Multimedia Tools and Applications (2023) 82:15305–15326
https://doi.org/10.1007/s11042-022-13903-y

Completed sample correlations and feature dependency-based unsupervised feature selection

Tong Liu[1] · Rongyao Hu[1] · Yongxin Zhu[2]

Received: 28 December 2021 / Revised: 26 April 2022 / Accepted: 12 September 2022 /
Published online: 3 October 2022

Abstract

Sample correlations and feature relations are two pieces of information that are needed to be considered in the unsupervised feature selection, as labels are missing to guide model construction. Thus, we design a novel unsupervised feature selection scheme, in this paper, via considering the completed sample correlations and feature dependencies in a unified framework. Specifically, self-representation dependencies and graph construction are conducted to preserve and select the important neighbors for each sample in a comprehensive way. Besides, mutual information and sparse learning are designed to consider the correlations between features and to remove the informative features, respectively. Moreover, various constraints are constructed to automatically obtain the number of important neighbors and to conduct graph partition for the clustering task. Finally, we test the proposed method and verify the effectiveness and the robustness on eight data sets, comparing with nine state-of-the-art approaches with regard to three evaluation metrics for the clustering task.

Keywords Unsupervised learning · Sample correlation · Unsupervised feature selection · Graph learning · Self-representation · Mutual information · Sparse learning

1 Introduction

For various research fields, i.e., machine learning, pattern recognition, network security, facial expression recognition, education, biology, psychology, medicine, many high-dimensional data are used to describe the complex subject [23, 33, 34, 48, 51].

The high-dimensional data are not easy to process due to the issues, such as “dimension disaster”, the cost of the hardware, and the execution times. Hence, the dimensionality

✉ Rongyao Hu
hurongyao123@gmail.com

[1] Massey University Albany Campus, Auckland 0745, New Zealand

[2] Jinan Laboratory of Applied Nuclear Science, The Institute of High Energy Physics of the Chinese Academy of Sciences, Beijing, China

reduction technology has attracted attentions which can remove noise data as well as find the valuable information by using the predefined rules or constructed models [4].

The main purpose of dimensionality reduction is to search a projection matrix in original high-dimensional data space, obtain the novel data with lower dimensions so that the data become divisible in the low-dimensional feature subspace and remove unnecessary outliers. Generally, the existing dimensionality reduction ways are broadly classified to feature extraction ways and feature selection ways [18, 29]. Feature extraction ways derive useful information from the original data to construct the novel data description, while feature selection ways exploit important information among original data with certain evaluation schemes, removing the influence of noise, outliers and redundant features and, at the same time, preserving the representative features [29].

For the number of obtaining labels in training procedure, feature selection ways contain supervised ways, semi-supervised ways, and unsupervised ways [40, 42]. In supervised ways, labels guide the selection of discriminant features according to certain criteria [21, 44]. Semi-supervised ways select important features using a particularly small percentage of labeled data and the numerous unlabeled data [22, 54]. For unsupervised ways, they employ various score evaluation metrics to exploit features through the uniqueness of features, such as variance [42, 47]. However, labels are difficult to mark by experts and the annotation cost is expensive. By this reason, unsupervised feature selection ways have become the important technology for the unseen and unknown data.

In unsupervised feature selection, many researchers usually embed the manifold regularization method into the feature selection model to find low-dimensional embedding, and apply smoothness to adjacent cluster labels for obtaining more compact data representation [17, 37]. For example, Cai et al. achieved good clustering performance by jointly learning the importance of each feature in different dimensions [7]. Liu et al. proposed a method by combining the global similarity with the local similarity to search the subset of features [26]. Feng et al. used automatic encoder to find the representative features subset, preserving the local correlations among data through the hidden layer [13]. Moreover, the common methods explore the representative features by the conducted model and utilize the selected features to connect with the task learning. In this way, it could easily lead to sub-optimal solution by these two steps, that is, the selected features are not suitable to task learning while task learning is not related to the model construction.

To solve the above issues, we design a novel unsupervised feature selection approach via taking the completed sample correlations and feature dependencies into account in a unified framework. Specifically, we consider sample importance to consider global structure among the samples by using a self-representation way, where each sample can be represented by all samples linearly, and a dynamic graph representation can be learned to consider local structure between the samples during the procedure. Besides, we take the dependencies between the features into account by mutual information, and then, a controllable sparse learning (i.e., $\ell_{2,p}$-norm) is utilized to remove the noisy and redundant features. Moreover, rank constraint and neighbors constraint is designed to conduct graph partition and to search the optimal number of neighbor, respectively, for clustering task and each sample. The final experiments on eight data sets with nine comparison methods test the clustering performance.

We summarize the contributions of the proposed method in the following.

- The proposed method considers completed sample correlations and feature dependencies in a joint unsupervised model, it not only extends the novel method of unsupervised learning, but it also indicates the flexibility of embedded way which can embed different technologies to solve the issues (e.g., guarantee the optimization of model and task simultaneously in the framework of unsupervised learning) for various types of data. Moreover, experiments on eight data sets have verified the validity of the proposed method in comparison to other methods.

- The proposed method provides a reliable unsupervised learning method and interpretable features for task learning. Besides, we design a new optimization algorithm to optimize our proposed method and obtain a global optimal solution.

The flow of the remaining parts is described as: information about related works is given in Section 2. The descriptions of proposed method is showed in Section 3. In Section 4, the optimization process, the convergence analysis, the complexity analysis, and deterministic parameter are discussed. Section 5 shows the experiments' performance. Finally, a conclusion is given in Section 6.

2 Related work

Feature selection focuses on searching a small percentage of features to represent all data, and it is useful to help unseen data to obtain the reliable predict label for the task learning [3, 58]. Many feature selection algorithms have been developed including nature-inspired algorithms [5, 46]. However, since a great expense of obtaining labels in real world, unsupervised learning has become the mainstream way to conduct feature selection.

Generally, the existing unsupervised learning schemes regarding the strategies of feature selection contain three types of categories, i.e., filter-based, wrapper-based and embedded-based [1, 2, 20]. Specifically, the filter-based approaches select the informative feature subset by ranking the scores of features according to the predefined evaluation criteria [36, 38, 45]. For example, Cekik and Uysal proposed a filter-based unsupervised learning method to conduct shot text classification by the rough set theory [8]. Solorio et al. devised a filter-based learning method to recognize mixed features by considering information theory and spectral way [43]. He et al. conducted a graph Laplacian to preserve the local correlations among the data and select the subset of features with larger scores by decomposing the Laplacian matrix [17]. Yao et al. designed the locally linear embedding score to select the important features for the clustering task [52].

Compared with the filter-based schemes, the wrapper-based schemes construct the learner to search features so better features selection performance can be obtained [42]. Zhao et al. proposed a joint architecture considering both supervised and unsupervised, and utilized the graph Laplacian score by taking the correlation of features into account [55]. Chen et al. presented a framework by integrating a cosine measurement and support vector machine to select important features and obtain the classification performance [9]. Nouri-Moghaddam et al. proposed a wrapper feature selection method utilizing the forest optimization method which selects the informative features and extracts the non-dominated solution [32]. Feofanov et al. designed a wrapper feature selection method with partially labeled data, which distributes pseudo labels to unlabeled data and conducts a feature selection genetic method to guarantee sparsity and to remove unimportant features [14]. Hence, the wrapper-based methods can achieve the better classification results by conducting the feature selection method.

However, the filter-based method is heavily dependent on the selected score calculation, so the effect of the score method directly determines the final classification or clustering performance. Besides, the simple learning method in wrapper is hard to solve the data with high dimensions, leading to the consumption of heavy computational resources. Moreover, both the filter and the wrapper methods select the features independently, and it may lead to a sub-optimal solution to model construction for the tasks (e.g., classification or clustering). Consequently, the filter-based and the wrapper-based schemes may not solve the real issues in practice. In order to solve the above issues, the embedded-based method is proposed and show the potential ability for the tasks. This is because the embedded-based methods provides a joint model to combine feature selection with task learning, which have demonstrated better performance for a variety of tasks [28, 39, 42, 53]. For example, Zhu et al. conducted a feature selection combined with coupled dictionary learning, where dictionary learning is used to reconstruct the data and a coefficient matrix is learned to represent feature importance [56]. Zhu et al. [57] used subspace clustering to guide embedded-based unsupervised model via iteratively learning the clustering labels and important features. Zhu et al. proposed a robust unsupervised spectral feature selection which considers the local structure of samples and the global structure of features, and employs the sparse learning (i.e., $\ell_{2,1}$-norm) to remove the redundant features [58]. Liu et al. presented a robust neighborhood embedding method to select the important subset of features by minimizing the error of loss function and the regularization term [27]. Lim and Kim proposed a feature dependency-based unsupervised learning model to maintain the correlations between the features by using information theory [24]. Wu et al. developed an adaptive embedded-based way to search reliable and informative features subset in the intrinsic subspace [50]. Hence, the embedded-based methods demonstrate better learning ability and have easier operability for selecting a reliable subset of features.

3 Method

3.1 Notation

The used symbol markers in Table 1 and the scalar, vector, and matrix is defined by normal italic letter, boldface lower-case, and boldface upper-case letter, respectively, in this section.

Table 1 The used symbol markers

$\mathbf{X}$	A matrix
$\mathbf{x}$	A vector of $\mathbf{X}$
$\mathbf{x}_{i,\cdot}$	The i-th row of $\mathbf{X}$
$\mathbf{x}_{\cdot,j}$	The j-th column of $\mathbf{X}$
$x_{i,j}$	The element in the i-th row and the j-th column of $\mathbf{X}$
$\|\mathbf{X}\|_F$	The Frobenius norm of $\mathbf{X}$, i.e., $\sqrt{\sum_{i,j}\mathbf{x}_{i,j}^2}$
$\|\mathbf{X}\|_{2,p}$	The $\ell_{2,p}$-norm of $\mathbf{X}$, i.e., $(\sum_i\sqrt{\sum_j x_{ij}^2})^{1/p}$
rank($\mathbf{X}$)	The rank of $\mathbf{X}$
$tr(\mathbf{X})$	The trace of $\mathbf{X}$
$\mathbf{X}^T$	The transpose of $\mathbf{X}$
$\mathbf{X}^{-1}$	The inverse of $\mathbf{X}$
$\mathbf{1}$	An all-one-element vector

3.2 Completed sample correlations

Given an original matrix $\mathbf{X} \in \mathbb{R}^{d \times n}$, where n and d, respectively, denote the number of samples and features. The common supervised feature selection is

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

where coefficient weight is $\mathbf{W} \in \mathbb{R}^{d \times c}$, the reduced dimension is c, regularization term is $\|.\|_{2,1}$ and γ is the trade-off positive parameter. However, the labels are costly to obtain in practise due to the huge volume of data limitation as well as the expensive access to labels, so that (1) has a limitation to apply for numerous data in various research topics.

