Pattern Recognition 114 (2021) 107873

Contents lists available at ScienceDirect

Pattern Recognition

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

Dual space latent representation learning for unsupervised feature selection

Ronghua Shang*, Lujuan Wang, Fanhua Shang, Licheng Jiao, Yangyang Li

Key Laboratory of Intelligent Perception and Image Understanding of Ministry of Education, Xidian University, Xi'an, Shaanxi Province 710071, China

ARTICLE INFO

Article history:
Received 27 March 2020
Revised 3 December 2020
Accepted 31 January 2021
Available online 2 February 2021

Keywords:
Latent representation learning
Unsupervised feature selection
Dual space
Sparse regression

ABSTRACT

In real-world applications, data instances are not only related to high-dimensional features, but also interconnected with each other. However, the interconnection information has not been fully exploited for feature selection. To address this issue, we propose a novel feature selection algorithm, called dual space latent representation learning for unsupervised feature selection (DSLRL), which exploits the internal association information of data space and feature space to guide feature selection. Firstly, based on latent representation learning in data space, DSLRL produces dual space latent representation learning, which characterizes the inherent structure of data space and feature space, respectively. Secondly, in order to overcome the problem of the lack of label information, DSLRL optimizes the low-dimensional latent representation matrix of data space as a pseudo-label matrix to provide clustering indicators. Moreover, the latent representation matrix of feature space is unified with the transformation matrix to benefit the matching of the data matrix and the clustering indicator matrix. In addition, DSLRL uses non-negative and orthogonal conditions to constrain the sparse transform matrix, making it more accurate for evaluating features. Finally, an alternating method is employed to optimize the objective function. Compared with seven state-of-the-art algorithms, experimental results on twelve datasets show the effectiveness of DSLRL.

1. Introduction

With the development of science and technology, the era of big data has arrived. A large amount of data is generated every day, and the dimension of data is increasing [1]. Processing these high-dimensional data directly not only greatly increases the computation time and storage space, but also results in poor performance due to the existence of noise, irrelevant features, and redundant features [2]. Therefore, it is necessary to overcome the "dimensional disaster" caused by large-scale high-dimensional data. Experiments show that effective dimensionality reduction methods can not only reduce the cost of data processing, but also effectively improve the performance of clustering algorithm [3,4]. Feature selection is one of the common dimensionality reduction methods, which is designed to select a representative subset to represent the original data [5].

According to the availability of sample labels, previous feature selection methods are usually divided into three categories: supervised methods [2,6], semi-supervised methods [7,8], and unsupervised methods [9,10]. In the supervised feature selection methods, all training sample labels are known in advance. These methods evaluate the importance of each feature based on the correlation between labels and features, and select discriminative features. When there are only a few data labels, the semi-supervised feature selection methods can effectively improve the accuracy of feature selection [7], which mine the relationship between the data and build a similarity matrix. In supervised and semi-supervised feature selection methods, all or part of the data labels need to be known. However, in most practical applications, it is laborious to obtain data labels. In this case, the advantages of unsupervised feature selection methods are obvious compared to the first two methods. These methods can determine the importance of features through the underlying attributes of the original data without the label information [9]. Therefore, the unsupervised feature selection method is more suitable for dimension reduction of high-dimensional data. In this paper, we focus on unsupervised feature selection.

According to different search strategy, unsupervised feature selection methods can generally be classified into three types, including filter methods [11-13], wrapper methods [14,15], and embedded methods [16-18]. Filter methods evaluate the importance of features based on the statistical characteristics of data, and then select top-ranked features [11]. Commonly used metrics include

* Corresponding author.
E-mail address: rhshang@mail.xidian.edu.cn (R. Shang).

https://doi.org/10.1016/j.patcog.2021.107873
0031-3203/

variance, Laplace score, similarity of features, and more. Due to the low cost and high efficiency, filter methods are widely applied in engineering field. But the subset selected by these methods often contains noise. Wrapper methods usually select features based on learning tasks such as clustering [14]. In general, wrapper-based methods outperform filter methods. However, because of the high computational cost, wrapper methods are not suitable for processing dimensionality reduction of large-scale data. Embedded methods combine feature selection and model optimization. Thus, they can quickly select proper subset during the learning process. Compared with filter methods and wrapper methods, embedded methods are not only efficient but also have better performance. And how to construct a suitable model is the most critical issue in the embedded methods [17].

In the past few decades, many different types of unsupervised feature selection algorithms have been proposed, and the processes of them are roughly similar. The importance of each feature is evaluated according to a certain evaluation criterion, and then a small number of representative and discriminative features are selected as a subset to complete the clustering or classification task [19]. For example, He et al. [12] proposed Laplacian score (LapScor), which calculates the weight of feature based on data manifold information. Higher scores indicate that features are more important. Under the same principle, spectral feature selection (SPEC) [11] adopts another criterion to calculate feature weight. Both of them are the classical feature selection algorithms which construct an affinity graph to model the local geometric structure, and SPEC is an extension of Laplacian score. Cai et al. [20] proposed multi-cluster feature selection (MCFS), which first obtains data geometric structure information via spectral analysis techniques, and then uses a sparse transformation matrix to embed the data into low-dimensional space. Minimum redundancy spectral feature selection (MRSF) is similar to MCFS, the main difference is that the former utilizes the l_1-*norm* to constrain sparse transformation matrix, while the latter uses the $l_{2,1}$-*norm* constraint [21]. They both adopt a two-step strategy to perform embedding learning and regression separately. Unlike MCFS and MRFS, Hou et al. [22] proposed joint embedding learning and sparse regression (JELSR), which adopts a single step strategy to optimize embedding matrix and sparse transformation matrix simultaneously, thus better performing feature selection. Subsequently, Nie et al. proposed an unsupervised feature selection approach, called structured optimal graph feature selection (SOGFS), which performs feature selection and local structure learning simultaneously. Thus, it can adaptively determine the similarity matrix [23]. Li et al. proposed generalized uncorrelated regression with adaptive graph for unsupervised feature selection (URAFS) [24]. Meanwhile, Shang et al. [25] proposed unsupervised feature selection based on self-representation sparse regression and local similarity preserving (UFSRL), which is sparse reconstruction of the original data itself. UFSRL has imposed the $l_{2,1/2}$-matrix norm on the coefficient matrix, making the proposed model sparse and robust to noise. In recent years, some unsupervised feature selection algorithms based on representation learning have been proposed. He et al. [26] proposed feature self-representation based hypergraph unsupervised feature selection via low-rank representation (SHLFS), which could efficiently select a subset of informative features from unlabeled data. SHLFS integrates the low-rank constraint, hypergraph theory, and the self-representation property of features in a unified framework. In particular, SHLFS represents each feature by other features to conduct unsupervised feature selection via the feature-level self-representation property. Tang et al. [27] proposed robust unsupervised feature selection via dual self-representation and manifold regularization (DSRMR). DSRMR constructs both feature self-representation and sample self-representation terms, which are used to respectively learn the feature representation coefficient matrix and sample similarity graph to guide feature selection. Fan et al. also [28] considered the distribution information of feature space, and they proposed latent space embedding for unsupervised feature selection via joint dictionary learning (LSEUFS), which capture the common distribution of feature space and pseudo label space. LSEUFS integrates joint dictionary learning, spectral analysis and feature selection into a unified model.

The traditional unsupervised feature selection method usually assumes that the data instances are ideally independently distributed. However, in the real world, due to the influence of external conditions, data instances are not only related to high-dimensional features, but also inherently associated with each other. For addressing this issue, Tang et al. [29] proposed unsupervised feature selection by latent representation learning and manifold regularization (LRLMR), which exploits the link information between data instances to select relevant features. Meanwhile, the local structure of original data is preserved by a graph regularization term in a low-dimensional feature space. However, LRLMR only utilizes the internal information of data space, and ignores the internal interconnection information of feature space. The relationship between features becomes more complicated as the number increases, such as similarity and redundancy. Redundant feature means that the information it contains can be derived from other features. Therefore, the internal information in the feature space is worth exploring.

The latent representation model from the link information could capture the clustering structure through symmetric non-negative matrix factorization [30]. In recent years, several related algorithms based on the information of both data space and feature space have been proposed and show good performance. Luo et al. [31] proposed dual-regularized multi-view non-negative matrix factorization (DMvNMF), which is developed for multi-view data clustering. DMvNMF is able to preserve the geometric information of multi-view data in both the data space and the feature space. Based on concept factorization(CF),Ye and Jin [32] proposed dual-graph regularized concept factorization clustering (GCF), which simultaneously construct data graph and feature graph to model the geometric structures of both spaces. Compared with traditional one-sided clustering algorithms, GCF shows better performance. Then Ye and Jin [33] also proposed adaptive dual-graph regularized CF with Feature selection ($ADGCF_{FS}$), which unified feature selections and dual-graph regularized CF into a joint objective function. The above algorithms that exploit the information of dual space have better performance than algorithms that only uses the information of data space.

Based on the above considerations, dual space latent representation learning for unsupervised feature selection (DSLRL) is proposed in this work, which utilizes the intrinsic association information of data space and feature space to improve the performance of feature selection. The clustering structure of data clustering is obtained by latent representation learning in data space, and the clustering structure of feature clustering is obtained through latent representation learning in feature space. Specifically, the proposed algorithm constructs affinity matrices in both data space and feature space, respectively, which are used to characterize the internal relationships of the samples and the internal relationships of the features. Through the affinity matrices, latent representation learning is performed in dual space to separately obtain the low-dimensional representations of data and feature. The former reveals the relevant information between instances and clusters, while the latter records the relationship between features and clusters. In both, the larger value, the more relevant. In the previous work [23,24], a good projection transformation matrix should match the data matrix to the cluster indicator matrix as accurate as possible [34]. In the latent representation matrix of features, the correlation information between features and clusters is beneficial

to perform the projection process more accurately. Therefore, the low-dimensional latent representation of feature is unified with the projection transformation matrix, and the low-dimensional latent representation of data is equal to the pseudo label matrix. In this way, the internal information of dual space is fully used to guide feature selection. Some previous methods measured the feature importance in the original data space, and the performance of these methods is usually influenced by the noisy features [29]. Rather than those methods, DSLRL performs feature selection in the low-dimensional latent representation space to reduce the impact of noisy information. Our main contributions are highlighted as follows.

