shows the time taken by the feature selection algorithm on each data set and the number of selected features. In Table 7, *time* represents the CPU time required for each algorithm to complete the feature selection process, and r represents the number of selected features. In addition, the mean values of *time* and r are given in the last line of Table 7.

Table 6 Computational complexity analysis

Algorithms	Time complexity
LapScore	$O(mn^2)$
SPEC	$O(mn^2)$
MCFS	$O(mn^2+cr^3+cnr^2)$
JELSR	$O(mn^2+t(n^3+cmn))$
LSPE	$O(mn^2+t(n^3+cmn))$
DFSC	$O(m^2n+mn^2+tcmn)$
UFSRL	$O(mn^2+tcmn)$

From Table 7, it can be seen that the algorithms need to select different numbers of features on different data sets in order to achieve the best clustering results. In addition, compared to JELSR, LSPE and DFSC feature selection algorithms, the proposed UFSRL used less time.

4.8 Parameters sensitivity

As mentioned above, the proposed UFSRL have some parameters to be set in advance, such as k, σ, α, β and r. Here we focus on the impact of two main parameters for the algorithm, i.e., α and β, which control the local similarity of data and the row sparsity of the feature selection matrix respectively. Since the parameter determine is still an open problem to solve, we use the greedy method to search for these two parameters, and then tune the values of α and β within the determined range. For seven data-sets, i.e., *Umist, Isolet, ORL, Ionosphere, BC, Prostate-GE* and *Dbworld_bodies*, we fix k, σ and r as constants. For different data-sets, the values of the parameters k, σ and r may be different. The value of α is set as $\{0.1, 3.0, 7.0, 13, 17\}$, and β is chosen from $\{10^2, 10^3, 10^4, 10^5, 10^6\}$. The k-means algorithm adopted in experiments, and we take the average results of 20 times as three-dimensional figures.

As shown in Fig. 3, we get the clustering ACC results of UFSRL with respect to different values of α and β. From Fig. 3 we can see that different parameters will lead to different results. At the same time, the results of UFSRL algorithm is relatively stable on *Isolet, BC* and *Prostate-GE* data-sets, while the results are sensitive to parameter on other data-sets. It is easy to understand that the optimal parameter values for different data-sets tend to be different, i.e. the determination of the optimal parameter value depends on the specific property to the data itself. In addition, how to determine the hyper parameter is still an open problem. From the experimental results, we find that the best clustering ACC of each data set is generally in the case of taking a larger β, which is also consistent with the second section mentioned. The parameter β should be large appropriate to ensure the

Table 7 Comparison of computational time (in seconds) and the number of selected features on seven datasets

	LapScore		SPEC		MCFS		JELSR		LSPE		DFSC		UFSRL	
	Time	r	Time	r	Time	r	Time	r	Time	r	Time	r	Time	r
Umist	0.05	12	0.04	42	0.46	35	1.75	50	26.97	150	1.05	180	0.40	3
ORL	0.04	90	0.04	270	0.12	90	1.79	200	15.19	230	0.70	340	2.15	85
Isolet	0.25	60	0.10	300	5.58	300	13.12	200	30.60	150	2.14	300	1.57	300
Ionosphere	0.01	6	0.01	10	0.23	5	0.25	10	0.12	10	0.07	10	0.02	8
BC	0.01	10	0.02	10	0.13	9	1.67	9	0.18	8	0.52	15	0.02	6
Prostate_GE	0.06	270	0.04	300	0.60	6	79.74	60	>500	85	81.53	190	76.36	680
Dbworld-bodies	0.07	800	0.03	380	0.46	75	39.30	500	>500	560	41.75	600	31.16	470
mean	0.07	178	0.04	187	1.08	74	19.66	147	>500	171	18.25	233	15.95	221