To keep the relations between the samples, sample self-representation is employed to search the global correlations of samples via assuming that each sample has the potential correlations with other parts of samples, that is, we can represent every sample by its important neighbor ones. It is an inherent property of one sample and has shown better performance in machine learning [19, 25]. Specifically, a linear combination way can be conducted to each sample $\mathbf{x}_{\cdot,i}$ in $\mathbf{X}$ and we have a sparse unsupervised feature selection model by

$$\min_{\mathbf{W}} \|\mathbf{W}^T\mathbf{X} - \mathbf{W}^T\mathbf{X}\mathbf{S}\|_F^2 + \gamma\|\mathbf{W}\|_{2,1}, \tag{2}$$

where $\mathbf{S} \in \mathbb{R}^{n*n}$ is self-representation matrix, where each row is defined as $\mathbf{W}^T\mathbf{x}_{\cdot,i} \approx \sum_{j=1}^{n} \mathbf{W}^T\mathbf{x}_{\cdot,j}s_{i,j}$. Equation (2) can obtain dependable self-representation weight matrix and the reasons can be showed as follows, (1) there exist abundant noisy features in the original data set, but $\mathbf{W}^T\mathbf{X}$ can project the original data $\mathbf{X}$ into the low-dimensional subspace, obtaining the "clean" data. (2) It uses the inter-reconstruction representation and nearest neighbors to represent each sample that may cover the manifold structure among the original data set. Obviously, the larger value the $s_{i,j}$ has, the more important the related neighbors are involved in the self-representation weight matrix.

Equation (2) has shown the global relations of samples, but the completed relationship is composed of global representation and local representation. Hence, we describe the local correlation consideration by

$$\min_{\mathbf{W}} \sum_{i,j}^{n} \|\mathbf{W}^T\mathbf{x}_{\cdot,i} - \mathbf{W}^T\mathbf{x}_{\cdot,j}\|_2^2 s_{i,j}, \tag{3}$$

where each element $s_{i,j}$ represents the strength of similarity between two samples (i.e., $\mathbf{x}_{\cdot,i}$ and $\mathbf{x}_{\cdot,j}$) in Euclidean space. That is, the value of $s_{i,j}$ is larger while $\mathbf{x}_{\cdot,j}$ is close to $\mathbf{x}_{\cdot,i}$, otherwise $s_{i,j}$ is small. Besides, a Gaussian kernel is employed to calculate the distance between samples, such as $d(\mathbf{x}_{\cdot,i}, \mathbf{x}_{\cdot,j}) = \exp(-\|\mathbf{x}_{\cdot,i} - \mathbf{x}_{\cdot,j}\|_2^2/2\sigma)$ where σ is an adjustable parameter.

Although (2) or (3) are widely used in the existing unsupervised feature selection works [18, 35], there exist various aspects to improve the above models. (1) Both (2) and (3) have to select the important adjacent samples (i.e., the value of K) in advance since the size of K is essential parameter to model construction, but the predefined identify size K to each sample may lead to capturing inaccurate manifold information among data [11]. For example, if K takes the small value, it is hard to capture the important neighboring information of each sample. In contrast, the selected neighboring samples may contain uncorrelated ones while the vale of K is larger. (2) The parameter σ controls the size of edge weights between each samples and its neighbors in the Gaussian kernel function, but it is a time-consuming process. In this way, it increases the difficulty of model construction as well as rises the time complexity of dealing with big data. (3) As we examine this formula (3) carefully, we find the graph matrix is learned from the original data projection by $\mathbf{W}^T$ in the intrinsic subspace. In other words, graph matrix is dependent on the weight matrix, the reverse is not true.

Motivated by the above discussions, we modify (3) by the following two aspects, (1) a joint model is conducted to learn graph matrix and weight matrix simultaneously, and each of them iteratively optimize itself by alternating mutual iterative optimisation until obtaining their optimal results; (2) the distribution of the samples in the intrinsic subspace are considered to search optimal graph representation, as well as constraints that are designed to find optimal number of neighbors for each sample, rather than employ both K and σ to conduct a fixed graph representation. Hence, we integrate (2) with (3) in a joint structure and design the new objective function as follows,

$$\begin{aligned}&\min_{\mathbf{W},\mathbf{S}}\|\mathbf{W}^T\mathbf{X}-\mathbf{W}^T\mathbf{X}\mathbf{S}\|_F^2+\gamma\|\mathbf{W}\|_{2,1}\\&\quad+\alpha\left(\sum\nolimits_{i,j}^{n}\left(\|\mathbf{W}^T\mathbf{x}_{\cdot,i}-\mathbf{W}^T\mathbf{x}_{\cdot,j}\|_2^2 s_{i,j}+\lambda\|\mathbf{s}_{i,\cdot}\|_2^2\right)\right)\\&s.t., \forall i, \mathbf{s}_i^T\mathbf{1}=1, s_{i,i}=0, s_{i,j}\geq 0, \text{if } j\in\mathcal{N}(i), \text{otherwise } 0,\end{aligned} \tag{4}$$

where α and β are trade-off parameters, $\|\mathbf{s}_{i,\cdot}\|_2^2$ is utilized to avoid trivial solution, $\mathcal{N}(i)$ denotes the nearest neighbors set for i-th sample, and $\mathbf{s}_i^T\mathbf{1}=1$ achieves shift invariant similarity.

Different from the previous unsupervised feature selection approaches [19, 25, 31, 58], our (4) conducts a dynamic graph matrix via considering completed sample correlations (i.e., global message and local information) and an adaptive weight matrix in a joint framework. Obviously, each sample by our method can adaptively select the different number of nearest neighbors and can take global information and local information into account at the same time.

3.3 Feature dependency consideration

An optimal subset of features are learned through all features during feature selection and the corresponding common time complexity is O(2^d). However, it may result in non-trivial problem. By this reason, we convert the combination way into an estimation way, that is, we employ mutual information [36] to consider the feature dependencies rather than use the iterative feature combination approach. Specifically, according to the definition of mutual information in information theory, the joint distribution and the marginal distribution are two basic variables. Then, mutual information is the relative entropy between joint distribution and marginal distribution. Hence, we consider mutual information between features that can be represented by

$$MI(\mathbf{x}_{\cdot,i};\mathbf{x}_{\cdot,j})=F(\mathbf{x}_{\cdot,i})+F(\mathbf{x}_{\cdot,j})-F(\mathbf{x}_{\cdot,i},\mathbf{x}_{\cdot,j}), \tag{5}$$

where $F(\mathbf{x}_{\cdot,i},\mathbf{x}_{\cdot,j}) = -\sum_i^d\sum_j^d P(\mathbf{x}_{\cdot,i},\mathbf{x}_{\cdot,j})\log\frac{P(\mathbf{x}_{\cdot,i},\mathbf{x}_{\cdot,j})}{P(\mathbf{x}_{\cdot,i})P(\mathbf{x}_{\cdot,j})}$ denotes the joint entropy, $F(\mathbf{x}_{\cdot,i})$ and $F(\mathbf{x}_{\cdot,j})$ denote the marginal entropy.

Due to the mutual information representing degree of interdependence between features, we calculate the dependence matrix $\mathbf{Q}$ to preserve the feature dependency and it can be described as

$$Q_{i,j}=MI(\mathbf{x}_{\cdot,i};\mathbf{x}_{\cdot,j}),\ \ if\ \ i\neq j;\ \ Q_{i,j}=\sum\nolimits_{i\neq k}Q_{i,k}, otherwise. \tag{6}$$

where $\mathbf{Q}$ denotes the dependencies between features and selects the independent features. It emphasizes the feature dependencies by its adjacent neighbor using mutual information i.e., $i = j$, otherwise, it only calculates mutual information between features. Furthermore, we conduct one regularization term $tr(\mathbf{W}^T\mathbf{QW})$ embedding into the minimization problem in (4), since a positive semi-definite matrix $\mathbf{Q}$ which would be tractable in the optimization process and can be updated by the change of adjacent information.