1) Latent representation learning based on dual space is proposed, which characterizes the inherent structure of data space and feature space, respectively, to reduce the negative influence of noise and redundant information.
2) The latent representation matrix of data space is regarded as a pseudo label matrix to provide discriminative information. Moreover, the latent representation matrix of feature space is unified with the transformation matrix to benefit the matching of the data matrix and the clustering indicator matrix, thereby making full use of the internal information of dual space in feature selection.
3) In order to avoid the emergence of trivial solutions, non-negative constraints and orthogonal constraints are imposed on the sparse transformation matrix, so that the importance of each feature can be better reflected.

The rest of the paper is organized as follows. Some related feature selection methods are introduced in Section 2. Section 3 presents the proposed algorithm, optimization method, convergence analysis and complexity analysis. In Section 4, the experimental results and analysis of DSLRL and seven comparison algorithms are shown. The conclusions and future work are summarized in Section 5.

2. Related work

This section introduces the concept of latent representation learning and its application in unsupervised feature selection. In addition, several related unsupervised feature selection methods are briefly explained.

Before introducing the following content, a notation table is listed to more clearly explain the notations which are used in this paper. Table 1 is a notation comparison table.

2.1. Latent representation

In recent years, latent representation has been found to benefit for many data mining and machine learning tasks, especially for network data [34]. As a result, it has attracted increasingly attention [36,37]. In the network, there are connections between instances due to various factors, and these hidden factors are often referred as latent representations [38]. Latent representations of different instances interact with each other and form link information. In general, instances with similar latent representations are more likely to be interconnected than instances with dissimilar latent representations [39]. Therefore, the adjacency matrix $\mathbf{Z} \in \mathbb{R}^{n\times n}$ is used to describe the association between data instances, and the latent representations is obtained from it. Usually, the latent representations from link information can be generated through a symmetric non-negative matrix factorization model [30,40], which decomposes $\mathbf{Z}$into the product form of the non-negative matrix $\mathbf{U}$ and its transpose $\mathbf{U}^T$ as follows:

$$\begin{aligned} &\arg\min ||\mathbf{Z}-\mathbf{U}\mathbf{U}^T||_F^2 \\ &s.t. \mathbf{U} \geq 0 \end{aligned} \tag{1}$$

Table 1
The notation comparison table

Notation	Notation description
$\mathbf{X}$	data matrix
$\mathbf{x}_i$	the ith data sample
$\mathbf{x}^i$	the ith data feature
$\mathbf{Z}$	adjacency matrix
$\mathbf{U}$	latent representation matrix
n	the number of samples
d	the number of features
f	the number of latent factors
c	the number of categories
m	the dimension of low-dimensional space
l	the number of selected features
$Niter$	maximum iteration number
$\mathbf{F}$	low-dimensional embedding matrix
$\mathbf{W}$	projection transformation matrix
$\mathbf{S}$	similarity matrix
$\mathbf{L}$	graph Laplacian matrix
$\mathbf{S}_t$	total scatter matrix
$\mathbf{V}$	latent representation of data space
$\mathbf{A}$	affinity matrix of data space
$\mathbf{B}$	affinity matrix of feature space

where $\mathbf{U} \in \mathbb{R}^{n\times f}$represents the latent representation matrix of n instances, and f denotes the number of latent factors. In [30], symmetric non-negative matrix factorization model is used to capture the clustering structure for data clustering. In other words, latent representation learning clusters n instances into f classes according to the connection information between the instances. Tang et al. borrowed this idea and learned the latent representation from the affinity matrix of data space for feature selection [29]. However, they only considered the internal information of data space, but ignored the application of latent representation learning in feature space.

2.2. Unsupervised feature selection

2.2.1. MCFS

MCFS proposed by Cai et al. [20] mainly consists of two steps. First, the low-dimensional embedding matrix $\mathbf{F} \in \mathbb{R}^{n\times m}$ is obtained from data $\mathbf{X} \in \mathbb{R}^{n\times d}$through manifold analysis, where n denotes the number of samples, d indicates the number of feature, m represents the dimension of the low-dimensional embedding space, and $m<d$. Then the regression coefficient matrix$\mathbf{W} \in \mathbb{R}^{d\times m}$is constrained by l_1-*norm* to more accurately reflect the importance of each feature. Its objective function is formulated as follows:

$$\begin{aligned} &\arg\min_{\mathbf{F}^T\mathbf{F}=\mathbf{I}_m} Tr(\mathbf{F}^T\mathbf{L}\mathbf{F}) \\ &\min_{\mathbf{W}} ||\mathbf{X}\mathbf{W}-\mathbf{F}||_F^2+\alpha||\mathbf{W}||_1 \end{aligned} \tag{2}$$

where $Tr(\cdot)$ denotes the trace of a matrix, $||\mathbf{W}||_1 = \sum_{i=1}^{d}\sum_{j=1}^{m}|\mathbf{W}_{ij}|$ represents l_1-norm for sparse constraints.

MCFS performs embedding learning and sparse regression separately, and the two interact with each other.

2.2.2. JELSR

Different from MCFS, JELSR [22] adopts a single-step strategy, which performs low-dimensional embedding learning and sparse regression simultaneously. And the matrix $\mathbf{W}$ is constrained by $l_{2,1}$-*norm* with better robustness. Its objective function is formulated as follow:

$$\arg\min_{\mathbf{W},\mathbf{F}^T\mathbf{F}=\mathbf{I}_m} Tr(\mathbf{F}^T\mathbf{L}\mathbf{F})+||\mathbf{X}\mathbf{W}-\mathbf{F}||_F^2+\alpha||\mathbf{W}||_{2,1} \tag{3}$$

JELSR integrates the merits of embedding learning and sparse regression to perform feature selection, and further improves its effect.

2.2.3. SOGFS

Different from conventional embedded unsupervised methods, which always need to construct the similarity matrix, SOGFS performs feature selection and local structure learning simultaneously, thus the similarity matrix can be determined adaptively [23]. Given the ith sample $\mathbf{x}_i$ of the data matrix $\mathbf{X} \in \mathbb{R}^{n\times d}$, SOGFS defines that sample $\mathbf{x}_i$ can be connected by others with probability s_{ij}. And the probability of two samples becoming neighbor can be considered as the similarity between them, so s_{ij} is an element of similarity matrix $\mathbf{S}$. The similarity matrix is optimized to obtain the ideal state of neighbor assignment, which is beneficial for feature selection. Its objective function is formulated as follows:

$$\begin{aligned}&\arg\min_{\mathbf{W},\mathbf{F},\mathbf{S}} \sum_{i,j}(||\mathbf{W}^T\mathbf{x}_i - \mathbf{W}^T\mathbf{x}_j||_2^2 s_{ij} + \alpha s_{ij}^2) + \gamma||\mathbf{W}||_{2,1} + 2\lambda Tr(\mathbf{F}^T\mathbf{L}\mathbf{F})\\ &s.t. \forall i, s_i^T\mathbf{1} = 1, 0 \le s_{ij} \le 1, \mathbf{F} \in \mathbb{R}^{n\times m}, \mathbf{F}^T\mathbf{F} = \mathbf{I}, \mathbf{W}^T\mathbf{W} = \mathbf{I}\end{aligned} \tag{4}$$

SOGFS constrains the similarity matrix $\mathbf{S}$ to make it contain more accurate information of data structure, so this method can select more valuable features.

2.2.4. URAFS

Taking into account the redundancy of features, Li et al. improved the sparse regression model for feature selection, and proposed a generalized uncorrelated regression model (GURM) for seeking uncorrelated yet discriminative features [24]. In addition, the graph regularization term based on the principle of maximum entropy is also incorporated into the GURM model, there by URAFS is proposed. Its objective function is expressed as:

$$\begin{aligned}&\arg\min_{\mathbf{W},\mathbf{F},\mathbf{S}} ||\mathbf{G}(\mathbf{X}^T\mathbf{W} - \mathbf{F})||_F^2 + \lambda||\mathbf{W}||_{2,1}\\ &+2\alpha\left(Tr(\mathbf{F}^T\mathbf{L}\mathbf{F}) + \beta\sum_{i=1}^{n}\sum_{j=1}^{n}(s_{ij}\log s_{ij})\right)\\ &s.t. \mathbf{W}^T(\mathbf{S}_t + \lambda\mathbf{D})\mathbf{W} = \mathbf{I}, \mathbf{F}^T\mathbf{F} = \mathbf{I}, \sum_{j=1}^{n} s_{ij} = 1, s_{ij} \ge 0\end{aligned} \tag{5}$$

where $\mathbf{G} = \mathbf{I} - (1/n)\mathbf{1}\mathbf{1}^T$ is referred as the centering matrix. $\mathbf{F}$ denotes the indicator matrix, and s_{ij} is an element of similarity matrix $\mathbf{S}$. $\mathbf{W}^T(\mathbf{S}_t + \lambda\mathbf{D})\mathbf{W} = \mathbf{I}$ is a generalized uncorrelated constraint, where $\mathbf{D}$ is defined as a $d \times d$ diagonal matrix and d_{ii} is derived from$||\mathbf{W}||_{2,1}$, $\mathbf{S}_t$=$\mathbf{X}\mathbf{G}\mathbf{X}^T$ is the total scatter matrix.

2.2.5. LRLMR

Conventional unsupervised feature selection methods are under the assumption that the data is independently distributed, and ignore the connections in data instances. The connections exist in real world and can be used to explore the internal structure of the data. LRLMR [29] embeds the latent representation learning into feature selection to exploit interconnection information in data space. Meanwhile, the local geometric structure of original data is preserved through manifold learning. The objective function of LRLMR is formulated as follows:

$$\begin{aligned}&\arg\min_{\mathbf{W},\mathbf{V}} ||\mathbf{X}\mathbf{W} - \mathbf{V}||_F^2 + \alpha||\mathbf{W}||_{2,1} + \beta||\mathbf{A} - \mathbf{V}\mathbf{V}^T||_F^2\\ &+\gamma Tr(\mathbf{W}^T\mathbf{X}^T\mathbf{L}\mathbf{X}\mathbf{W}) \quad s.t. \mathbf{V} \ge 0\end{aligned} \tag{6}$$

Data labels have contributed to the selection of discriminative and representative features. But in reality, due to the inconvenience of obtaining labels, unsupervised feature selection is considered as a difficult problem. LRLMR models the inherent link information of data space as a pseudo-label matrix to provide clustering indicators, thereby improving the effect of feature selection, but it ignores the intrinsic information from the feature space.