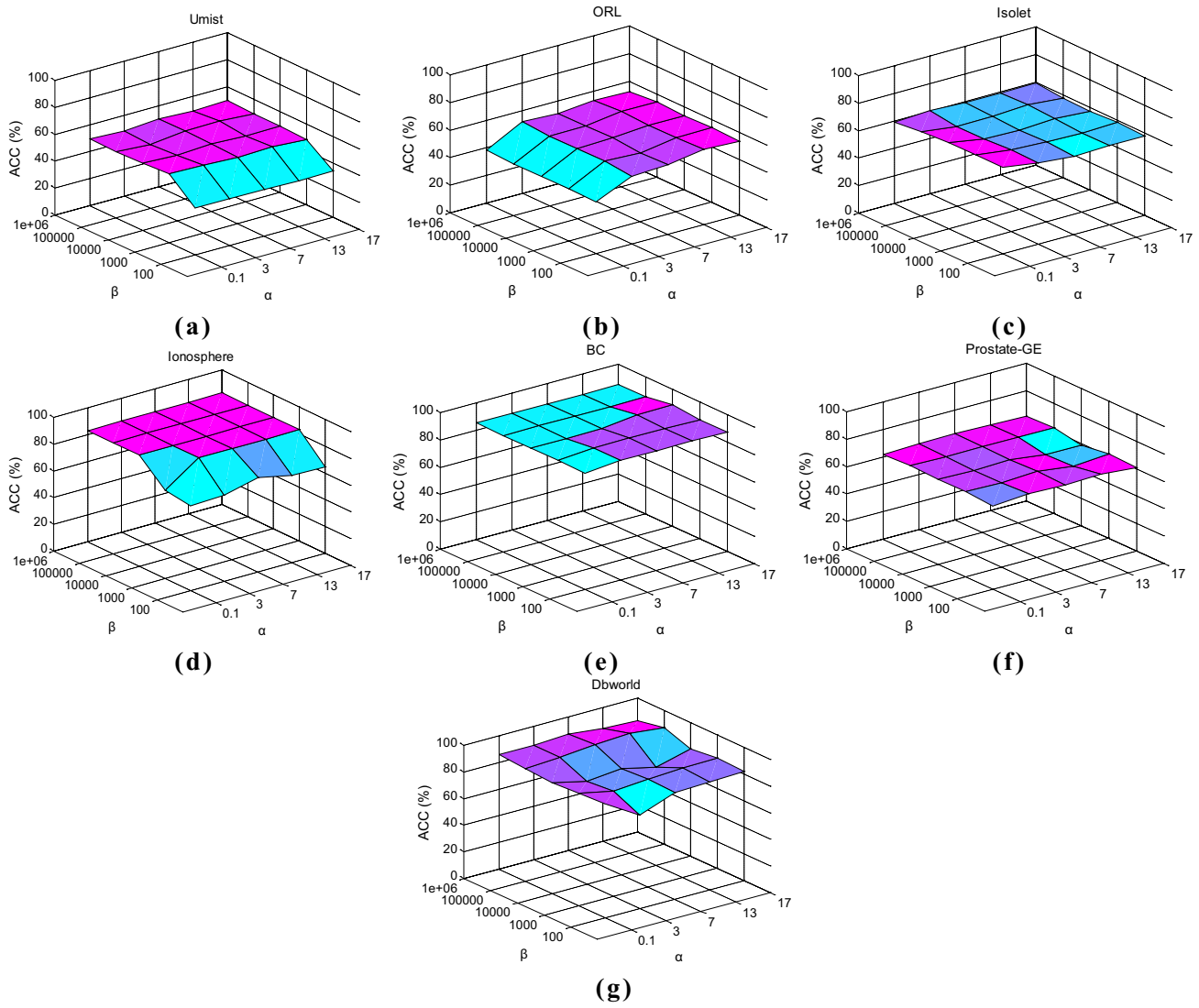

Fig. 3 The ACC of UFSRL with regard to different values of α and β. **a** Umist, **b** ORL, **c** Isolet, **d** Ionosphere, **e** BC, **f** Prostate-GE, **g** Dbworld_bodies

sparseness of the coefficient matrix $\boldsymbol{A}$. The parameter α is used to protect the similarity structure, and the results will be different depending on the variation of α, which indicates that different data-sets need to select the appropriate value of α to achieve the best results.