3.4 Proposed method

The learned graph $\mathbf{S}$ comes from both sample self-representation and intrinsic space spanned by $\mathbf{W}^T\mathbf{X}$. Moreover, graph matrix $\mathbf{S}$ guides weight matrix learning $\mathbf{W}$ and feature dependency matrix construction $\mathbf{Q}$, but the three matrices are not known in advance. This paper integrates completed sample correlations in (4) with feature dependency consideration in a unified framework to solve this problem, and designs the objective function as

$$\begin{aligned}&\min_{\mathbf{W},\mathbf{S}}\|\mathbf{W}^T\mathbf{X}-\mathbf{W}^T\mathbf{XS}\|_F^2+\beta tr(\mathbf{W}^T\mathbf{QW})+\gamma\|\mathbf{W}\|_{2,1}\\&\quad+\alpha\sum\nolimits_{i,j}^{n}(\|\mathbf{W}^T\mathbf{x}_{\cdot,i}-\mathbf{W}^T\mathbf{x}_{\cdot,j}\|_2^2 s_{i,j}+\lambda\|\mathbf{s}_{i,\cdot}\|_2^2)\\&s.t., \forall i, \mathbf{s}_i^T\mathbf{1}=1, s_{i,i}=0, s_{i,j}\geq 0, \text{if } j\in\mathcal{N}(i), \text{otherwise } 0.\end{aligned}\tag{7}$$

Although sparsity-based term $\|\mathbf{W}\|_{2,1}$ can select informative features by ranking the importance of features, it is not flexible to control the degree of features. Thus, we employ $\ell_{2,p}$-norm to select important features which may share more structures when p is small. Besides, low-rank constraint is embedded into graph $\mathbf{S}$ to preserve the connection with the task learning. Moreover, there are three parameters in (7) that need to be adjusted that increase the time complexity and decrease the model execution efficiency, so we employ a parameter-free way to release the parameters on regularization term $tr(\mathbf{W}^T\mathbf{QW})$. As the main diagonal element in $\mathbf{Q}$ is connected with the graph matrix $\mathbf{S}$, and other elements are not affected, so feature dependency matrix construction $\mathbf{Q}$ does not need to be updated as an independent variable in the optimization process. Therefore, we have

$$\begin{aligned}&\min_{\mathbf{W},\mathbf{S}}\|\mathbf{W}^T\mathbf{X}-\mathbf{W}^T\mathbf{XS}\|_F^2+\beta tr(\mathbf{W}^T\mathbf{QW})+\gamma\|\mathbf{W}\|_{2,p}\\&\quad+\alpha\sum\nolimits_{i,j}^{n}(\|\mathbf{W}^T\mathbf{x}_{\cdot,i}-\mathbf{W}^T\mathbf{x}_{\cdot,j}\|_2^2 s_{i,j}+\lambda\|\mathbf{s}_{i,\cdot}\|_2^2)\\&s.t., \forall i, \mathbf{s}_i^T\mathbf{1}=1, s_{i,i}=0, s_{i,j}\geq 0,\\&\quad\text{if } j\in\mathcal{N}(i), \text{otherwise } 0,\quad rank(\mathbf{L}_S)=n-c,\end{aligned}\tag{8}$$

where $\beta=\sqrt{1/tr(\mathbf{W}^T\mathbf{QW})}$ can be automatic updated, p denotes a parameter for adjusting the sparsity of features, $\mathbf{S}$ learns both global and local information as well as extract c connected components for the final clustering task. $rank(\mathbf{L}_S)$ denote the i-th smallest eigenvalues for $\mathbf{L}_S$ and $\mathbf{L}_S$ denote a positive semi-definite matrix and $rank(\mathbf{L}_S)\geq 0$. Moreover, according to the theorem of Ky Fan [12] and the definition of $rank(\mathbf{L}_S)$ in [30], we convert (8) into the following objective function

$$\begin{aligned}&\min_{\mathbf{W},\mathbf{S},\mathbf{F}}\|\mathbf{W}^T\mathbf{X}-\mathbf{W}^T\mathbf{XS}\|_F^2+\beta tr(\mathbf{W}^T\mathbf{QW})\\&\quad+\alpha\sum\nolimits_{i,j}^{n}(\|\mathbf{W}^T\mathbf{x}_{\cdot,i}-\mathbf{W}^T\mathbf{x}_{\cdot,j}\|_2^2 s_{i,j}+\lambda\|\mathbf{s}_{i,\cdot}\|_2^2)\\&\quad+\gamma\|\mathbf{W}\|_{2,p}+\delta tr(\mathbf{FL}_S\mathbf{F}^T)\\&s.t., \forall i, \mathbf{s}_i^T\mathbf{1}=1, s_{i,i}=0, s_{i,j}\geq 0, \text{if } j\in\mathcal{N}(i), \text{otherwise } 0,\end{aligned}\tag{9}$$

where the parameter δ controls the importance of regularization term. To decrease the number of parameters, we utilize the parameter-free way to release this parameter and set it to $\delta = \sqrt{1/tr(\mathbf{F}L_S\mathbf{F}^T)}$.

4 Model analysis

For model analysis, we give a detail discussion of the proposed method from four aspects, i.e., optimization, convergence analysis, complexity analysis, and deterministic parameter λ.

4.1 Optimization

Equation (9) is not jointly convex for all variables (i.e., **W**, **S**, and **F**), but is convex for each variable while fixing others. By observing this, the alternative optimization scheme is employed to optimize (9) until the proposed model converges. The pseudo description can be found in the Algorithm 1.

```
Input: $\mathbf{X} \in \mathbb{R}^{d\times n}$, cluster number $c$, parameters $\alpha, \gamma, p$;
Output: $\mathbf{S} \in \mathbb{R}^{n\times n}$ with $c$ blocks;
1. Initialize $\mathbf{S} \in \mathbb{R}^{n\times n}$, $\mathbf{W} \in \mathbb{R}^{d\times c}$, $\mathbf{Q} \in \mathbb{R}^{d\times d}$;
2. Repeat:
   2.1 Optimize $\mathbf{F}$ via (10);
   2.2 Update $\delta$ via $\sqrt{1/tr(\mathbf{F}L_S\mathbf{F}^T)}$;
   2.3 Optimize $\mathbf{S}$ via (19);
   2.4 Update $\lambda$ via (28);
   2.5 Optimize $\mathbf{Q}$ via (6) based on the neighbors distribution in $\mathbf{S}$;
   2.6 Update $\beta$ via $\sqrt{1/tr(\mathbf{W}^T\mathbf{Q}\mathbf{W})}$;
   2.7 Optimize $\mathbf{W}$ via (14);
  Until The iteration result between the objective function value in (9) is less than $10^{-5}$.
```

Algorithm 1 The pseudo of the proposed model in (9).

(i) Optimize **F** via fixing **W** and **S**

After fixing **W** and **S**, (9) can be converted into

$$\min_{\mathbf{F}} \delta tr(\mathbf{F}\mathbf{L}_S\mathbf{F}^T). \tag{10}$$

The optimal **F** can be obtained by conducting eigenvalue decomposition to $\mathbf{L}_S$ and selecting all eigenvectors relating to the smallest c non-zero eigenvalues.

(ii) Optimize **W** via fixing **S** and **F**

While **S** and **F** are fixed, the optimization process of **W** is convex without non-smooth due to using $\ell_{2,p}$-norm. Thus, we use the Iteratively Reweighted Least Square (IRLS) [10] to optimize **W** until the convergence condition is satisfied.

Then, the (9) about $\mathbf{W}$ can be changed to

$$\begin{aligned}&\min_{\mathbf{W}}\|\mathbf{W}^T\mathbf{X}-\mathbf{W}^T\mathbf{X}\mathbf{S}\|_F^2+\beta tr(\mathbf{W}^T\mathbf{Q}\mathbf{W})\\&\quad+\alpha\sum\nolimits_{i,j}^{n}\|\mathbf{W}^T\mathbf{x}_{\cdot,i}-\mathbf{W}^T\mathbf{x}_{\cdot,j}\|_2^2 s_{i,j}+\gamma\|\mathbf{W}\|_{2,p}.\end{aligned}\tag{11}$$

By utilizing the IRLS scheme, (11) is equal to

$$\begin{aligned}&\min_{\mathbf{W}} tr(\mathbf{W}^T\mathbf{X}-\mathbf{W}^T\mathbf{X}\mathbf{S})^T tr(\mathbf{W}^T\mathbf{X}-\mathbf{W}^T\mathbf{X}\mathbf{S})\\&\quad+\beta tr(\mathbf{W}^T\mathbf{Q}\mathbf{W})+\alpha tr(\mathbf{W}^T\mathbf{X}\mathbf{L}_S\mathbf{X}^T\mathbf{W})\\&\quad+\gamma tr(\mathbf{W}^T\mathbf{M}\mathbf{W}),\end{aligned}\tag{12}$$

where $\mathbf{L}_S=\mathbf{D}_S-\mathbf{S}\in\mathbb{R}^{n\times n}$ denote a Laplacian matrix, $\mathbf{D}_S$ denote a diagonal matrix $\mathbf{D}_{Si,i}=\sum_j^n s_{i,j}$, and then each element in diagonal matrix $\mathbf{M}\in\mathbb{R}^{d\times d}$ is described as

$$m_{i,i}=\frac{1}{(2/p)(\|\mathbf{m}_{i,\cdot}\|_2)^{2-p}}.\tag{13}$$

As each term in (12) is convex, and we change it to

$$\begin{aligned}&\min_{\mathbf{W}} tr(\mathbf{W}^T((\mathbf{X}-\mathbf{X}\mathbf{S})(\mathbf{X}-\mathbf{X}\mathbf{S})^T\\&\quad+\beta\mathbf{Q}+\alpha(\mathbf{X}\mathbf{L}_S\mathbf{X})+\gamma\mathbf{M})\mathbf{W}).\end{aligned}\tag{14}$$

The optimal solution of $\mathbf{W}$ can be calculated to select c eigenvectors of $((\mathbf{X}-\mathbf{X}\mathbf{S})(\mathbf{X}-\mathbf{X}\mathbf{S})^T+\beta\mathbf{Q}+\alpha(\mathbf{X}\mathbf{L}_S\mathbf{X})+\gamma\mathbf{M})$ relating to c smallest non-zero eigenvalues.

(iii) Optimize $\mathbf{S}$ via fixing $\mathbf{W}$ and $\mathbf{F}$

When $\mathbf{W}$ and $\mathbf{F}$ are fixed, we have a formula about $\mathbf{S}$ as follows,

$$\begin{aligned}&\min_{\mathbf{S}}\|\mathbf{W}^T\mathbf{X}-\mathbf{W}^T\mathbf{X}\mathbf{S}\|_F^2+\delta tr(\mathbf{F}\mathbf{L}_S\mathbf{F}^T)\\&\quad+\alpha\sum\nolimits_{i,j}^{n}(\|\mathbf{W}^T\mathbf{x}_{\cdot,i}-\mathbf{W}^T\mathbf{x}_{\cdot,j}\|_2^2 s_{i,j}+\lambda\|\mathbf{s}_{i,\cdot}\|_2^2)\\&s.t., \forall i, \mathbf{s}_i^T\mathbf{1}=1, s_{i,i}=0, s_{i,j}\geq 0, \text{if } j\in\mathcal{N}(i), \text{otherwise } 0.\end{aligned}\tag{15}$$

According to the definition of $tr(\mathbf{F}\mathbf{L}_S\mathbf{F}^T)$ in [30], and we have

$$\begin{aligned}&\min_{\mathbf{S}}\sum\nolimits_{i,j}^{n}\|\mathbf{B}_{\cdot,i}-\mathbf{B}\mathbf{S}_{\cdot,j}\|_2^2+\delta\|\mathbf{f}_{\cdot,i}-\mathbf{f}_{\cdot,j}\|_2^2 s_{i,j}\\&\quad+\alpha(\|\mathbf{W}^T\mathbf{x}_{\cdot,i}-\mathbf{W}^T\mathbf{x}_{\cdot,j}\|_2^2 s_{i,j}+\lambda\mathbf{s}_{i,j}^2)\\&s.t., \forall i, \mathbf{s}_i^T\mathbf{1}=1, s_{i,i}=0, s_{i,j}\geq 0, \text{if } j\in\mathcal{N}(i), \text{otherwise } 0,\end{aligned}\tag{16}$$

where $\mathbf{B}=\mathbf{W}^T\mathbf{X}$.