3. The proposed method

In this section, we will give a detailed introduction of DSLRL. In addition, the optimization method and convergence analysis of DSLRL are also described. In our work, the data set is represented by a matrix $\mathbf{X} \in \mathbb{R}^{n\times d}$, the remaining notations that will be used are listed in Table 1.

3.1. Dual space latent representation learning

Traditional unsupervised feature selection methods always suppose that the distribution state of the data is independent and uniform under ideal conditions. However, this distribution is under ideal conditions, not in the real world. Similar to the homophile effect, under the influence of various external conditions, data instances generated from homologous or heterogeneous sources often rely on each other [35]. Hence, it is very necessary to characterize the intrinsic data structure via link information. LRLMR implements this idea by latent representation learning, and applies it to unsupervised feature selection. Inspired by LRLMR, this algorithm proposes latent representation learning based on dual space, which mines the interconnection information of both data space and feature space, simultaneously, in order to characterize their intrinsic structure, thereby better performing feature selection.

Firstly, we choose Gaussian function [41] as weight measures, and then construct an affinity matrix $\mathbf{A}$ in data space to represent the correlation information between instances. The Gaussian function of $\mathbf{A}$ is defined as follows:

$$\mathbf{A}_{ij} = \exp\left(\frac{||\mathbf{x}_i - \mathbf{x}_j||^2}{-2\sigma_1^2}\right) \tag{7}$$

where $i, j = 1, 2, \cdots, n$, $\mathbf{x}_i$ means the ith row of the data matrix $\mathbf{X}$, which indicates the ith sample, and σ_1 is a bandwidth parameter. According to the practical meaning, $0 < \mathrm{A}_{ij} \le 1$.

Similar to the construction of matrix $\mathbf{A}$, we also construct an affinity matrix $\mathbf{B}$ in feature space to represent the interconnection information between features. The Gaussian function of $\mathbf{B}$ is defined as follows:

$$\mathbf{B}_{ij} = \exp\left(\frac{||\mathbf{x}^i - \mathbf{x}^j||^2}{-2\sigma_2^2}\right) \tag{8}$$

where $i, j = 1, 2, \cdots, d$, $\mathbf{x}^i$ means the ith column of the data matrix $\mathbf{X}$, which indicates the ith feature. And σ_2 is a bandwidth parameter corresponding to matrix $\mathbf{B}$. According to the practical meaning, $0 < \mathbf{B}_{ij} \le 1$.

Next, we will carry out dual space latent representation learning. To learn the latent representation of data space from the affinity matrix $\mathbf{A}$, the following objective function needs to be solved:

$$\begin{aligned}&\arg\min_{\mathbf{V}} ||\mathbf{A} - \mathbf{V}\mathbf{V}^T||_F^2\\ &s.t. \mathbf{V} \ge 0\end{aligned} \tag{9}$$

where $\mathbf{V} \in \mathbb{R}^{n\times m}$ is the data latent representation matrix, m<n and m<d. Since $\mathbf{V}$ can in turn be regarded as a pseudo-label matrix to provide discriminative information for feature selection, we make m equal to sample category number c in the datasets.

Similarly, in the feature space, in order to learn the low-dimensional latent representation from the affinity matrix $\mathbf{B}$, the following objective function is formulated as follows:

$$\begin{aligned}&\arg\min_{\mathbf{W}} ||\mathbf{B} - \mathbf{W}\mathbf{W}^T||_F^2\\ &s.t. \mathbf{W} \ge 0\end{aligned} \tag{10}$$

where $\mathbf{W} \in \mathbb{R}^{d\times m}$ represents the feature latent representation matrix. Since $\mathbf{W}$ is unified with the transformation matrix, m is equal to c.

3.2. Objective function

DSLRL utilizes sparse transformation matrix to project the original data samples into low-dimensional space. Different from the conventional sparse regression method, our method uses both the low-dimensional latent representation matrices **V** and **W**. In the optimization process, the pseudo-class matrix **V** is regarded as the data in the low-dimensional space. At the same time, the low-dimensional latent representation **W** of feature space is related to the learning of transformation matrix. So, the transformation matrix can be denoted by **W**. Therefore, we obtain the following objective function:

$$\arg\min_{\mathbf{W},\mathbf{V}} ||\mathbf{XW}-\mathbf{V}||_F^2 \tag{11}$$

By optimizing the objective function, an appropriate transformation matrix **W** is obtained. Since **W** is used to calculate the weight of each feature, it is necessary to impose $l_{2,1}$-*norm* on the matrix **W** to ensure the row sparsity. Moreover, in order to avoid the emergence of trivial solutions, we apply the orthogonal constraint to matrix **W**. Therefore, the new objective function is expressed as:

$$\begin{aligned}&\arg\min_{\mathbf{W},\mathbf{V}} ||\mathbf{XW}-\mathbf{V}||_F^2+\alpha||\mathbf{W}||_{2,1}\\ &s.t.\mathbf{W^TW}=\mathbf{I}_m\end{aligned} \tag{12}$$

where $||\mathbf{W}||_{2,1}=\sum_{i=1}^{d}||\mathbf{w}_i||=\sum_{i=1}^{d}\sqrt{\sum_{j=1}^{m}\mathbf{W}_{ij}{}^2}$, and the parameter $\alpha>0$, which is used to control the sparseness of the model.

DSLRL embeds the latent representation learning based on data space and feature space into feature selection framework. Combining (9), (10), and (12), we obtain the final objective function of DSLRL:

$$\begin{aligned}\arg\min_{\mathbf{W},\mathbf{V}} ||\mathbf{XW}-\mathbf{V}||_F^2+\alpha||\mathbf{W}||_{2,1}+\beta||\mathbf{A}-\mathbf{VV}^T||_F^2+\gamma||\mathbf{B}-\mathbf{WW}^T||_F^2\\ s.t.\mathbf{V}\geq 0,\mathbf{W}\geq 0,\mathbf{W}^T\mathbf{W}=\mathbf{I}_m\end{aligned} \tag{13}$$

where the parameters $\beta>0,\gamma>0$. They are used to balance latent representation learning in data space and feature space, respectively.

By optimizing the objective function of DSLRL, the matrices **W** and **V** can be obtained. $\mathbf{w}_i$ represents each row of **W**, and $||\mathbf{w}_i||_2$ is used to evaluate the importance of each feature. The larger the evaluation value, the more important the i-th feature. The evaluation values of each feature are sorted in descending order and the top l features are selected to form a new data matrix **X**.

3.3. Optimization

Next the optimization process of the objective function (13) will be introduced in detail. Since the objective function (13) is non-convex in both **W** and **V** at the same time, it becomes difficult to solve. However, it is pleasing that it is convex for a single variable when fixing others, so we utilize an alternating iterative method [41] to optimize the objective function (13), which decomposes the whole optimization problem into small subproblems. In each subproblem, the new objective function is convex for one variable, so we can easily find the solution of the problem.

We construct the Lagrange function of (13) as follows:

$$\begin{aligned}L(\mathbf{W},\mathbf{V})=||\mathbf{XW}-\mathbf{V}||_F^2+\alpha||\mathbf{W}||_{2,1}+\beta||\mathbf{A}-\mathbf{VV}^T||_F^2+\gamma||\mathrm{B}\\ -\mathbf{WW}^T||_F^2+\lambda||\mathbf{W}^T\mathbf{W}-\mathbf{I}_m||_F^2+Tr(\mathbf{\Phi V}^T)+Tr(\mathbf{\Psi W}^T)\end{aligned} \tag{14}$$

where the parameters $\lambda>0$, $\mathbf{\Phi}$ and $\mathbf{\Psi}$ are Lagrange multipliers for non-negative constraints $\mathbf{V}\geq 0$ and $\mathbf{W}\geq 0$, respectively.

Before solving this problem, we need to introduce a diagonal matrix $\mathbf{H}\in\mathbb{R}^{d\times d}$, with the ith diagonal element as:

$$\mathbf{H}_{ii}=\frac{1}{2||\mathbf{w}_i||_2} \tag{15}$$

In order to avoid overflow, a small constant ε is usually introduced in the definition of the matrix **H** as follows:

$$\mathbf{H}_{ii}=\frac{1}{2max(||\mathbf{w}_i||_2,\varepsilon)} \tag{16}$$

Thus, $||\mathbf{W}||_{2,1}=Tr(\mathbf{W}^T\mathbf{HW})$. For an arbitrary matrix **M**, $||\mathbf{M}||_F^2=Tr(\mathbf{MM}^T)$. We rewrite all norm-bound terms into the form of traces, and obtain the following Lagrange function:

$$\begin{aligned}L(\mathbf{W},\mathbf{V})=Tr((\mathbf{XW}-\mathbf{V})(\mathbf{XW}-\mathbf{V})^T)+\alpha Tr(\mathbf{W}^T\mathbf{HW})\\ +\beta Tr((\mathbf{A}-\mathbf{VV}^T)(\mathbf{A}-\mathbf{VV}^T)^T)+\gamma Tr((\mathbf{B}-\mathbf{WW}^T)(\mathbf{B}-\mathbf{WW}^T)^T)\\ +\lambda Tr((\mathbf{W}^T\mathbf{W}-\mathbf{I}_m)(\mathbf{W}^T\mathbf{W}-\mathbf{I}_m)^T)+Tr(\mathbf{\Phi V}^T)+Tr(\mathbf{\Psi W}^T)\end{aligned} \tag{17}$$

The alternating and iterative method to solve the problem (15) contains two subproblems: we first fix the variable **V** to calculate the optimal, and then fix the variable **W** to calculate the optimal **V.**