As shown in Fig. 4, we get the clustering NMI results of UFSRL algorithm with respect to different values of α and β. Overall, the trend of clustering NMI is almost the same as the variation of ACC. For different data-sets, the appropriate parameters α and β are needed to get the best results. The best NMI for each data set also appears in the case of larger values of β. In addition, we have the observation that the best ACC and NMI in the numerical may be a big gap. In the case of the *Prostate-GE* dataset, the value of ACC is around 0.6, and the best NMI value does not exceed 0.1, whereas the ACC is around 0.55 and the NMI is around 0.75 on *ORL* dataset. This has nothing to do with the validity of the proposed algorithm, but because the ACC and NMI are two different evaluation criteria, their results will be different.

5 Conclusion

This paper proposed a new unsupervised feature selection algorithm, named UFSRL. The UFSRL is a joint sparse reconstruction framework based on self-representation, which can preserve the manifold structure of the data while minimizing the reconstruction error. The $l_{2,1/2}$-matrix norm imposed on the coefficient matrix in the proposed algorithm. Regularization constraint can make

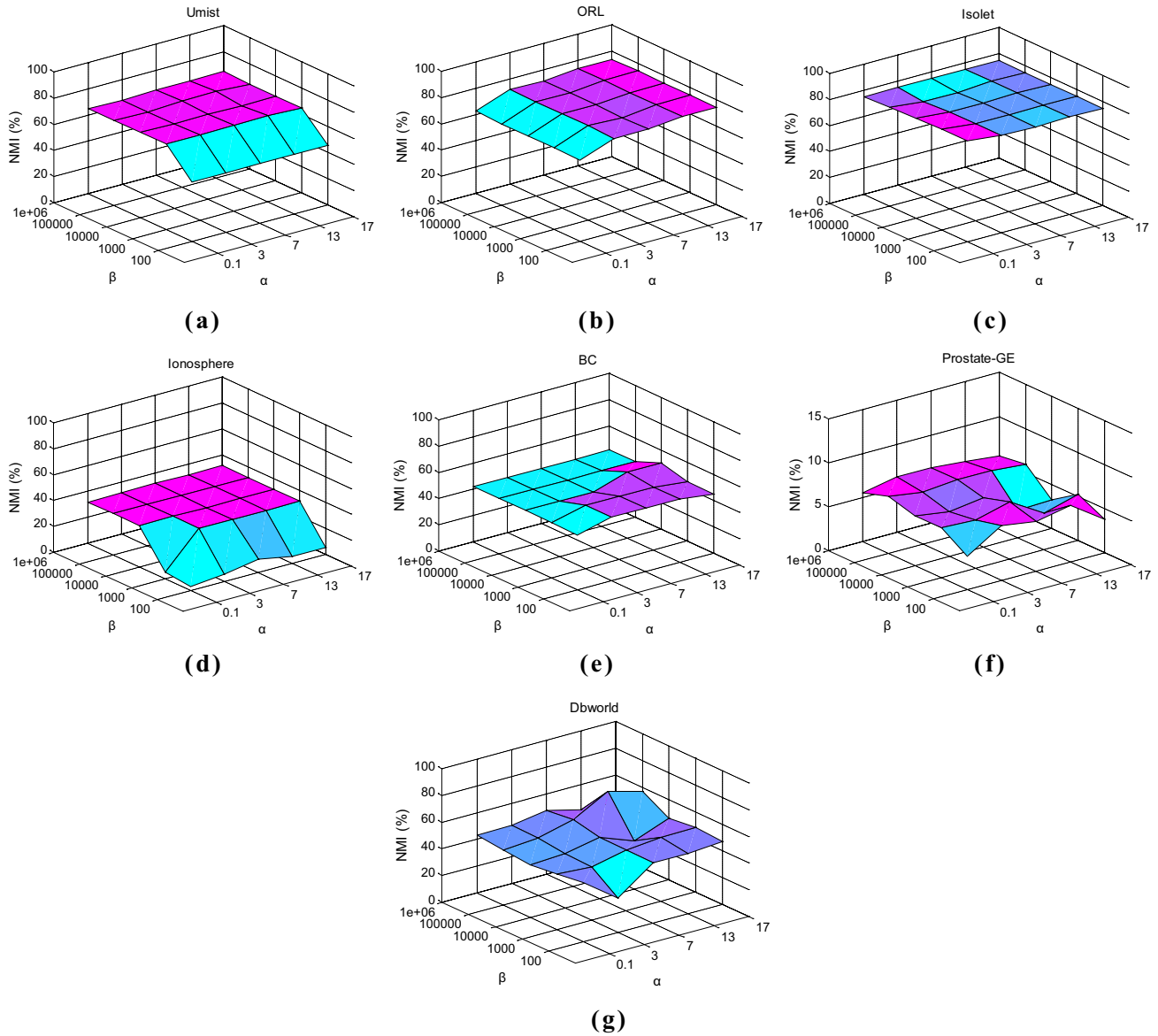

Fig. 4 The NMI of UFSRL with regard to different values of α and β. **a** Umist, **b** ORL, **c** Isolet, **d** Ionosphere, **e** BC, **f** Prostate-GE, **g** Dbworld_bodies

the coefficient matrix have row sparsity, which makes the proposed model more suitable for feature selection and robust to noise. We design an iterative algorithm to solve the objective function of UFSRL model. The proposed UFSRL can effectively identify the most discriminative feature subset of the original data. The effectiveness of UFSRL has been proven in experiments on eight synthetic and real-world data-sets. The experiment results also show that feature selection can not only reduce the dimensionality of high dimensional data but also can improve the quality of learning algorithm. For future work, we will continue to focus on feature selection research, and further improve the effect and parameter robustness of the proposed UFSRL algorithm. Besides, combining feature selection with deep learning to solve the corresponding problem is also one of our directions for future research.

Acknowledgements This work was partially supported by the National Natural Science Foundation of China, under Grants 61773304 and 61371201, the National Basic Research Program (973 Program) of China under Grant 2013CB329402, the Program for Cheung Kong Scholars and Innovative Research Team in University under Grant IRT_15R53.