Note that each i in (16) is independent by each other, and we define $e_{i,j} = \alpha\|\mathbf{W}^T\mathbf{x}_{\cdot,i} - \mathbf{W}^T\mathbf{x}_{\cdot,j}\|_2^2 + \delta\|\mathbf{f}_{\cdot,i} - \mathbf{f}_{\cdot,j}\|_2^2$ so (16) is formulated as

$$\min_{\mathbf{s}_i^T\mathbf{1}=1, s_{i,i}=0, s_{i,j}\geq 0} \|\mathbf{s}_{i,\cdot} - \mathbf{e}_{i,\cdot}\|_2^2. \tag{17}$$

The Lagrangian function of (17) is

$$\|\mathbf{s}_{i,\cdot} - \mathbf{e}_{i,\cdot}\|_2^2 - \Delta_1(\mathbf{s}_i^T\mathbf{1} - 1) - \Delta_2 s_{i,j}, \tag{18}$$

where Δ_1 and Δ_2 denote the Lagrangian multipliers. By using the Karush-Kuhn-Tucker (KKT) conditions [6] and calculating the derivative on (18) regarding to $\mathbf{s}_{i,j}$, we obtain the closed-form solution of $\mathbf{s}_{i,j} = \max(\mathbf{e}_{i,j} + \Delta_1, 0)$. After ranking the element in $\mathbf{e}_{i,\cdot}$ with descending order, we achieve the newly sorted vector $\hat{\mathbf{e}}_{i,\cdot} = [\hat{e}_{i,1}, ..., \hat{e}_{i,n}]$. Moreover, due to the constraint $\mathbf{s}_i^T\mathbf{1} = 1$ and each data point is represented by it K-nearest neighbors, the closed-form solution of $\mathbf{s}_{i,j},\quad j = 1, ..., n$ is obtained as

$$\mathbf{s}_{i,j} = \hat{\mathbf{e}}_{i,j} + \frac{1}{K} - \frac{1}{K}\sum_{j}^{K}\hat{\mathbf{e}}_{i,j}, \text{ if } j \in \mathcal{N}(i), \ otherwise\ 0. \tag{19}$$

4.2 Convergence analysis

We discuss the convergence of the proposed method in Algorithm 1, and we first employ the Lemma as follows,

Lemma 1 *The inequality*

$$\sqrt{v} - \frac{v}{2\sqrt{u}} \leq \sqrt{u} - \frac{u}{2\sqrt{u}} \tag{20}$$

always hold for all positive real number of u and v [49].

Then, we design a new Theorem 1 to prove the convergence of Algorithm 1.

Theorem 1 *The objective function value of* (8) *monotonically decreases until Algorithm 1 converges.*

Proof After the t-th iteration, we have obtained the current optimal $\mathbf{F}^{(t)}$, $\mathbf{W}^{(t)}$ and $\mathbf{S}^{(t)}$ in (9). In the (t+1)-th iteration, we have need to optimize $\mathbf{F}^{(t+1)}$ by fixing $\mathbf{W}^{(t+1)}$ and $\mathbf{S}^{(t+1)}$.

According to (10), $\mathbf{F}^{(t+1)}$ has a closed-form solution, and thus we have the inequality as follows

$$\begin{aligned}
&\|\mathbf{W}^{(t+1)^T}\mathbf{X} - \mathbf{W}^{(t+1)^T}\mathbf{X}\mathbf{S}^{(t+1)}\|_F^2 \\
&\quad+\alpha\sum_{i,j}^{n}(\|\mathbf{W}^{(t+1)^T}\mathbf{x}_{\cdot,i} - \mathbf{W}^{(t+1)^T}\mathbf{x}_{\cdot,j}\|_2^2 s_{i,j}^{(t+1)} \\
&\quad+\lambda\|\mathbf{s}_{i,\cdot}^{(t+1)}\|_2^2) + \beta tr(\mathbf{W}^{(t+1)^T}\mathbf{Q}\mathbf{W}^{(t+1)}) \\
&\quad+\delta tr(\mathbf{F}^{(t+1)}\mathbf{L}_S\mathbf{F}^{(t+1)^T}) + \gamma\|\mathbf{W}^{(t+1)}\|_{2,p}
\end{aligned}$$

$$\begin{aligned}&\leq \|\mathbf{W}^{(t+1)^T}\mathbf{X}-\mathbf{W}^{(t+1)^T}\mathbf{X}\mathbf{S}^{(t+1)}\|_F^2\\&\quad+\alpha\sum\nolimits_{i,j}^{n}(\|\mathbf{W}^{(t+1)^T}\mathbf{x}_{\cdot,i}-\mathbf{W}^{(t+1)^T}\mathbf{x}_{\cdot,j}\|_2^2 s_{i,j}^{(t+1)}\\&\quad+\lambda\|\mathbf{s}_{i,\cdot}^{(t+1)}\|_2^2)+\beta tr(\mathbf{W}^{(t+1)^T}\mathbf{Q}\mathbf{W}^{(t+1)})\\&\quad+\delta tr(\mathbf{F}^{(t)}\mathbf{L}_S\mathbf{F}^{(t)^T})+\gamma\|\mathbf{W}^{(t+1)}\|_{2,p}\end{aligned}\tag{21}$$

When fixing $\mathbf{F}^{(t)}$ and $\mathbf{W}^{(t+1)}$ to update $\mathbf{S}^{(t+1)}$, $s_{i,j}^{(t+1)}$ has a closed-form solution according to (19), i.e., global optimal solution, for all $i, j = 1, ..., n$, and thus the following inequality we have is

$$\begin{aligned}&\|\mathbf{W}^{(t+1)^T}\mathbf{X}-\mathbf{W}^{(t+1)^T}\mathbf{X}\mathbf{S}^{(t+1)}\|_F^2\\&\quad+\alpha\sum\nolimits_{i,j}^{n}(\|\mathbf{W}^{(t+1)^T}\mathbf{x}_{\cdot,i}-\mathbf{W}^{(t+1)^T}\mathbf{x}_{\cdot,j}\|_2^2 s_{i,j}^{(t+1)}\\&\quad+\lambda\|\mathbf{s}_{i,\cdot}^{(t+1)}\|_2^2)+\beta tr(\mathbf{W}^{(t+1)^T}\mathbf{Q}\mathbf{W}^{(t+1)})\\&\quad+\delta tr(\mathbf{F}^{(t)}\mathbf{L}_S\mathbf{F}^{(t)^T})+\gamma\|\mathbf{W}^{(t+1)}\|_{2,p}\\&\leq \|\mathbf{W}^{(t+1)^T}\mathbf{X}-\mathbf{W}^{(t+1)^T}\mathbf{X}\mathbf{S}^{(t)}\|_F^2\\&\quad+\alpha\sum\nolimits_{i,j}^{n}(\|\mathbf{W}^{(t+1)^T}\mathbf{x}_{\cdot,i}-\mathbf{W}^{(t+1)^T}\mathbf{x}_{\cdot,j}\|_2^2 s_{i,j}^{(t)}\\&\quad+\lambda\|\mathbf{s}_{i,\cdot}^{(t)}\|_2^2)+\beta tr(\mathbf{W}^{(t+1)^T}\mathbf{Q}\mathbf{W}^{(t+1)})\\&\quad+\delta tr(\mathbf{F}^{(t)}\mathbf{L}_S\mathbf{F}^{(t)^T})+\gamma\|\mathbf{W}^{(t+1)}\|_{2,p}\end{aligned}\tag{22}$$

When fixing $\mathbf{F}^{(t)}$ and $\mathbf{S}^{(t)}$ to update $\mathbf{W}^{(t+1)}$, and according to (14), (22) can be rewritten by

$$\begin{aligned}&\|\mathbf{W}^{(t+1)^T}\mathbf{X}-\mathbf{W}^{(t+1)^T}\mathbf{X}\mathbf{S}^{(t)}\|_F^2\\&\quad+\alpha\sum\nolimits_{i,j}^{n}(\|\mathbf{W}^{(t+1)^T}\mathbf{x}_{\cdot,i}-\mathbf{W}^{(t+1)^T}\mathbf{x}_{\cdot,j}\|_2^2 s_{i,j}^{(t)}\\&\quad+\lambda\|\mathbf{s}_{i,\cdot}^{(t)}\|_2^2)+\beta tr(\mathbf{W}^{(t+1)^T}\mathbf{Q}\mathbf{W}^{(t+1)})\\&\quad+\delta tr(\mathbf{F}^{(t)}\mathbf{L}_S\mathbf{F}^{(t)^T})+\gamma\sum_{i}^{c}\frac{(\|\mathbf{w}_{\cdot,i}^{(t+1)}\|_2)^{2(2-p)}}{(2/p)(\|\mathbf{w}_{\cdot,i}^{(t)}\|_2)^{(2-p)}}\\&\leq \|\mathbf{W}^{(t)^T}\mathbf{X}-\mathbf{W}^{(t)^T}\mathbf{X}\mathbf{S}^{(t)}\|_F^2\\&\quad+\alpha\sum\nolimits_{i,j}^{n}(\|\mathbf{W}^{(t)^T}\mathbf{x}_{\cdot,i}-\mathbf{W}^{(t)^T}\mathbf{x}_{\cdot,j}\|_2^2 s_{i,j}^{(t)}\\&\quad+\lambda\|\mathbf{s}_{i,\cdot}^{(t)}\|_2^2)+\beta tr(\mathbf{W}^{(t)^T}\mathbf{Q}\mathbf{W}^{(t)})\\&\quad+\delta tr(\mathbf{F}^{(t)}\mathbf{L}_S\mathbf{F}^{(t)^T})+\gamma\sum_{i}^{c}\frac{(\|\mathbf{w}_{\cdot,i}^{(t)}\|_2)^{2(2-p)}}{(2/p)(\|\mathbf{w}_{\cdot,i}^{(t)}\|_2)^{(2-p)}}\end{aligned}\tag{23}$$