A. Update **W**

When **V** is fixed, we take the partial derivative of the Lagrange function (17) with respect to **W**, and then have:

$$\begin{aligned}\frac{\partial L}{\partial \mathbf{W}}=&\mathbf{X}^T\mathbf{XW}-\mathbf{X}^T\mathbf{V}+\alpha\mathbf{HW}+2\gamma\mathbf{WW}^T\mathbf{W}-2\gamma\mathbf{BW}\\ &+2\lambda\mathbf{WW}^T\mathbf{W}-2\lambda\mathbf{W}+\mathbf{\Psi}\end{aligned} \tag{18}$$

According to the Karush–Kuhn–Tucker (KKT) condition [33], $\mathbf{\Psi}_{ij}\mathbf{W}_{ij}=0$, we obtain

$$\begin{aligned}&\left[\mathbf{X}^T\mathbf{XW}-\mathbf{X}^T\mathbf{V}+\alpha\mathbf{HW}+2\gamma\mathbf{WW}^T\mathbf{W}-2\gamma\mathbf{BW}+2\lambda\mathbf{WW}^T\mathbf{W}\right.\\ &\left.-2\lambda\mathbf{W}\right]_{ij}\mathbf{W}_{ij}=0\end{aligned} \tag{19}$$

Then the updating formula for **W** is as follows:

$$\mathbf{W}_{ij}\leftarrow\mathbf{W}_{ij}\frac{\left[\mathbf{X}^T\mathbf{V}+2\gamma\mathbf{BW}+2\lambda\mathbf{W}\right]_{ij}}{[\mathbf{X}^T\mathbf{XW}+\alpha\mathbf{HW}+2\gamma\mathbf{WW}^T\mathbf{W}+2\lambda\mathbf{WW}^T\mathbf{W}]_{ij}} \tag{20}$$

B. Update **V**

When **W** is fixed, we take the partial derivative of the Lagrange function (17) with respect to **V**, and then have:

$$\frac{\partial L}{\partial \mathbf{V}}=\mathbf{V}-\mathbf{XW}+2\beta\mathbf{VV}^T\mathbf{V}-2\beta\mathbf{AV}+\mathbf{\Phi} \tag{21}$$

According to the Karush–Kuhn–Tucker (KKT) condition [42], $\mathbf{\Phi}_{ij}\mathbf{V}_{ij}=0$, we have

$$\left[\mathbf{V}-\mathbf{XW}+2\beta\mathbf{VV}^T\mathbf{V}-2\beta\mathbf{AV}\right]_{ij}\mathbf{V}_{ij}=0 \tag{22}$$

Therefore, the updating formula for **V** is denoted as follows:

$$\mathbf{V}_{ij}\leftarrow\mathbf{V}_{ij}\frac{[\mathbf{XW}+2\beta\mathbf{AV}]_{ij}}{[\mathbf{V}+2\beta\mathbf{VV}^T\mathbf{V}]_{ij}} \tag{23}$$

Algorithm 1 summarizes the procedure of DSLRL.

3.4. Convergence analysis

The optimization of the objective function (13) involves two variables: **W**and **V**. Therefore, we need to prove that the objective function (13) is convergent under the updating rules (20) and (23), respectively.

Firstly, we prove that the objective function (13) is monotonically decreasing under the updating rules (23) for **V**.

Definition 1: In [43], if there is a function $J(r,r')$making $L(r)$ satisfies:

$$J\left(r,r'\right)\geq L(r),J(r,r)=L(r) \tag{24}$$

where, $J(r, r')$ is an auxiliary function of $L(r)$. Then $L(r)$ is monotonically decreasing under the updating formula:

$$r^{t+1} = \arg\min_{r} J\left(r, r^{t}\right) \tag{25}$$

When $\mathbf{W}$ is fixed, we only retain the terms containing $\mathbf{V}$ in the objective function (13), then obtain the following function:

$$\begin{aligned} L(\mathbf{V}) &= ||\mathbf{XW}-\mathbf{V}||_F^2+\beta||\mathbf{A}-\mathbf{VV}^T||_F^2 \\ &= Tr((\mathbf{XW}-\mathbf{V})(\mathbf{XW}-\mathbf{V})^T)+\beta Tr((\mathbf{A}-\mathbf{VV}^T)(\mathbf{A}-\mathbf{VV}^T)^T) \end{aligned} \tag{26}$$

The first-order and second-order partial derivatives of $L(\mathbf{V})$ with respect to $\mathbf{V}$ are

$$L_{ij}{}' = \left[\frac{\partial L(\mathbf{V})}{\partial \mathbf{V}}\right]_{ij} = \left[2\mathbf{V}-2\mathbf{XW}-4\beta\mathbf{AV}+4\beta\mathbf{VV}^T\mathbf{V}\right]_{ij} \tag{27}$$

$$L_{ij}{}'' = \left[2\mathbf{I}_n-4\beta\mathbf{A}+4\beta\mathbf{VV}^T\right]_{ii} \tag{28}$$

Lemma 1. The following function:

$$\begin{aligned} J_{ij}(\mathbf{V}_{ij}, \mathbf{V}_{ij}^{(t)}) &= L_{ij}(\mathbf{V}_{ij}^{(t)}) + L_{ij}{}'(\mathbf{V}_{ij}^{(t)})(\mathbf{V}_{ij}-\mathbf{V}_{ij}^{(t)}) \\ &+ \frac{\left[\mathbf{V}^{(t)}+2\beta\mathbf{V}^{(t)}\mathbf{V}^{(t)^T}\mathbf{V}^{(t)}\right]_{ij}}{\mathbf{V}_{ij}^{(t)}}(\mathbf{V}_{ij}-\mathbf{V}_{ij}^{(t)})^2 \end{aligned} \tag{29}$$

is the auxiliary function of $L_{ij}(\mathbf{V}_{ij})$.

Proof: The Taylor expansion of $L_{ij}(\mathbf{V}_{ij})$ is

$$L_{ij}(\mathbf{V}_{ij}) = L_{ij}(\mathbf{V}_{ij}^{(t)})+L_{ij}{}'(\mathbf{V}_{ij}^{(t)})(\mathbf{V}_{ij}-\mathbf{V}_{ij}^{(t)}) + \frac{1}{2}L_{ij}{}''(\mathbf{V}_{ij}^{(t)})(\mathbf{V}_{ij}-\mathbf{V}_{ij}^{(t)})^2 \tag{30}$$

According to (29) and (30), $J(\mathbf{V}_{ij}, \mathbf{V}_{ij}^{(t)}) \geq L(\mathbf{V}_{ij})$ is equivalent to

$$\frac{\left[\mathbf{V}^{(t)}+2\beta\mathbf{V}^{(t)}\mathbf{V}^{(t)^T}\mathbf{V}^{(t)}\right]_{ij}}{\mathbf{V}_{ij}^{(t)}} \geq \frac{1}{2}\left[2\mathbf{I}_n-4\beta\mathbf{A}+4\beta\mathbf{V}^{(t)}\mathbf{V}^{(t)^T}\right]_{ii} \tag{31}$$

Due to $\beta \geq 0, \mathbf{A}_{ij} \geq 0$, it is obvious that

$$\begin{aligned} &\left[\mathbf{V}^{(t)}+2\beta\mathbf{V}^{(t)}\mathbf{V}^{(t)^T}\mathbf{V}^{(t)}\right]_{ij} = \sum_{k=1}^{n}\left[\mathbf{I}_n+2\beta\mathbf{V}^{(t)}\mathbf{V}^{(t)^T}\right]_{ik}\mathbf{V}_{kj}^{(t)} \\ &\geq \sum_{k=1}^{n}\left[\mathbf{I}_n-2\beta\mathbf{A}+2\beta\mathbf{V}^{(t)}\mathbf{V}^{(t)^T}\right]_{ik}\mathbf{V}_{kj}^{(t)} \\ &\geq \left[\mathbf{I}_n-2\beta\mathbf{A}+2\beta\mathbf{V}^{(t)}\mathbf{V}^{(t)^T}\right]_{ii}\mathbf{V}_{ij}^{(t)} \end{aligned} \tag{32}$$

Therefore, (31) holds, and $J(\mathbf{V}_{ij}, \mathbf{V}_{ij}^{(t)}) \geq L(\mathbf{V}_{ij})$ holds. When $\mathbf{V}_{ij}$=$\mathbf{V}_{ij}^{(t)}$, according to (29), $J(\mathbf{V}_{ij}, \mathbf{V}_{ij})$=$L(\mathbf{V}_{ij})$ also holds. Thus, the function $J(\mathbf{V}_{ij}, \mathbf{V}_{ij}^{(t)})$ in *Lemma 1* satisfies (24).

Next, we prove that the variable $\mathbf{V}$ conforms to the updating formula (25) that makes L monotonically non-increasing.

Proof: Replace $J(\mathbf{V}_{ij}, \mathbf{V}_{ij}^{(t)})$ in (29) into (25), and let $\frac{\partial J(\mathbf{V}_{ij}, \mathbf{V}_{ij}^{(t)})}{\partial \mathbf{V}_{ij}}$=0, then we obtain:

$$\begin{aligned} \mathbf{V}_{ij}^{(t+1)} &= \mathbf{V}_{ij}^{(t)} - \mathbf{V}_{ij}^{(t)}\frac{L_{ij}{}'(\mathbf{V}_{ij}^{(t)})}{2\left[\mathbf{V}^{(t)}+2\beta\mathbf{V}^{(t)}\mathbf{V}^{(t)^T}\mathbf{V}^{(t)}\right]_{ij}} \\ &= \mathbf{V}_{ij}^{(t)}\frac{\left[\mathbf{XW}+2\beta\mathbf{AV}^{(t)}\right]_{ij}}{\left[\mathbf{V}^{(t)}+2\beta\mathbf{V}^{(t)}\mathbf{V}^{(t)^T}\mathbf{V}^{(t)}\right]_{ij}} \end{aligned} \tag{33}$$

Obviously, formula (33) is equivalent to the updating rule (23) for $\mathbf{V}$. Therefore, it can be proved that L_{ij} is monotonically non-increasing under (23).