References

1. Gu B, Sheng VS, Tay KY, Romano W, Li S (2015) Incremental support vector learning for ordinal regression. IEEE Trans Neural Netw Learn Syst 26(7):1403–1416
2. Gu B, Sun XM, Sheng VS (2016) Structural minimax probability machine. IEEE Trans Neural Netw Learn Syst. https://doi.org/10.1109/TNNLS.2016.2544779
3. Tian Q, Chen S (2017) Cross-heterogeneous-database age estimation through correlation representation learning. Neurocomputing 238:286–295
4. Mutch J, Lowe DG (2006) Multiclass object recognition with sparse localized features. In: Proceedings IEEE computer society conference on computer vision pattern recognit, pp 11–18
5. Li Z, Tang J (2015) Unsupervised feature selection via nonnegative spectral analysis and redundancy control. IEEE Trans Image Process 24(12):5343–5355
6. Gu B, Sheng VS (2016) A robust regularization path algorithm for ν-support vector classification. IEEE Trans Neural Netw Learn Syst 28(5):1241–1248
7. Zhu YY, Liang JW, Chen JY, Ming Z (2017) An improved NSGA-III algorithm for feature selection used in intrusion detection. Knowl Based Syst 116:74–85
8. Tang V, Yan H (2012) Noise reduction in microarray gene expression data based on spectral analysis. Int J Mach Learn Cyber 3(1):51–57
9. Gu B, Sheng VS, Wang Z, Ho D, Osman S, Li S (2015) Incremental learning for ν-support vector regression. Neural Netw 67:140–150
10. Wang H, Jing XJ, Niu B (2017) A discrete bacterial algorithm for feature selection in classification of microarray gene expression cancer data. Knowl Based Syst 126:8–19
11. Wang H, Niu B (2017) A novel bacterial algorithm with randomness control for feature selection in classification. Neurocomputing 228:176–186
12. Sharma A, Imoto S, Miyano S, Sharma V (2012) Null space based feature selection method for gene expression data. Int J Mach Learn Cybern 3(4):269–276
13. Xiang S, Nie F, Meng G, Pan C, Zhang C (2012) Discriminative least squares regression for multiclass classification and feature selection. IEEE Trans Neutral Netw Learn Syst 23(11):1738–1754
14. Hu Q, Pan W, An S, Ma P, Wei J (2010) An efficient genes election technique for cancer recognition based on neighborhood mutual information. Int J Mach Learn Cybern 1(1):63–74
15. Yu SQ, Chen HF, Wang Q, Shen LL, Huang YZ (2017) Invariant feature extraction for gait recognition using only one uniform model. Neurocomputing 239:81–93
16. Wan MH, Lai ZH (2017) Feature extraction via sparse difference embedding (SDE). KSII Trans Internet Inf Syst 11(7):3594–3607
17. MartõÂnez AM, Kak AC (2001) PCA versus LDA. IEEE Trans Pattern Anal Mach Intell 23(3):228–233
18. Tao D, Tang X, Li X, Wu X (2006) Asymmetric bagging and random subspace for support vector machines-based relevance feedback in image retrieval. IEEE Trans Pattern Anal Mach Intell 28(7):1088–1099
19. Gui J, Sun Z, Ji S, Tao D, Tan T (2016) Feature selection based on structured sparsity: a comprehensive study. IEEE Trans Neutral Netw Learn Syst 28(7):1490–1507
20. Peng H, Long F, Ding C (2005) Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans Pattern Anal Mach Intell 27(8):1226–1238
21. Xu J, Yang G, Man H, He H (2013) L_1 graph based on sparse coding for feature selection. In: Proceedings of international symposium on neural networks (ISNN), pp 594–601