According to Lemma 1 and for each i, we obtain

$$(\|\mathbf{w}_{\cdot,i}^{(t+1)}\|_2)^{(2-p)}-\frac{(\|\mathbf{w}_{\cdot,i}^{(t+1)}\|_2)^{2(2-p)}}{(2/p)(\|\mathbf{w}_{\cdot,i}^{(t)}\|_2)^{(2-p)}}\leq(\|\mathbf{w}_{\cdot,i}^{(t)}\|_2)^{(2-p)}-\frac{(\|\mathbf{w}_{\cdot,i}^{(t)}\|_2)^{2(2-p)}}{(2/p)(\|\mathbf{w}_{\cdot,i}^{(t)}\|_2)^{(2-p)}}\tag{24}$$

By integrating (24) with (23) and for all c, we obtain

$$\begin{aligned}
&\|\mathbf{W}^{(t+1)^T}\mathbf{X} - \mathbf{W}^{(t+1)^T}\mathbf{X}\mathbf{S}^{(t)}\|_F^2 \\
&\quad +\alpha\sum\nolimits_{i,j}^{n}(\|\mathbf{W}^{(t+1)^T}\mathbf{x}_{\cdot,i} - \mathbf{W}^{(t+1)^T}\mathbf{x}_{\cdot,j}\|_2^2 s_{i,j}^{(t)} \\
&\quad +\lambda\|\mathbf{s}_{i,\cdot}^{(t)}\|_2^2) + \beta tr(\mathbf{W}^{(t+1)^T}\mathbf{Q}\mathbf{W}^{(t+1)}) \\
&\quad +\delta tr(\mathbf{F}^{(t)}\mathbf{L}_S\mathbf{F}^{(t)^T}) + \gamma\|\mathbf{W}^{(t+1)}\|_{2,p} \\
&\leq \|\mathbf{W}^{(t)^T}\mathbf{X} - \mathbf{W}^{(t)^T}\mathbf{X}\mathbf{S}^{(t)}\|_F^2 \\
&\quad +\alpha\sum\nolimits_{i,j}^{n}(\|\mathbf{W}^{(t)^T}\mathbf{x}_{\cdot,i} - \mathbf{W}^{(t)^T}\mathbf{x}_{\cdot,j}\|_2^2 s_{i,j}^{(t)} \\
&\quad +\lambda\|\mathbf{s}_{i,\cdot}^{(t)}\|_2^2) + \beta tr(\mathbf{W}^{(t)^T}\mathbf{Q}\mathbf{W}^{(t)}) \\
&\quad +\delta tr(\mathbf{F}^{(t)}\mathbf{L}_S\mathbf{F}^{(t)^T}) + \gamma\|\mathbf{W}^{(t)}\|_{2,p}
\end{aligned} \tag{25}$$

After combining (21), (22) with (25), we finally have

$$\begin{aligned}
&\|\mathbf{W}^{(t+1)^T}\mathbf{X} - \mathbf{W}^{(t+1)^T}\mathbf{X}\mathbf{S}^{(t+1)}\|_F^2 \\
&\quad +\alpha\sum\nolimits_{i,j}^{n}(\|\mathbf{W}^{(t+1)^T}\mathbf{x}_{\cdot,i} - \mathbf{W}^{(t+1)^T}\mathbf{x}_{\cdot,j}\|_2^2 s_{i,j}^{(t+1)} \\
&\quad +\lambda\|\mathbf{s}_{i,\cdot}^{(t+1)}\|_2^2) + \beta tr(\mathbf{W}^{(t+1)^T}\mathbf{Q}\mathbf{W}^{(t+1)}) \\
&\quad +\delta tr(\mathbf{F}^{(t+1)}\mathbf{L}_S\mathbf{F}^{(t+1)^T}) + \gamma\|\mathbf{W}^{(t+1)}\|_{2,p} \\
&\leq \|\mathbf{W}^{(t)^T}\mathbf{X} - \mathbf{W}^{(t)^T}\mathbf{X}\mathbf{S}^{(t)}\|_F^2 \\
&\quad +\alpha\sum\nolimits_{i,j}^{n}(\|\mathbf{W}^{(t)^T}\mathbf{x}_{\cdot,i} - \mathbf{W}^{(t)^T}\mathbf{x}_{\cdot,j}\|_2^2 s_{i,j}^{(t)} \\
&\quad +\lambda\|\mathbf{s}_{i,\cdot}^{(t)}\|_2^2) + \beta tr(\mathbf{W}^{(t)^T}\mathbf{Q}\mathbf{W}^{(t)}) \\
&\quad +\delta tr(\mathbf{F}^{(t)}\mathbf{L}_S\mathbf{F}^{(t)^T}) + \gamma\|\mathbf{W}^{(t)}\|_{2,p}
\end{aligned} \tag{26}$$

According to (26), we prove that Algorithm 1 can converge to the optimal solutions. □

4.3 Complexity analysis

For every iteration, the time cost of Algorithm 1 focuses on the computation cost to $\mathbf{L}_S$, $((\mathbf{X}-\mathbf{X}\mathbf{S})(\mathbf{X}-\mathbf{X}\mathbf{S})^T+\beta\mathbf{Q}+\alpha(\mathbf{X}\mathbf{L}_S\mathbf{X})+\gamma\mathbf{M})$, the expression in (19), and the relating time complexity are $O(cn^2)$, $O(cd^2)$, and $O(nd^2)$ where n, d, and c, respectively, denote the number of the samples, the features, the clusters. Hence, the time complexity of Algorithm 1 is $max\{O(cn^2), O(nd^2)\}$ where $n, d > c$.

4.4 Deterministic parameter λ

K-nearest neighbors (KNN) is the main way to guide the graph matrix construction, and this also determines the parameter λ value. In detail, we reorder $\tilde{\mathbf{e}}_{i,\cdot}$ in (19) and obtain new vector $\tilde{\mathbf{e}}'_{i,\cdot} = [\tilde{e}'_{i,1}, ..., \tilde{e}'_{i,n}]$ after ranking all elements with descending order. By following the KNN way to reconstruct each sample, $s_{i,k+1} = 0$ can be obtained. Therefore, we have

$$\mathbf{s}_{i,j} - \frac{\tilde{e}'_{i,j}}{2\lambda} + \frac{1}{K} - \frac{1}{K}\sum\nolimits_{j}^{K}(\mathbf{s}_{i,j} - \frac{\tilde{e}'_{i,j}}{2\lambda}) = 0 \tag{27}$$

and we obtain the optimal value of λ by

$$\lambda = \frac{K\tilde{e}'_{i,j} - \sum_j^K \tilde{e}'_{i,j}}{2(1 + K\mathbf{s}_{i,j} - \sum_j^K \mathbf{s}_{i,j})} \tag{28}$$

5 Experiments

We introduce the source of eight data sets, and discuss nine comparison methods in detail. Then, the related experimental setting is provided to all methods. Moreover, experimental results on all data sets and the experimental analysis are discussed.

5.1 Data sets

Eight data sets are employed to conduct our experiments, including four benchmark data sets (i.e., Chess, Hillva, Vechile, and Newsgroup) from UCI Machine Learning Repository,[1] and image data sets (i.e., Coil,[2] Yaleb,[3] Orl[4]) and one signal data set Isolet.[5] We give a detailed description about these data sets in Table 2.

5.2 Comparison methods

To test the performance of the proposed method, nine state-of-the-art are employed to be tested including the Baseline, two filter-based methods (i.e., LS LLES), two graph-based manifold methods (i.e., Ncut and Rcut), four embedded-based methods (i.e., CDLFS, SCUFS, RUSFS, and RNE). Besides, we introduce the comparison schemes as follows,

- **Baseline** [16] uses K-means way to work on all features, and then, updates samples assignment and centroids computation iteratively until the clustering results are obtained.
- **Ratio cut (Rcut)** [15] constructs a graph based on all samples, and utilizes normalized segmentation criterion to finish data division.
- **Normalized cut (Ncut)** [41] converts clustering task into graph partitioning problem by extracting the global representation of all data.
- **Laplacian Score (LS)** [17] belongs to a filter-based unsuperivsed learning model which keeps local relationships by graph Laplacian and selects the feature subset with larger scores.
- **Coupled Dictionary Learning Feature Selection (CDLFS)** [56] uses dictionary learning to reconstruct the data and learns a coefficient matrix to represent feature importance.
- **Locally Linear Embedding Score (LLES)** [52] measures relations of features and maintains the local structure between features in original data, as well as selects larger scores of features for the final clustering task.

[1] https://archive.ics.uci.edu/ml/.

[2] https://featureselection.asu.edu/datasets/php.

[3] https://www.cad.zju.edu.cn/home/dengcai/Data/data.html.

[4] featureselection.asu.edu/datasets/php.

[5] featureselection.asu.edu/datasets/php.

Table 2 The description of the data sets

Data sets	Samples	Dimensions	Classes	Types
Chess	3196	36	2	Multivariate
Hillva	606	101	2	Sequential
Vechile	846	18	4	Multivariate
Newsgroup	20000	256	12	Text
Coil	1024	1440	20	Image
Isolet	1560	617	26	Signal
Yaleb	1024	2414	38	Image
Orl	1024	400	40	Image

- **Subspace Clustering guided Unsupervised Feature Selection (SCUFS)** [57] employs self-representation and sparse subspace learning to search a robust graph representation and coefficient matrix.
- **Robust Unsupervised Spectral Feature Selection (RUSFS)** [58] considers both local structure and global structure of samples and features separately, and employs $\ell_{2,1}$-norm to remove the unimportant features.
- **Robust Neighborhood Embedding (RNE)** [27] selects the important features subset by minimizing reconstruction error with ℓ_1-norm regularization constrain.