The proof of the convergence of the objective function under the updating rule of $\mathbf{W}$ is similar to that of $\mathbf{V}$. When $\mathbf{V}$ is fixed, the terms containing $\mathbf{W}$ in the objective function (13) are

$$\begin{aligned} L(\mathbf{W}) &= ||\mathbf{XW}-\mathbf{V}||_F^2+\alpha||\mathbf{W}||_{2,1}+\gamma||\mathrm{B}-\mathbf{WW}^T||_F^2+\lambda||\mathbf{W}^T\mathbf{W}-\mathbf{I}_m||_F^2 \\ &= Tr((\mathbf{XW}-\mathbf{V})(\mathbf{XW}-\mathbf{V})^T)+\alpha Tr(\mathbf{W}^T\mathbf{HW}) \\ &+\gamma Tr((\mathbf{B}-\mathbf{WW}^T)(\mathbf{B}-\mathbf{WW}^T)^T)+\lambda Tr((\mathbf{W}^T\mathbf{W}-\mathbf{I}_m)(\mathbf{W}^T\mathbf{W}-\mathbf{I}_m)^T) \end{aligned} \tag{34}$$

The first-order and second-order partial derivatives of $L(\mathbf{W})$ with respect to $\mathbf{W}$ are

$$\begin{aligned} L_{ij}{}' &= \left[\tfrac{\partial L(\mathbf{W})}{\partial \mathbf{W}}\right]_{ij} = \left[2\mathbf{X}^T(\mathbf{XW}-\mathbf{V})+2\alpha\mathbf{HW}-4\gamma(\mathbf{B}-\mathbf{WW}^T)\mathbf{W}+4\lambda\mathbf{W}(\mathbf{W}^T\mathbf{W}-\mathbf{I}_m)\right]_{ij} \\ &= \left[2\mathbf{X}^T\mathbf{XW}-2\mathbf{X}^T\mathbf{V}+2\alpha\mathbf{HW}-4\gamma\mathbf{BW}+4\gamma\mathbf{WW}^T\mathbf{W}+4\lambda\mathbf{WW}^T\mathbf{W}-4\lambda\mathbf{W}\right]_{ij} \end{aligned} \tag{35}$$

$$L_{ij}{}'' = [2\mathbf{X}^T\mathbf{X}+2\alpha\mathbf{H}-4\gamma\mathbf{B}+4\gamma\mathbf{WW}^T+4\lambda\mathbf{WW}^T-4\lambda\mathbf{I}_d]_{ii} \tag{36}$$

Lemma 2. The following function:

$$\begin{aligned} &J_{ij}(\mathbf{W}_{ij}, \mathbf{W}_{ij}^{(t)}) \\ &= L_{ij}(\mathbf{W}_{ij}^{(t)}) + L_{ij}{}'(\mathbf{W}_{ij}^{(t)})(\mathbf{W}_{ij}-\mathbf{W}_{ij}^{(t)}) \\ &+ \frac{\left[\mathbf{X}^T\mathbf{XW}^{(t)}+\alpha\mathbf{HW}^{(t)}+2\gamma\mathbf{W}^{(t)}\mathbf{W}^{(t)^T}\mathbf{W}^{(t)}+2\lambda\mathbf{W}^{(t)}\mathbf{W}^{(t)^T}\mathbf{W}^{(t)}\right]_{ij}}{\mathbf{W}_{ij}^{(t)}}(\mathbf{W}_{ij}-\mathbf{W}_{ij}^{(t)})^2 \end{aligned} \tag{37}$$

is the auxiliary function of $L_{ij}(\mathbf{W}_{ij})$.

Proof: The Taylor expansion of $L_{ij}(\mathbf{W}_{ij})$ is

$$\begin{aligned} L_{ij}(\mathbf{W}_{ij}) &= L_{ij}(\mathbf{W}_{ij}^{(t)}) + L_{ij}{}'(\mathbf{W}_{ij}^{(t)})(\mathbf{W}_{ij}-\mathbf{W}_{ij}^{(t)}) \\ &+ \frac{1}{2}L_{ij}{}''(\mathbf{W}_{ij}^{(t)})(\mathbf{W}_{ij}-\mathbf{W}_{ij}^{(t)})^2 \end{aligned} \tag{38}$$

According to (37) and (38), $J(\mathbf{W}_{ij}, \mathbf{W}_{ij}^{(t)}) \geq L(\mathbf{W}_{ij})$ is equivalent to

$$\begin{aligned} &\frac{\left[\mathbf{X}^T\mathbf{XW}^{(t)}+\alpha\mathbf{HW}^{(t)}+2\gamma\mathbf{W}^{(t)}\mathbf{W}^{(t)^T}\mathbf{W}^{(t)}+2\lambda\mathbf{W}^{(t)}\mathbf{W}^{(t)^T}\mathbf{W}^{(t)}\right]_{ij}}{\mathbf{W}_{ij}^{(t)}} \\ &\geq \tfrac{1}{2}\left[2\mathbf{X}^T\mathbf{X}+2\alpha\mathbf{H}-4\gamma\mathbf{B}+4\gamma\mathbf{WW}^T+4\lambda\mathbf{WW}^T-4\lambda\mathbf{I}_d\right]_{ii} \end{aligned} \tag{39}$$

Due to $\gamma \geq 0, \lambda \geq 0, \mathbf{B}_{ij} \geq 0, \mathbf{I}_{ij} \geq 0$, it is obvious that

$$\begin{aligned} &\left[\mathbf{X}^T\mathbf{XW}^{(t)}+\alpha\mathbf{HW}^{(t)}+2\gamma\mathbf{W}^{(t)}\mathbf{W}^{(t)^T}\mathbf{W}^{(t)}+2\lambda\mathbf{W}^{(t)}\mathbf{W}^{(t)^T}\mathbf{W}^{(t)}\right]_{ij} \\ &= \sum_{s=1}^{d}\left[\mathbf{X}^T\mathbf{X}+\alpha\mathbf{H}+2\gamma\mathbf{W}^{(t)}\mathbf{W}^{(t)^T}+2\lambda\mathbf{W}^{(t)}\mathbf{W}^{(t)^T}\right]_{is}\mathbf{W}_{sj}^{(t)} \\ &\geq \sum_{s=1}^{d}\left[\mathbf{X}^T\mathbf{X}+\alpha\mathbf{H}-2\gamma\mathbf{B}+2\gamma\mathbf{W}^{(t)}\mathbf{W}^{(t)^T}+2\lambda\mathbf{W}^{(t)}\mathbf{W}^{(t)^T}-2\lambda\mathbf{I}_d\right]_{is}\mathbf{W}_{sj}^{(t)} \\ &\geq \left[\mathbf{X}^T\mathbf{X}+\alpha\mathbf{H}-2\gamma\mathbf{B}+2\gamma\mathbf{W}^{(t)}\mathbf{W}^{(t)^T}+2\lambda\mathbf{W}^{(t)}\mathbf{W}^{(t)^T}-2\lambda\mathbf{I}_d\right]_{ii}\mathbf{W}_{ij}^{(t)} \end{aligned} \tag{40}$$

Therefore, (39) holds, and $J(\mathbf{W}_{ij}, \mathbf{W}_{ij}^{(t)}) \geq L(\mathbf{W}_{ij})$ holds. When $\mathbf{W}_{ij}$=$\mathbf{W}_{ij}^{(t)}$, according to (37), $J(\mathbf{W}_{ij}, \mathbf{W}_{ij})$=$L(\mathbf{W}_{ij})$ also holds. Thus, the function $J(\mathbf{W}_{ij}, \mathbf{W}_{ij}^{(t)})$ in *Lemma 2* satisfies (24).

Combine (37) and (25), and let $\frac{\partial J(\mathbf{W}_{ij}, \mathbf{W}_{ij}^{(t)})}{\partial \mathbf{W}_{ij}}$=0, then we obtain

$$\begin{aligned} \mathbf{W}_{ij}^{(t+1)} &= \mathbf{W}_{ij}^{(t)} - \mathbf{W}_{ij}^{(t)}\frac{L_{ij}{}'(\mathbf{W}_{ij}^{(t)})}{2\left[\mathbf{X}^T\mathbf{XW}^{(t)}+\alpha\mathbf{HW}^{(t)}+2\gamma\mathbf{W}^{(t)}\mathbf{W}^{(t)^T}\mathbf{W}^{(t)}+2\lambda\mathbf{W}^{(t)}\mathbf{W}^{(t)^T}\mathbf{W}^{(t)}\right]_{ij}} \\ &= \mathbf{W}_{ij}^{(t)}\frac{\left[\mathbf{X}^T\mathbf{V}+2\gamma\mathbf{BW}^{(t)}+2\lambda\mathbf{W}^{(t)}\right]_{ij}}{\left[\mathbf{X}^T\mathbf{XW}^{(t)}+\alpha\mathbf{HW}^{(t)}+2\gamma\mathbf{W}^{(t)}\mathbf{W}^{(t)^T}\mathbf{W}^{(t)}+2\lambda\mathbf{W}^{(t)}\mathbf{W}^{(t)^T}\mathbf{W}^{(t)}\right]_{ij}} \end{aligned} \tag{41}$$

Obviously, the formula (41) is equivalent to the updating rule (20) for $\mathbf{W}$. Thus, it can also be proved that L_{ij} is monotonically non-increasing under (20). In summary, the objective function of DSLRL is convergent under the updating rules (20) and (23).

Table 2
Characteristics of twelve datasets

Datasets	Instance	Feature	Class	Type
Yale	165	1024	15	Face images
warpPIE10P	210	2420	10	Face images
AT&T	400	10304	40	Face images
COIL20	1440	1024	20	Object images
Isolet	1560	617	26	Speech Signal
CLL_SUB_111	111	11340	3	Biological microarray
Mnist	5000	784	10	Digital image
PIE_pose27	2856	1024	68	Face images
Optdigit	3823	64	10	Digital image
Yale64	165	4096	15	Face images
GLIOMA	50	4434	4	Biological microarray
TOX-171	171	5748	4	Biological microarray

4. Simulation results and the analyses

In this section, we will compare DSLRL and seven state-of-the-art algorithms on the same datasets, and utilize the k-means clustering method [44] to test the performance of all the algorithms. Then, we illustrate the parameter sensitivity analysis of DSLRL.