22. Yang JB, Ong C-J (2012) Feature selection based on sparse imputation. In: Proceedings of international joint conference on neural networks (IJCNN), pp 1–7
23. Weston J, Mukherjee S, Chapelle O, Pontil M, Poggio T, Vapnik V (2000) Feature selection for SVMs. In: Proceedings of advances in neural information processing system, vol 12. Cambridge, pp 526–532
24. Duda RO, Hart PE, Stork DG (2001) Pattern classification, 2nd edn. Wiley, Hoboken
25. Gu Q, Li Z, Han J (2011) Generalized Fisher score for feature selection. In: Proceedings of 27th conference on uncertainty in artificial intelligence, pp 266–273
26. Guyon I, Weston J, Barnhill S, Vapnik V (2002) Gene selection for cancer classification using support vector machines. Mach Learn 46:389–422
27. Liu HW, Sun JG, Liu L, Zhang HJ (2009) Feature selection with dynamic mutual information. Pattern Recog 42(7):1330–1339
28. Martínez Sotoca J, Pla F (2010) Supervised feature selection by clustering using conditional mutual information-based distances. Pattern Recog 43(6):2068–2081
29. Ma ZG, Nie FP, Yang Y, Uijlings JRR, Sebe N (2012) Web image annotation via subspace-sparsity collaborated feature selection. IEEE Trans Multimed 14(4):1021–1030
30. Zhu X, Ghahramani Z, Lafferty JD (2003) Semi-supervised learning using gaussian fields and harmonic functions. In: Proceedings of 20th international conference machine learning, pp 912–919
31. Xu ZL, King IW, Lyu MR, Jin R (2010) Discriminative semi-supervised feature selection via manifold regularization. IEEE Trans Neural Netw 21(7):1033–1047
32. Liu Y, Nie FP, Wu JG, Chen LH (2010) Semi-supervised feature selection based on label propagation and subset selection. In: Proceedings of ICCIA, pp 293–296
33. Cai D, Zhang C, He X (2010) Unsupervised feature selection for multi-cluster data. In: Proceedings of the16th ACM SIGKDD international conference on knowledge discovery and data mining, pp 333–342
34. Tang JL, Liu H (2012) Unsupervised feature selection for linked social media data. In: Proceedings of KDD, pp 904–912
35. Li ZC, Yang Y, Liu J, Zhou XF, Lu HQ (2012) Unsupervised feature selection using nonnegative spectral analysis. In: Proceedings of AAAI, pp 1026–1032
36. Xiang S, Shen X, Ye J (2015) Efficient nonconvex sparse group feature selection via continuous and discrete optimization. Artif Intell 224:28–50
37. Xie Z, Xu Y (2014) Sparse group lasso based uncertain feature selection. Int J Mach Learn Cybern 5(2):201–210
38. Cong Y, Wang S, Liu J, Cao J, Yang Y, Luo J (2015) Deep sparse feature selection for computer aided endoscopy diagnosis. Pattern Recognit 48(3):907–917
39. He X, Cai D, Niyogi P (2005) Laplacian score for feature selection. Adv Neural Inf Process Syst 18:507–514
40. Foucart S, Lai MJ (2008) The sparest solutions of underdetermined linear system by lq-minimization for $0 < q \leq 1$. Appl Comput Harmonic Anal 26(3):395–407
41. Chartrand R (2009) Fast algorithms for nonconvex compressive sensing: MRI reconstruction from very few data. In: Proceedings of IEEE international symposium on biomedical imaging, pp 262–265
42. Nie FP, Huang H, Cai X, Ding C (2010) Efficient and robust feature selection via joint $L_{2,1}$-norms minimization. In: Proceedings of NIPS, pp 1813–1821
43. Wang L, Chen S, Wang Y (2014) A unified algorithm for mixed $l_{2,p}$-minimizations and its application in feature selection. Comput Optim Appl 58(2):409–421