The employed comparison methods contain one Baseline method, two graph partition methods (i.e., Rcut and Ncut), two filter methods (LS and LLES), and four embedded methods (CDLFS, SCUFS, RUSFS, and RNE). Thereinto, graph partition-based methods use graph representation to consider samples' local correlations combining with clustering task which can also guide the samples preservation among the data, but they do not consider the feature importance and select the representative features for the task learning. Filter-based methods employ the specific evaluation method (i.e., Laplacian score or locally linear embedding score) to select the informative features after ranking the scores of all features for the task learning. However, they do not take the importance of samples and features into account at the same time. Embedded-based methods employ different ways to the learning of important features where the coefficient weight is used to calculate the scores of features. But all of them consider either the part of samples importance using a self-representation way or feature importance using sparse learning, the considerations are insufficient.

5.3 Experimental setting

We repeated the K-means clustering process twenty times and obtained the final clustering results by calculating the average of all results, preserving the fair experimental environment and avoiding the problem of initialization for all methods. Additionally, we followed the parameters setting of each comparison method by the suggestion from the original paper. For our scheme, there are two variables (i.e., α and γ) that are setting to $\{10^{-3}, ..., 10^{3}\}$, and p is setting to [0.5, 1.0, 1.5, 2.0], and we reported the best clustering performance by employing a heuristic search strategy. For simplicity, we let c to be equal to the number of classes for all methods.

We compared all methods using three evaluation metrics, i.e., clustering ACCuracy (ACC), Normalized Mutual Information (NMI), and Purity. All metrics were limited between 0 and 1, and the higher value the metric is, the better result the clustering task

Table 3 The clustering results on four Benchmark data set

Methods	Chess			Hillva			Vechile			Newsgroup		
	ACC	NMI	Purity	ACC	NMI	Purity	ACC	NMI	Purity	ACC	NMI	Purity
Baseline	0.515	0.175	0.534	0.508	0.059	0.519	0.462	0.227	0.466	0.271	0.045	0.264
Rcut	0.531	0.397	0.590	0.539	0.101	0.554	0.492	0.371	0.484	0.277	0.037	0.271
Ncut	0.540	0.266	0.564	0.522	0.073	0.538	0.494	0.331	0.492	0.273	0.032	0.266
LS	0.595	0.477	0.582	0.551	0.151	0.548	0.561	0.401	0.555	0.381	0.112	0.415
CDLFS	0.651	0.501	0.644	0.584	0.189	0.572	0.593	0.444	0.580	0.401	0.144	0.433
LLES	0.657	0.519	0.658	0.599	0.192	0.610	0.627	0.420	0.611	0.459	0.202	0.466
SCUFS	0.674	0.490	0.666	0.618	0.174	0.604	0.619	0.488	0.607	0.470	0.198	0.483
RUSFS	**0.688**	0.503	0.675	0.634	0.202	**0.647**	0.644	0.489	**0.641**	0.517	**0.233**	0.497
RNE	0.672	0.511	0.680	0.642	0.222	0.628	0.655	**0.509**	0.633	0.528	0.206	0.513
Proposed	0.683	**0.532**	**0.687**	**0.649**	**0.238**	0.642	**0.662**	0.493	0.637	**0.535**	0.227	**0.518**

has. Moreover, we investigated the proposed method performance in two sides, such as the parameters' sensitivity, convergence performance of the proposed method.

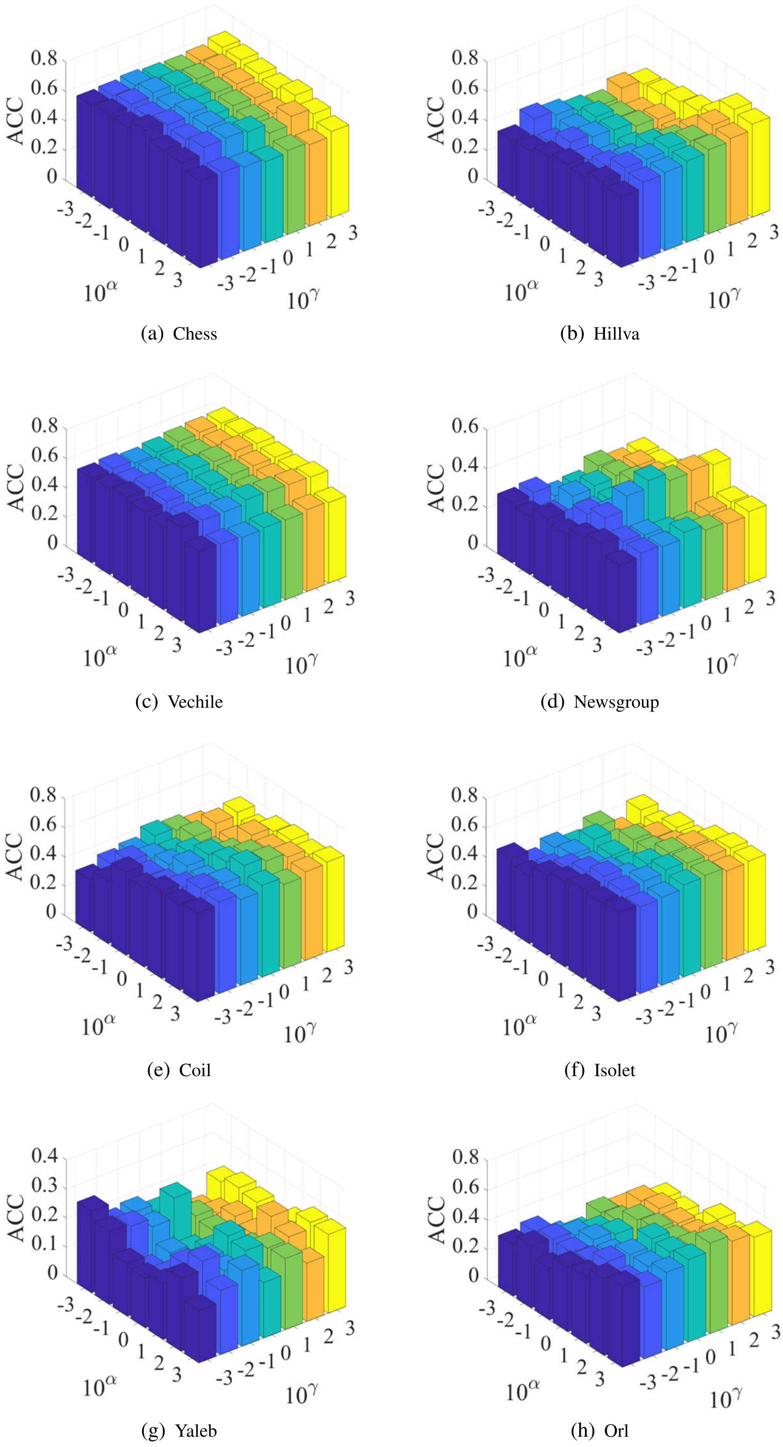
5.4 The results of clustering performance

Tables 3 and 4 indicated clustering results of all approaches we compared and all data sets we experimented with. Obviously, the proposed method obtained the best clustering performance in most of data sets, followed by RUSFS, RNE, SCUFS, LLES, CDLFS, LS, Rcut, Ncut, and Baseline. For instance, the proposed method improved averagely by 2.55%, 0.26%, and 1.63%, respectively, compared to the best comparison method (i.e., RUSFS) with regard to ACC, NMI, and Purity in all data sets. Hence, we obtain the observations as follows.

First, the clustering performance of embedded-based methods are better than filter-based and graph-partition. For example, the proposed method averagely increased by 6.18% and

Table 4 The clustering results on four common data sets

Methods	Coil			Isolet			Yaleb			Orl		
	ACC	NMI	Purity	ACC	NMI	Purity	ACC	NMI	Purity	ACC	NMI	Purity
Baseline	0.654	0.765	0.665	0.611	0.759	0.639	0.098	0.130	0.114	0.527	0.726	0.517
Rcut	0.642	0.748	0.645	0.577	0.680	0.594	0.118	0.190	0.134	0.494	0.699	0.531
Ncut	0.645	0.686	0.629	0.555	0.688	0.583	0.122	0.174	0.151	0.489	0.675	0.557
LS	0.627	0.735	0.632	0.592	0.737	0.565	0.091	0.142	0.123	0.496	0.713	0.519
CDLFS	0.648	0.744	0.641	0.629	0.727	0.615	0.137	0.234	0.145	0.552	0.723	0.564
LLES	0.666	0.767	0.662	0.559	0.688	0.551	0.137	0.195	0.124	0.534	0.734	0.547
SCUFS	0.685	0.775	0.674	0.612	0.748	0.623	0.159	0.242	0.173	0.565	0.734	0.560
RUSFS	**0.697**	**0.801**	0.720	0.645	0.788	**0.679**	0.180	**0.263**	0.211	0.594	**0.753**	0.611
RNE	0.669	0.770	0.663	0.628	0.772	0.634	0.117	0.171	0.136	0.554	0.752	0.599
Proposed	0.685	0.799	**0.721**	**0.657**	**0.791**	0.664	**0.271**	0.229	**0.264**	**0.661**	0.744	**0.678**