4.1. Compared algorithms and Datasets

In order to illustrate the effectiveness of DSLRL, we compare DSLRL with several state-of-the-art unsupervised feature selection algorithms, including LapScor [12], MCFS [20], JELSR [22], SOGFS [23], ADGCF$_{FS}$ [33], URAFS [24] and LRLMR [29].

Our experiment uses twelve datasets, including image datasets (i.e., Yale, warpPIE10P, AT&T, COIL20, Mnist, PIE_pose27, Optdigit, Yale64) [38,45,46], speech signal dataset (i.e., Isolet) [47], and biology datasets (i.e., CLL_SUB_111, GLIOMA, TOX-171) [29]. Table 2 shows the details of these datasets.

4.2. Evaluation metrics

In order to evaluate the clustering results of all algorithms, we choose two popular metrics: clustering accuracy (ACC) [32] and normalized mutual information (NMI) [48]. The higher the values of ACC and NMI are, the better the clustering result. Therefore, ACC and NMI can reflect the feature selection effectiveness of all algorithms.

ACC is defined as:

$$\mathrm{ACC} = \frac{1}{n}\sum_{i=1}^{n}\delta(c_i, map(g_i)) \tag{42}$$

where c_i and $g_i$respectively denote the clustering label and the true label of $\mathbf{x}_i$.$map(\cdot)$ is an optimal mapping function, which utilizes Hungarian method [49] to match clustering labels with true labels. $\delta(c_i, g_i)$ is an indicator function, if c_i=g_i, $\delta(c_i, g_i)$=1, otherwise, $\delta(c_i, g_i)$= 0.

NMI is defined as:

$$\mathrm{NMI} = \frac{MI(C, \widetilde{C})}{max(H(C), H(\widetilde{C}))} \tag{43}$$

where C and $\widetilde{C}$ respectively represent the clustering labels and the true labels. $H(C)$ is the entropy of C, and $H(\widetilde{C})$ is the entropy of $\widetilde{C}$. $MI(C, \widetilde{C})$ is the information entropy between C and $\widetilde{C}$

$$MI(C, \widetilde{C}) = \sum_{c_i \in C, \widetilde{c}_j \in \widetilde{C}} p(c_i, \widetilde{c}_j) log \frac{p(c_i, \widetilde{c}_j)}{p(c_i)p(\widetilde{c}_j)} \tag{44}$$

where $p(c_i)$ and $p(\widetilde{c}_j)$ respectively indicate the probabilities that a sample belongs to the clusters c_i and $\widetilde{c}_j$. $p(c_i, \widetilde{c}_j)$ is the joint probability that a sample simultaneously belongs to the clusters c_i and $\widetilde{c}_j$.

It is worth noting that ACC and NMI are two different evaluation metrics for clustering results. ACC reflects the accuracy of the clustering results, while NMI reflects the consistency between the clustering results and the true labels. For the clustering results on the same dataset, ACC and NMI may not reach the highest at the same time.

4.3. Experimental settings

The range of some parameters in our experiment needs to be set. For LapScor, MCFS, JELSR, URAFS, ADGCF$_{FS}$, SOGFS and LRLMR, we fix the neighborhood size to 5. As for DSLRL, we optimize the low-dimensional latent representation matrix $\mathbf{V}$ infinitely close to the ideal label matrix to provide discriminative information for feature selection, so that it can better provide discriminative information for feature selection. Therefore, m is equal to c that is the number of clusters. The parameter *Niter* representing the maximum number of iterations is 50. The parameters $\alpha, \beta, \gamma, \lambda$ are searched from $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10^{1}, 10^{2}, 10^{3}\}$. For the Optdigit dataset, we tune the feature selection parameter l in the range of {20, 30, 40}, and for the remaining datasets, l varies in {20, 30, 40, 50, 60, 70, 80, 90, 100}. Because the results of the k-means clustering method are dependent on initialization, we repeat the clustering for 20 runs independently and take the average of ACC and NMI, respectively.

4.4. Clustering results and analysis

Table 3 shows the average and standard deviation of the clustering accuracy (ACC) of all algorithms for feature selection on different datasets. Table 4 summarizes the average and standard deviation of the normalized mutual information (NMI) for the same experiments. In both tables, the best results are highlighted in bold, and the number of selected features is labeled.

From Table 3, we can see that the ACC results of DSLRL on the 12 datasets are better than others. As can be seen from Table 4, the NMI results of DSLRL are better than 7 comparison algorithms on different datasets. It should be noted that the clustering results of DSLRL on CLL_SUB_111 dataset are superior. In CLL_SUB_111 dataset, the number of features is much larger than the number of samples, and the number of clusters is relatively small. Such datasets usually have many features that are redundant, or even represent noise, they will affect the effect of feature selection. Different from other methods, DSLRL introduces dual space latent representation learning, where the data latent representation matrix describes the assignment information between instances and clusters, and the feature latent representation matrix records the correlation between features and clusters. Both of them are applied to the regression function, which improved the effectiveness of feature selection and facilitate subsequent data processing.

In order to better illustrate that the improvement of the clustering results of DSLRL in Tables 3 and 4 is obvious, we perform a statistical analysis on the results of DSLRL and comparison algorithms. Specifically, the paired t-test is used. Each algorithm needs to repeat clustering 20 times independently to obtain the average results in Tables 3 and 4. These 20 results are used as samples for paired t-test, and the significance level parameter *alpha*=0.05. Observing h and p which obtained from the statistical experiment, h=0 indicates that the null hypothesis cannot be rejected at the 5% significance level. On the contrary, h=1 indicates that the null hypothesis can be rejected at the 5% level. And p represents the p-value, which shows the significance level. When h=1 and the value of p is small, it is generally considered that there is a difference between the two samples, which indicates that the result of DSLRL is significantly improved. Tables 5-6 are the paired t-test of DSLRL and each comparison algorithm on all datasets.

Table 3
Best clustering results (ACC±STD%) of different algorithms on twelve datasets.

Datasets	LapScor	MCFS	JELSR	SOGFS	$ADGCF_{FS}$	URAFS	LRLMR	DSLRL
Yale	42.53±1.83(30)	43.23±2.27(40)	41.10±2.94(60)	44.29±3.28(20)	37.24±1.65(20)	37.47±2.26(50)	43.67±2.89(20)	**46.94±3.57**(50)
warpPIE10P	29.52±0.85(20)	36.10±2.74(20)	46.36±2.89(20)	51.43±2.36(20)	45.16±2.21(60)	31.80±1.88(20)	47.15±2.46(20)	**55.06±2.89**(20)
AT&T	47.16±1.91(50)	55.53±2.44(90)	56.58±2.71(100)	54.46±2.15(30)	45.96±1.88(100)	58.39±2.89(90)	57.07±2.26(100)	**62.17±2.72**(70)
COIL20	59.95±2.74(100)	65.08±2.58(70)	63.30±2.87(90)	48.70±1.79(70)	51.81±2.41(80)	62.08±2.85(100)	64.10±2.61(100)	**66.86±3.07**(100)
Isolet	55.69±2.17(100)	57.07±1.96(100)	58.15±1.93(100)	49.19±2.11(100)	53.38±1.11(90)	58.83±2.14(100)	62.37±2.95(100)	**65.10±2.19**(100)
CLL_SUB_111	55.66±1.67(70)	54.05±1.12(30)	54.49±1.34(20)	58.55±0.67(50)	57.83±0.97(100)	58.55±2.57(20)	51.15±2.51(40)	**63.36±1.56**(100)
Mnist	45.14±0.85(80)	47.56±1.34(100)	50.07±1.46(100)	47.44±1.30(100)	43.95±0.28(100)	50.68±0.84(100)	51.55±2.02(100)	**51.98±1.01**(100)
PIE_pose27	30.26±0.84(40)	20.05±0.77(50)	30.82±0.83(50)	19.08±0.69(100)	35.52±0.77(30)	30.68±0.81(20)	30.75±0.84(30)	**37.92±1.32**(80)
Optdigit	81.28±1.72(40)	78.35±0.68(40)	81.59±2.98(40)	78.62±0.81(40)	81.87±0.62(40)	80.13±2.05(40)	81.28±0.33(30)	**82.62±3.16**(40)
Yale64	51.10±2.77(70)	50.01±3.06(30)	51.45±3.18(100)	45.34±3.27(100)	49.51±3.15(90)	44.00±3.03(50)	46.41±2.68(90)	**52.30±3.67**(100)
GLIOMA	56.46±1.58(100)	57.18±7.79(80)	56.40±3.53(100)	59.56±2.29(90)	65.30±3.38(100)	55.20±4.28(100)	61.00±5.40(90)	**70.50±4.62**(100)
TOX-171	42.67±0.12(20)	41.05±4.18(90)	44.23±1.56(100)	40.09±2.18(100)	39.23±0.73(80)	46.14±3.77(100)	47.54±1.71(100)	**48.48±1.55**(90)

Table 4
Best clustering results (NMI±STD%) of different algorithms on twelve datasets.