44. Shi CJ, Ruan QQ, An GY, Zhao RZ (2015) Hessian semi-supervised sparse feature selection based on $L_{2,1/2}$-matrix norm. IEEE Trans Mutimed 17(1):16–28
45. Zhu P, Zuo W, Zhang L, Hu Q, Shiu SCK (2015) Unsupervised feature selection by regularized self-representation. Pattern Recognit 48:438–446
46. Zhao Z, Liu H (2007) Spectral feature selection for supervised and unsupervised learning. In: Proceedings of 24th international conference on machine learning, pp 1151–1158
47. Zhao Z, Wang L, Liu H (2010) Efficient spectral feature selection with minimum redundancy. In: Proceedings of 24th AAAI conference on artificial intelligence, pp 673–678
48. Hou C, Nie F, Li X, Yi D, Wu Y (2014) Joint embedding learning and sparse regression: a framework for unsupervised feature selection. IEEE Trans Cybern 44(6):793–804
49. Fang X, Xu Y, Li X, Fan Z, Liu H, Chen Y (2014) Locality and similarity preserving embedding for feature selection. Neurocomputing 128:304–315
50. Shang R, Zhang Z, Jiao L, Liu C, Li Y (2016) Self-representation based dual-graph regularized feature selection clustering. Neurocomputing 171:1242–1253
51. Yan H, Yang J, Yang JY (2016) Robust Joint feature weights learning framework. IEEE Trans Knowl Data Eng 28(5):1327–1339
52. Zhao Z, He XF, Cai D, Zhang LJ, Ng W, Zhuang YT (2016) Graph regularized feature selection with data reconstruction. IEEE Trans Knowl Data Eng 28(3):689–700
53. Lee DD, Seung HS (1999) Learning the parts of objects by non-negative matrix factorization. Nature 401:788–791
54. Liu H, Wu Z, Li X, Cai D, Huang TS (2012) Constrained non-negative matrix factorization for imagine representation. IEEE Trans Pattern Anal Mach Intell 34(7):1299–1311
55. Papadimitriou C, Steiglitz K (1998) Combinatorial optimization: algorithms and complexity. Dover, New York
56. Gibbons J, Dickinson, Chakraborti S (2011) Nonparametric statistical inference. Springer, Berlin