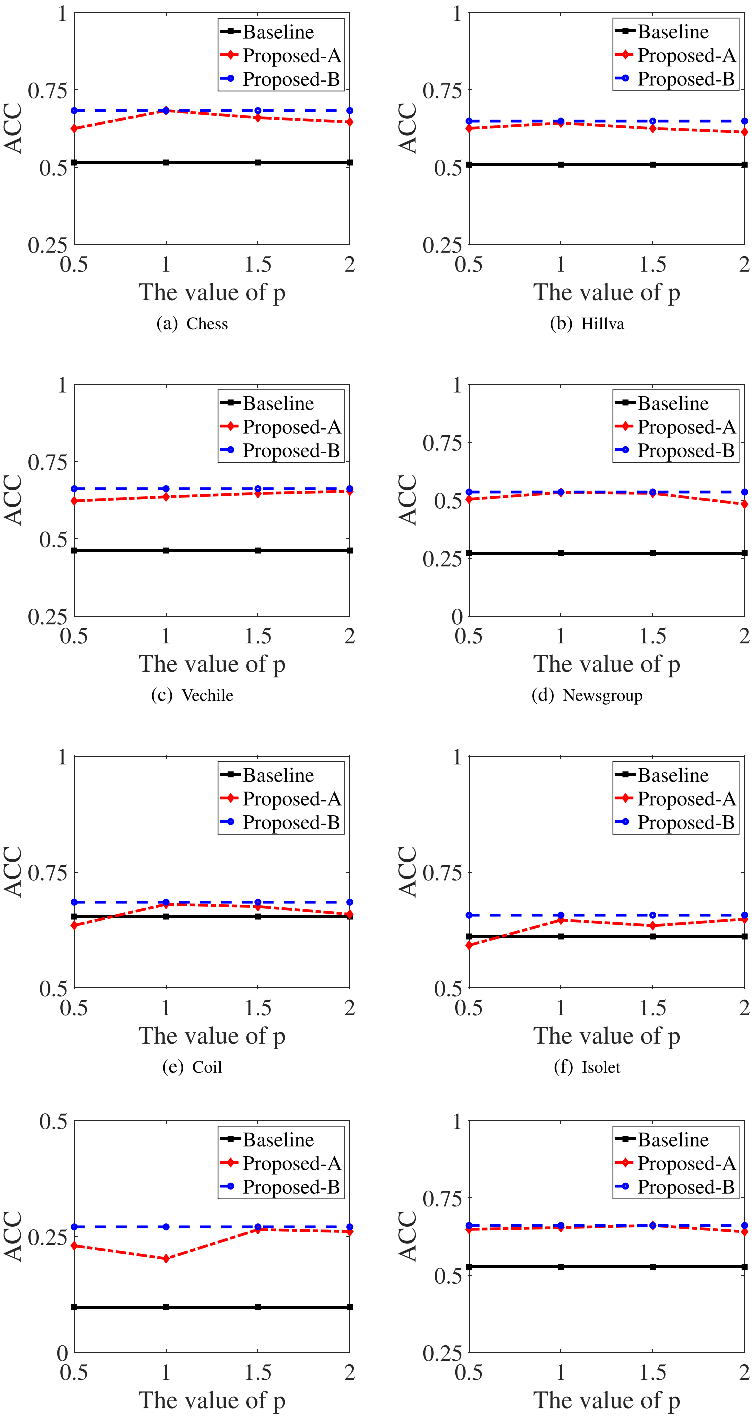
(a) Chess (b) Hillva

(c) Vechile (d) Newsgroup

(e) Coil (f) Isolet

(g) Yaleb (h) Orl

Fig. 1 The sensitivity of α and γ of the proposed method on all data sets



(a) Chess

(b) Hillva

(c) Vechile

(d) Newsgroup

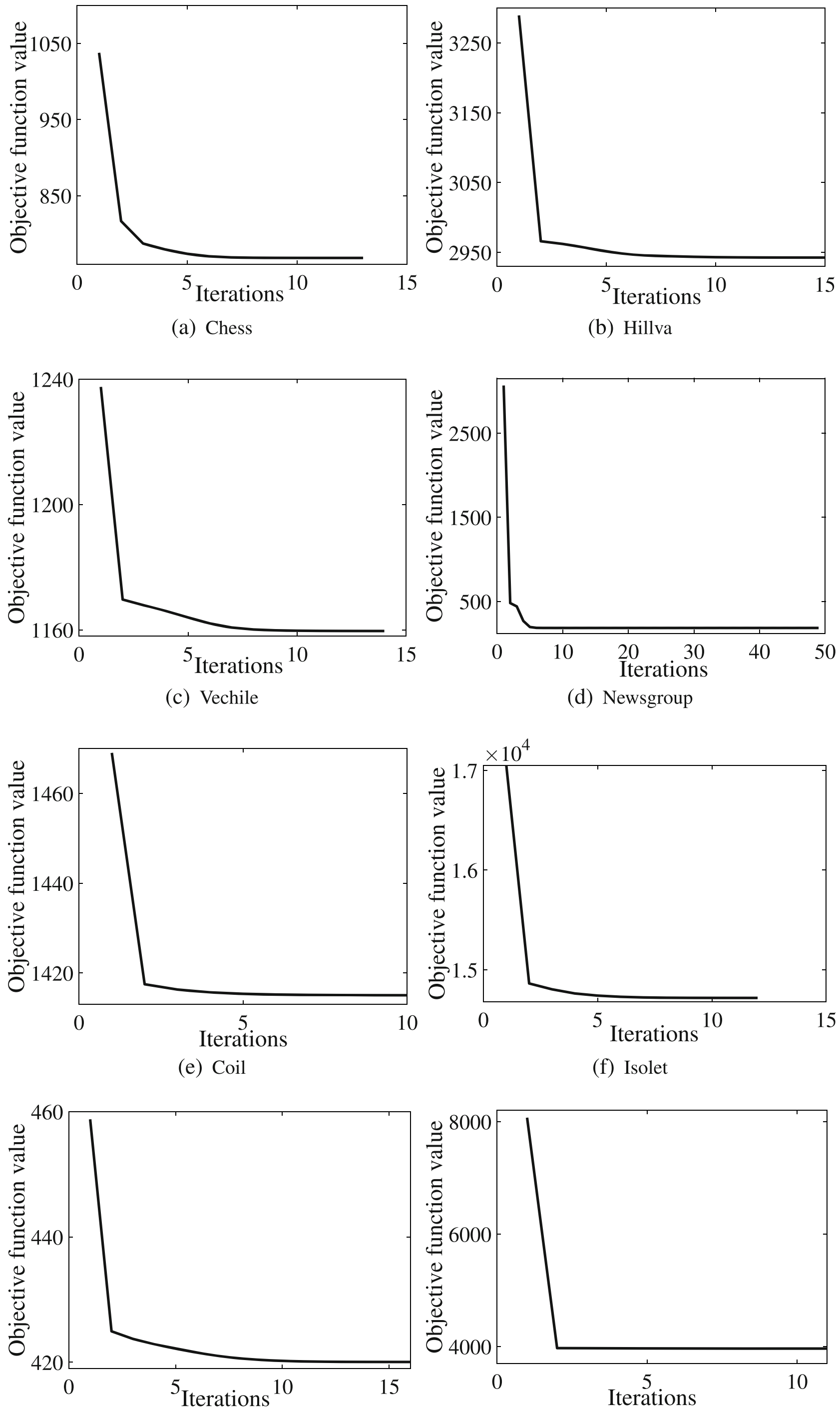
(e) Coil

(f) Isolet

(g) Yaleb

(h) Orl

Fig. 2 The sensitivity of *p* of the proposed method on all data sets



(a) Chess

(b) Hillva

(c) Vechile

(d) Newsgroup

(e) Coil

(f) Isolet

(g) Yaleb

(h) Orl

Fig. 3 The Objective function value of the proposed method at different iterations on all data sets

12.38% compared to the best filter-based method (i.e., LLES) and the best graph-partition method (i.e., Rcut), respectively, in terms of three evaluation metrics. This shows that the embedded method demonstrated the superior performance and the better model construction than the other two approaches.

Second, most of comparison methods are better than Baseline which utilizes K-means on the original data set. For example, the proposed method and the worst embedded-based method (i.e., CDLFS) improved on average by 14.46% and 6.86% in regards to ACC in all data sets, respectively. This demonstrates the effect of feature selection is necessary to explore the representative features and to be helpful for the clustering task.

5.5 Parameters' sensitivity

We adjusted the parameter α and γ in $\{10^{-3}, ..., 10^{3}\}$ and p in [0.5, 1.0, 1.5, 2.0], and the results can be found in Figs. 1 and 2.

In Fig. 1, the combination of parameters in the proposed method are sensitive to the parameters' setting in the experiments. In other words, various parameters combination may output various clustering results. Therefore, it is essential to tune the parameters for the proposed method. For example, our method obtains the best clustering results on the data set Vechile and Isolet while setting $\alpha = 10^2$, $\gamma = 10^{-3}$ and $\alpha = 10$, $\gamma = 10^{-2}$, respectively.

In Fig. 2, diverse values of the parameter p relate to the clustering performance have been demonstrated, which is denoted by "Proposed-A", and "Porpoosed-B" denotes the best clustering performance of our proposed method. For the most of data sets, as the value of p increases, the clustering result first rises and then falls. For example, our method obtains the best clustering results (i.e., 68.3%) when $p = 1$ and the worst clustering results (i.e., 62.5%) when $p = 0.5$ on the data set Chess. Similarly, our method obtains the best clustering results when the value of p is between 1 and 1.5 on the data set Coil.

5.6 Convergence analysis

In Fig. 3, the convergence performance of each data set can be found. Thereinto, the objective function value is monotonically reduced in (9) until Algorithm 1 obtains convergence. Moreover, by observing all figures, we found fifteen iterations may be the best times for most of the data sets which obtaining convergence results. Hence, the proposed Algorithm 1 is efficient for the most of data sets.

6 Conclusion

In this article, we have proposed a new unsupervised learning model to feature selection, via involving two components, including both global and local structure preservation to samples by self-representation way and graph representation, and local information and global information consideration to features by mutual information and sparse learning. Moreover, the better sample correlations and the reliable adjacent information are considered by dynamic graph learning and low-rank constraint, respectively, for clustering task. Experiments on eight data sets with nine comparison approaches have verified the validity of the proposed model.

Acknowledgements This work is partially supported by the Massey University Research Fund Early Career Round and the Massey University College of Sciences' REaDI funds.

Funding Open Access funding enabled and organized by CAUL and its Member Institutions.

Declarations

Conflict of Interests The authors declare that neither associations nor perceived conflicts of interest nor competing financial interests exist in this paper.