Datasets	LapScor	MCFS	JELSR	SOGFS	$ADGCF_{FS}$	URAFS	LRLMR	DSLRL
Yale	48.64±1.55(30)	47.62±2.26(100)	47.86±2.27(60)	49.67±2.53(20)	42.98±1.87(50)	42.64±2.02(50)	49.34±1.82(20)	**53.11±2.53**(50)
warpPIE10P	25.94±1.19(20)	40.20±3.19(20)	49.68±2.40(20)	53.62±1.77(20)	49.79±1.93(60)	30.18±2.61(20)	56.23±2.29(20)	**56.36±2.68**(20)
AT&T	71.13±1.04(60)	72.44±1.19(90)	74.19±1.67(90)	73.62±1.11(30)	67.91±1.01(100)	76.92±1.23(90)	75.46±1.09(100)	**79.90±1.43**(80)
COIL20	69.59±1.24(100)	74.53±1.39(70)	73.46±1.32(90)	61.24±1.19(70)	59.56±1.20(80)	73.00±1.58(100)	75.76±1.31(100)	**77.21±1.69**(100)
Isolet	69.63±0.86(100)	70.20±0.86(100)	71.33±0.75(100)	65.68±1.01(100)	67.38±0.80(90)	73.13±1.07(100)	74.15±2.58(100)	**74.92±1.17**(100)
CLL_SUB_111	17.55±1.59(80)	26.21±1.46(30)	26.31±2.23(20)	26.25±0.68(50)	25.70±0.68(100)	19.65±1.12(20)	16.28±1.82(40)	**31.07±1.06**(30)
Mnist	39.85±0.31(100)	42.90±0.47(100)	42.70±0.68(100)	37.88±0.61(100)	35.97±0.14(100)	39.64±0.42(80)	42.79±0.77(100)	**43.35±0.38**(100)
PIE_pose27	54.05±0.52(40)	41.78±0.52(50)	58.35±0.59(50)	42.76±0.49(100)	58.01±0.49(30)	56.90±0.55(20)	55.65±0.58(30)	**64.10±0.69**(80)
Optdigit	75.69±0.85(40)	73.75±0.36(40)	74.93±1.96(40)	72.50±0.71(40)	72.06±0.48(40)	74.76±4.06(40)	74.31±0.26(30)	**75.99±0.51**(40)
Yale64	58.34±1.87(70)	55.97±1.89(30)	55.11±2.18(100)	52.30±2.29(100)	54.12±1.81(90)	49.43±1.66(50)	52.30±1.95(90)	**59.78±2.61**(100)
GLIOMA	49.94±1.37(100)	36.90±6.66(80)	40.71±5.48(100)	51.64±4.28(60)	50.11±2.11(100)	37.59±6.25(100)	50.41±2.69(90)	**52.73±6.15**(100)
TOX-171	14.38±0.24(20)	12.97±4.71(90)	15.16±2.36(100)	11.91±0.99(100)	13.18±0.38(90)	21.09±6.22(100)	15.92±1.51(100)	**26.06±1.15**(90)

Table 5
The paired t-test result of ACC of DSLRL and comparison algorithm on all datasets

Datasets	LapScor		MCFS		JELSR		SOGFS		$ADGCF_{FS}$		URAFS		LRLMR	
	p	*h*	*p*	*h*	*p*	*h*	*p*	*h*	*p*	*h*	*p*	*h*	*p*	*h*
Yale	1.5977e-05	1	0.0025	1	6.6494e-08	1	3.7687e-07	1	2.5486e-02	1	0.0076	1	4.6980e-04	1
warpPIE10P	6.2283e-43	1	5.6499e-28	1	2.1852e-15	1	0.0063	1	4.3561e-31	1	1.1631e-23	1	2.7032e-11	1
AT&T	1.0888e-32	1	3.8543e-15	1	2.3548e-16	1	7.9700e-14	1	6.2541e-40	1	4.1373e-18	1	1.2564e-10	1
COIL20	5.7859e-14	1	0.5698	0	1.7728e-05	1	1.6530e-35	1	3.4645e-29	1	1.3724e-07	1	0.0519	0
Isolet	1.0734e-26	1	1.0310e-13	1	2.9988e-19	1	5.9270e-39	1	4.1523e-28	1	4.0867e-15	1	3.4358e-05	1
CLL_SUB_111	1.1169e-31	1	6.6579e-21	1	7.4568e-19	1	6.1147e-10	1	5.2156e-15	1	5.4621e-10	1	2.1543e-18	1
Mnist	7.5350e-38	1	6.4562e-12	1	2.4816e-08	1	3.7998e-25	1	1.2487e-40	1	0.3839	0	0.6653	0
PIE_pose27	4.2718e-21	1	2.7644e-42	1	3.6556e-37	1	1.3429e-41	1	3.7845e-09	1	4.2136e-32	1	9.1111e-29	1
Optdigit	0.4069	0	1.1549e-12	1	0.2248	0	1.3265e-23	1	0.0016	1	2.4235e-21	1	1.1391e-09	1
Yale64	0.6510	0	3.7278e-24	1	0.1849	0	6.7104e-14	1	5.6594e-11	1	7.4499e-18	1	5.5685e-10	1
GLIOMA	4.3265e-32	1	2.6542e-30	1	4.3684e-35	1	3.3941e-25	1	3.2145e-05	1	4.3521e-41	1	2.1545e-08	1
TOX-171	3.5487e-16	1	4.6584e-23	1	5.6842e-10	1	2.6548e-29	1	1.5487e-34	1	1.2645e-03	1	0.0039	1

Table 6
The paired t-test result of NMI of DSLRL and comparison algorithm on all datasets

Datasets	LapScor		MCFS		JELSR		SOGFS		$ADGCF_{FS}$		URAFS		LRLMR	
	p	*h*	*p*	*h*	*p*	*h*	*p*	*h*	*p*	*h*	*p*	*h*	*p*	*h*
Yale	3.0673e-10	1	1.1216e-10	1	3.0801e-11	1	4.3344e-05	1	6.3254e-35	1	2.9570e-28	1	5.0405e-11	1
warpPIE10P	1.4477e-43	1	4.5204e-31	1	6.6930e-16	1	8.7164e-06	1	3.2456e-15	1	1.7454e-20	1	0.0287	1
AT&T	2.9279e-37	1	6.8989e-25	1	5.3264e-21	1	1.3358e-27	1	1.3256e-40	1	4.3251e-06	1	6.3254e-09	1
COIL20	5.2313e-35	1	3.7378e-05	1	2.1435e-14	1	4.4759e-36	1	2.3564e-42	1	7.4121e-16	1	5.0157e-06	1
Isolet	1.0908e-36	1	8.2349e-39	1	6.1528e-21	1	2.4990e-23	1	2.1564e-19	1	2.4949e-10	1	0.0220	1
CLL_SUB_111	2.8837e-15	1	5.5705e-06	1	4.3251e-06	1	6.8170e-06	1	3.8461e-06	1	2.3564e-12	1	5.1254e-16	1
Mnist	1.9863e-39	1	1.1458e-03	1	1.0511e-09	1	2.2851e-40	1	4.2356e-43	1	7.5211e-37	1	3.3585e-04	1
PIE_pose27	3.1037e-25	1	9.3674e-40	1	1.0266e-39	1	2.4650e-43	1	5.4525e-38	1	1.1723e-44	1	1.8323e-31	1
Optdigit	0.3874	0	1.3091e-20	1	0.0011	1	1.2543e-07	1	1.3380e-40	1	5.5413e-14	1	8.8391e-29	1
Yale64	0.4627	0	1.1591e-04	1	9.3160e-04	1	2.1254e-24	1	2.5741e-15	1	3.7642e-35	1	4.8803e-25	1
GLIOMA	6.9584e-10	1	4.3518e-40	1	2.6978e-30	1	0.6738	0	4.6514e-06	1	6.9572e-38	1	2.7594e-07	1
TOX-171	4.6579e-37	1	5.9847e-42	1	4.6259e-34	1	5.6245e-45	1	3.6847e-40	1	2.6589e-13	1	4.8476e-40	1

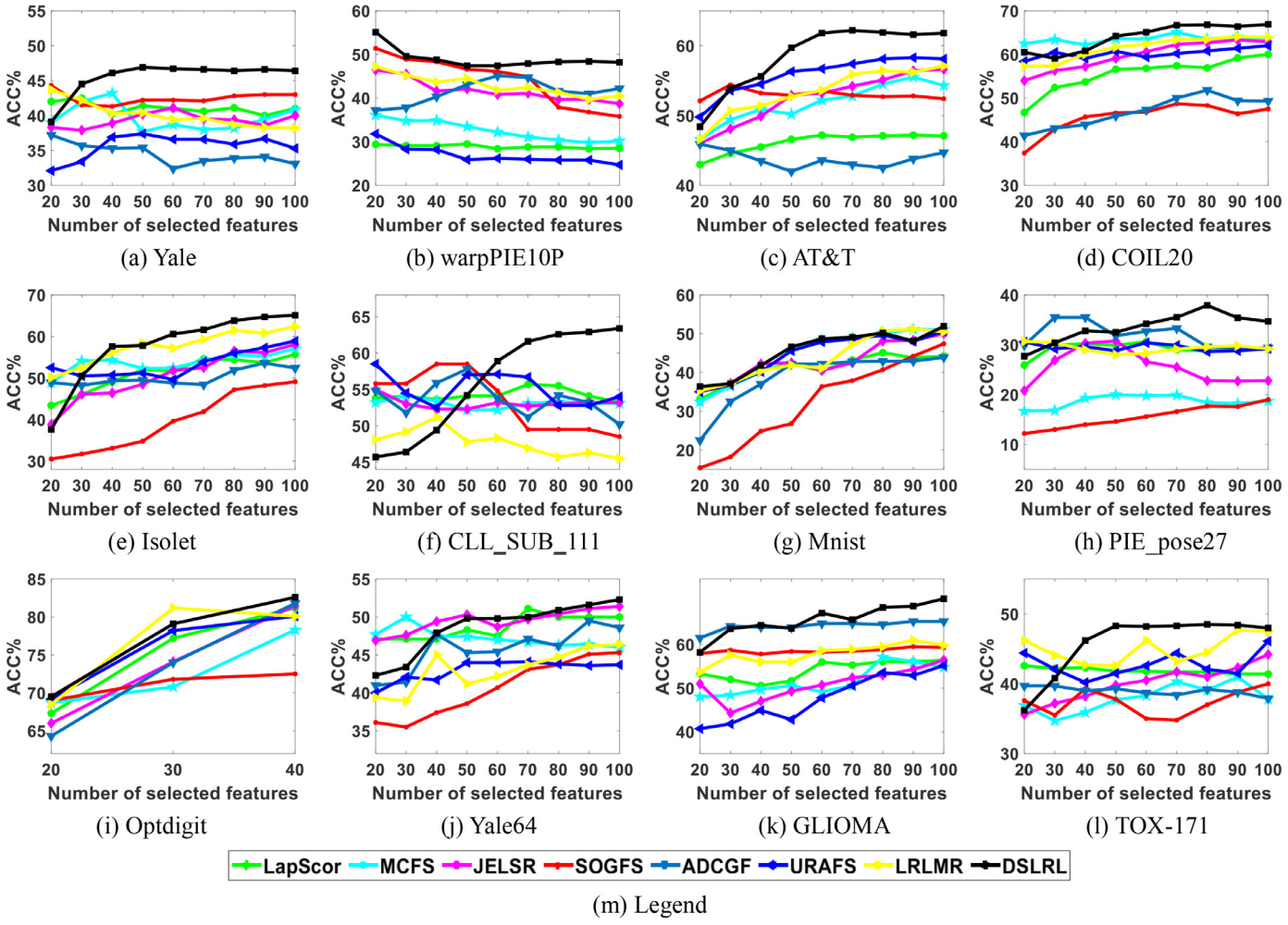

Fig. 1. The ACC of all the algorithms for selecting different numbers of features on the twelve datasets.