References

1. Abu Khurma R, Aljarah I, Sharieh A, Elaziz MA, Damaševičius R, Krilavičius T (2022) A review of the modification strategies of the nature inspired algorithms for feature selection problem. Mathematics 10(3):464
2. Agnihotri D, Verma K, Tripathi P (2017) Variable global feature selection scheme for automatic classification of text documents. Expert Syst Appl 81:268–281
3. Alsahaf A, Petkov N, Shenoy V, Azzopardi G (2022) A framework for feature selection through boosting. Expert Syst Appl 187:115895
4. Askari S (2021) Fuzzy c-means clustering algorithm for data with unequal cluster sizes and contaminated with noise and outliers: review and development. Expert Syst Appl 165:113856
5. Bommert A, Welchowski T, Schmid M, Rahnenführer J (2022) Benchmark of filter methods for feature selection in high-dimensional gene expression survival data. Brief Bioinform 23(1):bbab354
6. Boyd S, Boyd SP, Vandenberghe L (2004) Convex optimization. Cambridge University Press, Cambridge
7. Cai D, Zhang C, He X (2010) Unsupervised feature selection for multi-cluster data. In: Proceedings of the 16th ACM SIGKDD international conference on knowledge discovery and data mining, pp 333–342
8. Cekik R, Uysal AK (2020) A novel filter feature selection method using rough set for short text data. Expert Syst Appl 160:113691
9. Chen G, Chen J (2015) A novel wrapper method for feature selection and its applications. Neurocomputing 159:219–226
10. Daubechies I, DeVore R, Fornasier M, Sinan Güntürk C (2010) Iteratively reweighted least squares minimization for sparse recovery. Communications on Pure and Applied Mathematics: A Journal Issued by the Courant Institute of Mathematical Sciences 63(1):1–38
11. Elhamifar E, Vidal R (2013) Sparse subspace clustering: algorithm, theory, and applications. IEEE Trans Pattern Anal Mach Intell 35(11):2765–2781
12. Fan K (1949) On a theorem of weyl concerning eigenvalues of linear transformations i. Proc Natl Acad Sci U S A 35(11):652
13. Feng S, Duarte MF (2018) Graph autoencoder-based unsupervised feature selection with broad and local data structure preservation. Neurocomputing 312:310–323
14. Feofanov V, Devijver E, Amini M-R (2022) Wrapper feature selection with partially labeled data. Appl Intell:1–14
15. Hagen L, Kahng AB (1992) New spectral methods for ratio cut partitioning and clustering. IEEE Trans Comput Aided Des Integ Circ Syst 11(9):1074–1085
16. Hartigan JA, Wong MA (1979) Algorithm as 136: a k-means clustering algorithm. J R Stat Soc Ser C (Appl Stat) 28(1):100–108
17. He X, Cai D, Niyogi P (2005) Laplacian score for feature selection. Adv Neural Inf Process Syst:18
18. Hou C, Nie F, Li X, Yi D, Wu Y (2013) Joint embedding learning and sparse regression: a framework for unsupervised feature selection. IEEE Trans Cybern 44(6):793–804
19. Hu H, Lin Z, Feng J, Zhou J (2014) Smooth representation clustering. In: Computer vision and pattern recognition, pp 3834–3841

20. Hu R, Zhu X, Cheng D, He W, Yan Y, Song J, Zhang S (2017) Graph self-representation method for unsupervised feature selection. Neurocomputing 220:130–137
21. Hu R, Zhu X, Zhu Y, Gan J (2020) Robust svm with adaptive graph learning. World Wide Web 23(3):1945–1968
22. Hu R, Peng Z, Zhu X, Gan J, Zhu Y, Ma J, Wu G (2021) Multi-band brain network analysis for functional neuroimaging biomarker identification. IEEE Trans Med Imaging 40(12):3843–3855
23. Hu R, Gan J, Zhu X, Liu T, Shi X (2022) Multi-task multi-modality svm for early covid-19 diagnosis using chest ct data. Inf Process Manag 59(1):102782
24. Lim H, Kim D-W (2021) Pairwise dependence-based unsupervised feature selection. Pattern Recogn 111:107663
25. Liu G, Lin Z, Yan S, Sun J, Yu Y, Ma Y (2012) Robust recovery of subspace structures by low-rank representation. IEEE Trans Pattern Anal Mach Intell 35(1):171–184
26. Liu X, Wang L, Zhang J, Yin J, Liu H (2013) Global and local structure preservation for feature selection. IEEE Trans Neural Netw Learn Syst 25(6):1083–1095
27. Liu Y, Ye D, Li W, Wang H, Gao Y (2020) Robust neighborhood embedding for unsupervised feature selection. Knowl-Based Syst 193:105462
28. Luo M, Nie F, Chang X, Yang Y, Hauptmann AG, Zheng Q (2017) Adaptive unsupervised feature selection with structure regularization. IEEE Trans Neural Netw Learn Syst 29(4):944–956
29. Miao J, Yang T, Sun L, Fei X, Niu L, Shi Y (2022) Graph regularized locally linear embedding for unsupervised feature selection. Pattern Recogn 122:108299
30. Nie F, Wang X, Huang H (2014) Clustering and projected clustering with adaptive neighbors. In: SIGKDD, pp 977–986
31. Nie F, Zhu W, Li X (2016) Unsupervised feature selection with structured graph optimization. In: AAAI, pp 1302–1308
32. Nouri-Moghaddam B, Ghazanfari M, Fathian M (2021) A novel multi-objective forest optimization algorithm for wrapper feature selection. Expert Syst Appl 175:114737
33. Onyema EM, Elhaj MAE, Bashir SG, Abdullahi I, Hauwa AA, Hayatu AA, Edeh MO, Abdullahi I (2020) Evaluation of the performance of k-nearest neighbor algorithm in determining student learning styles. Int J Innov Sci Eng Technol 7(1):91–102
34. Onyema EM, Shukla PK, Dalal S, Mathur MN, Zakariah M, Tiwari B (2021) Enhancement of patient facial recognition through deep learning algorithm: convnet. J Healthc Eng 2021
35. Patel VM, Van Nguyen H, Vidal R (2013) Latent space sparse subspace clustering. In: ICCV, pp 225–232
36. 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
37. Qiao L, Chen S, Tan X (2010) Sparsity preserving projections with applications to face recognition. Pattern Recogn 43(1):331–341
38. Robnik-Šikonja M, Kononenko I (2003) Theoretical and empirical analysis of relieff and rrelieff. Mach Learn 53(1):23–69
39. Shang R, Wang W, Stolkin R, Jiao L (2017) Non-negative spectral learning and sparse regression-based dual-graph regularized feature selection. IEEE Trans Cybern 48(2):793–806
40. Sheikhpour R, Sarram MA, Gharaghani S, Chahooki MAZ (2017) A survey on semi-supervised feature selection methods. Pattern Recogn 64:141–158
41. Shi J, Malik J (2000) Normalized cuts and image segmentation. IEEE Trans Pattern Anal Mach Intell 22(8):888–905
42. Solorio-Fernández S, Ariel Carrasco-Ochoa J, Martínez-Trinidad JF (2020) A review of unsupervised feature selection methods. Artif Intell Rev 53(2):907–948
43. Solorio-Fernández S, Martínez-Trinidad JF, Ariel Carrasco-Ochoa J (2020) A supervised filter feature selection method for mixed data based on spectral feature selection and information-theory redundancy analysis. Pattern Recogn Lett 138:321–328
44. Song L, Smola A, Gretton A, Borgwardt KM, Bedo J (2007) Supervised feature selection via dependence estimation. In: Proceedings of the 24th international conference on machine learning, pp 823–830
45. Song QJ, Jiang HY, Liu J (2017) Feature selection based on fda and f-score for multi-class classification. Expert Syst Appl 81:22–27
46. Wahid A, Khan DM, Hussain I, Khan SA, Khan Z (2022) Unsupervised feature selection with robust data reconstruction (ufs-rdr) and outlier detection. Expert Syst Appl:117008
47. Wang S, Zhu W (2016) Sparse graph embedding unsupervised feature selection. IEEE Trans Syst Man Cybern: Syst 48(3):329–341
48. Wang C, Gong L, Jia F, Zhou X (2020) An fpga based accelerator for clustering algorithms with custom instructions. IEEE Trans Comput 70(5):725–732

49. Wen Z, Yin W (2013) A feasible method for optimization with orthogonality constraints. Math Program 142(1):397–434
50. Wu J-S, Song M-X, Min W, Lai J-H, Zheng W-S (2021) Joint adaptive manifold and embedding learning for unsupervised feature selection. Pattern Recogn 112:107742
51. Xu W, Jang-Jaccard J, Liu T, Sabrina F (2022) Training a bidirectional gan-based one-class classifier for network intrusion detection. arXiv:2202.01332
52. Yao C, Liu Y-F, Bo J, Han J, Han J (2017) Lle score: a new filter-based unsupervised feature selection method based on nonlinear manifold embedding and its application to image recognition. IEEE Trans Image Process 26(11):5257–5269
53. Yuan H, Li J, Lai LL, Tang YY (2019) Joint sparse matrix regression and nonnegative spectral analysis for two-dimensional unsupervised feature selection. Pattern Recogn 89:119–133
54. Zhang Y, Zhang Z, Qin J, Li Z, Li B, Li F (2018) Semi-supervised local multi-manifold isomap by linear embedding for feature extraction. Pattern Recogn 76:662–678
55. Zhao Z, Liu H (2007) Spectral feature selection for supervised and unsupervised learning. In: ICML, pp 1151–1157
56. Zhu P, Hu Q, Zhang C, Zuo W (2016) Coupled dictionary learning for unsupervised feature selection. In: AAAI
57. Zhu P, Zhu W, Hu Q, Zhang C, Zuo W (2017) Subspace clustering guided unsupervised feature selection. Pattern Recogn 66:364–374
58. Zhu X, Zhang S, Hu R, Zhu Y et al (2018) Local and global structure preservation for robust unsupervised spectral feature selection. IEEE Trans Knowl Data Eng 30(3):517–529