It can be seen from Table 5 that in the paired t-test of SOGFS, h=1 and all p-values are small, indicating that the ACC values of DSLRL and SOGFS are quite different. In other paired t-tests, h=1 and p-value is small on most datasets. And h=0 on a few datasets, which indicates that the ACC values of DSLRL are not obviously improved compared to other algorithms. In general, the ACC of DSLRL is significantly improved in most cases.

From Table 6, except for the results of the Optdigit and Yale64 datasets in the LapScor paired t-test and the results of the GLIOMA datasets in the SOGFS paired t-test, the rest h=1 and the p-value is small. It shows that the NMI of DSLRL and the comparison algorithms are obviously different, which means that the NMI obtained by DSLRL has a significant improvement. Tables 5-6 display that the clustering results of DSLRL are significantly improved compared with other algorithms, which verify the superiority of DSLRL.

In order to study the effect of the number of selected features on the proposed algorithm, this experiment shows the performance of DSLRL and seven comparison algorithms when different numbers of features are selected. Fig. 1 displays the clustering accuracy (ACC) of all algorithms for selecting different numbers of features on twelve datasets. In Figs. 1 and 2, the abscissa represents the number of selected features, the ordinate denotes ACC and NMI, respectively.

In Fig. 1, we use eight curves with different colors and shapes to express the eight feature selection algorithms, respectively, where the black curve represents DSLRL. From Fig. 1, we can see that on the warpPIE10P dataset, the black curve of DSLRL is always above other curves. It indicates that the ACC of DSLRL is much higher than the comparison algorithms on this dataset. On Yale, AT&T, COIL20, Isolet, GLIOMA and TOX-171 datasets, most points of the black curve are at the top. On the remaining four data sets, the black curve of DSLRL is located above most of the curves, and the highest point of the black curve is above the other curves. In short, the results of DSLRL for feature selection are better than other algorithms. The main reason is that, we embed latent representation learning and sparse learning in the framework of unsupervised feature selection. During the optimization process, the low-dimensional latent representation matrix of data space provides clustering information for sparse learning, and the sparse transformation matrix is unified with the latent representation matrix of feature space. The two courses interact with each other and improve the performance of DSLRL.

Fig. 2 demonstrates the normalized mutual information (NMI) for the same experiments.

As we can see from Fig. 2, on the warpPIE10P dataset, the black curves of DSLRL are above other curves. On the Yale, AT&T, PIE_pose27, Yale64 and TOX-171 datasets, most points of the black curve are higher than the points of the other curves. On the COIL20, Isolet, Minist, Optdigit and GLIOMA datasets, the black curve of DSLRL is located above most curves as a whole, and the best NMI of DSLRL is better than comparison algorithms. Overall, our proposed DSLRL improves the effect of clustering experiments. In summary, it can be proven that DSLRL has better performance than the other algorithms.

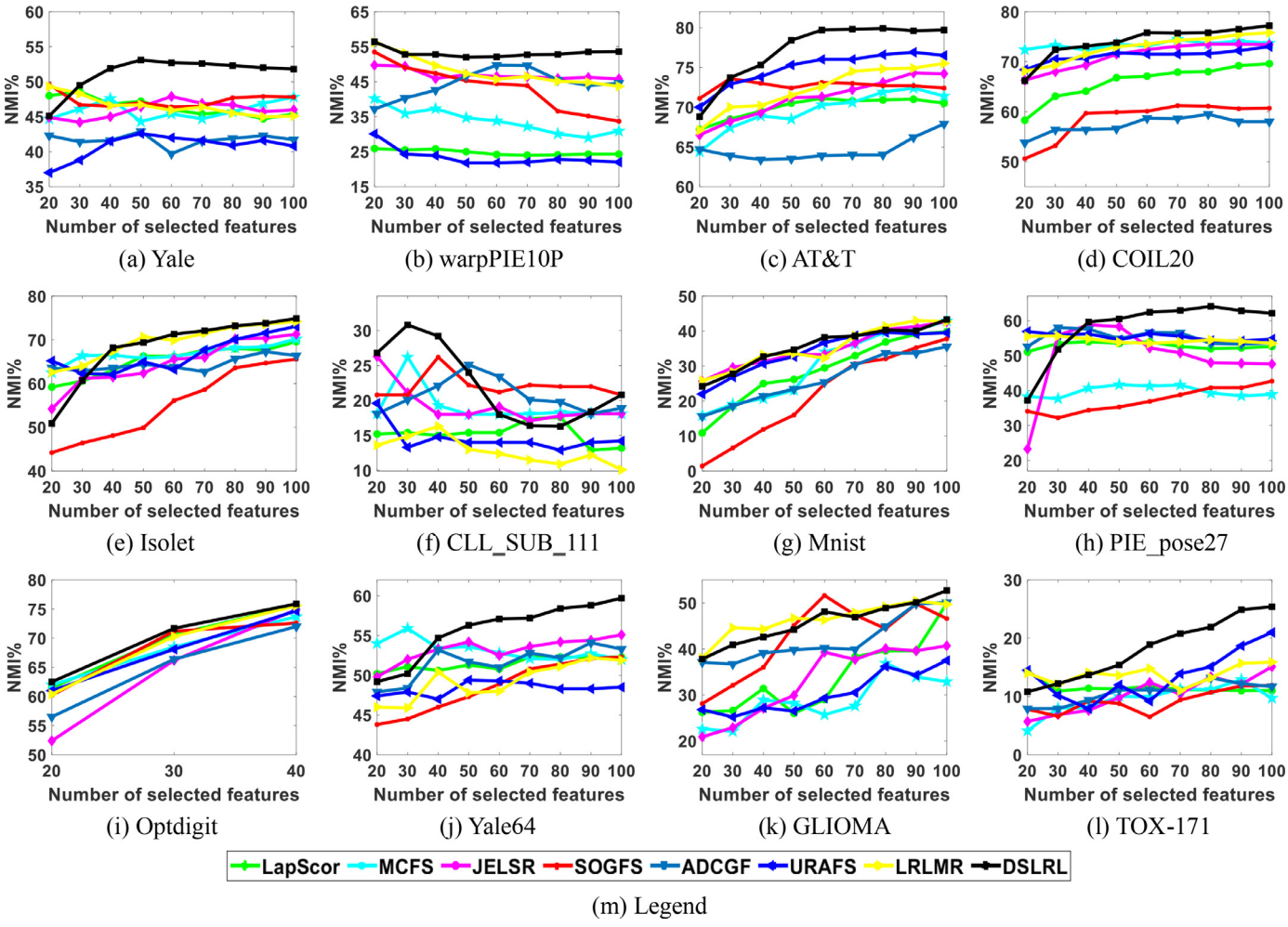

Fig. 2. The NMI of all the algorithms for selecting different numbers of features on the twelve datasets.

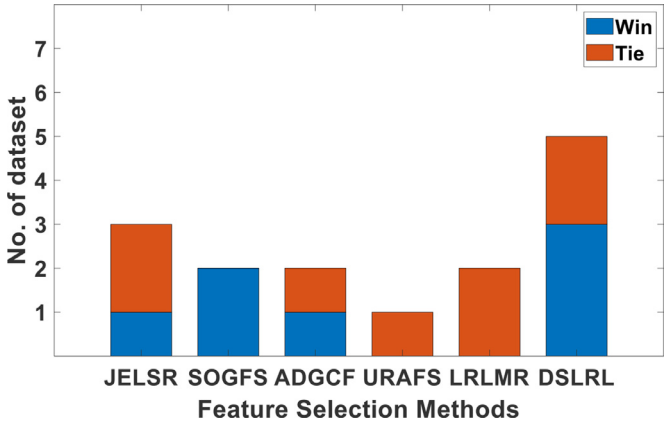

Fig. 3. Performance of feature selection methods based on Score

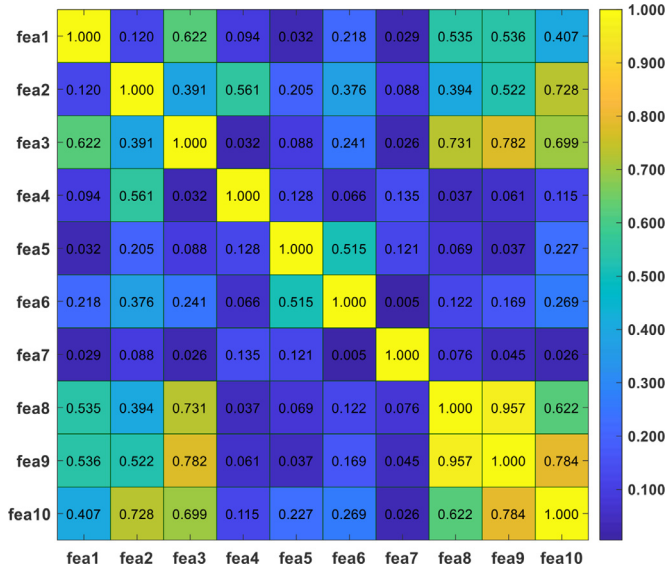

Fig. 4. The heat map of the Pearson correlation coefficient matrix

In order to understand the overall performance of DSLRL, we calculate Score as used in [50,51]. In [51], Sharmin et al. proposed a new metric namely Score and defined it as the weighted average of stability [50] and accuracy. In this paper, we employ equal weight for stability and accuracy. Since there is no iterative process in LapScor and MCFS, we only compare the remaining methods based on Score. Fig. 3 highlights the performance of the remaining feature selection methods based on Score. In Fig. 3, the abscissa represents the feature selection method, and the ordinate indicates the number of dataset.

Fig. 3 compares the performance of feature selection methods based on Score. The “Win” indicates the number of datasets for

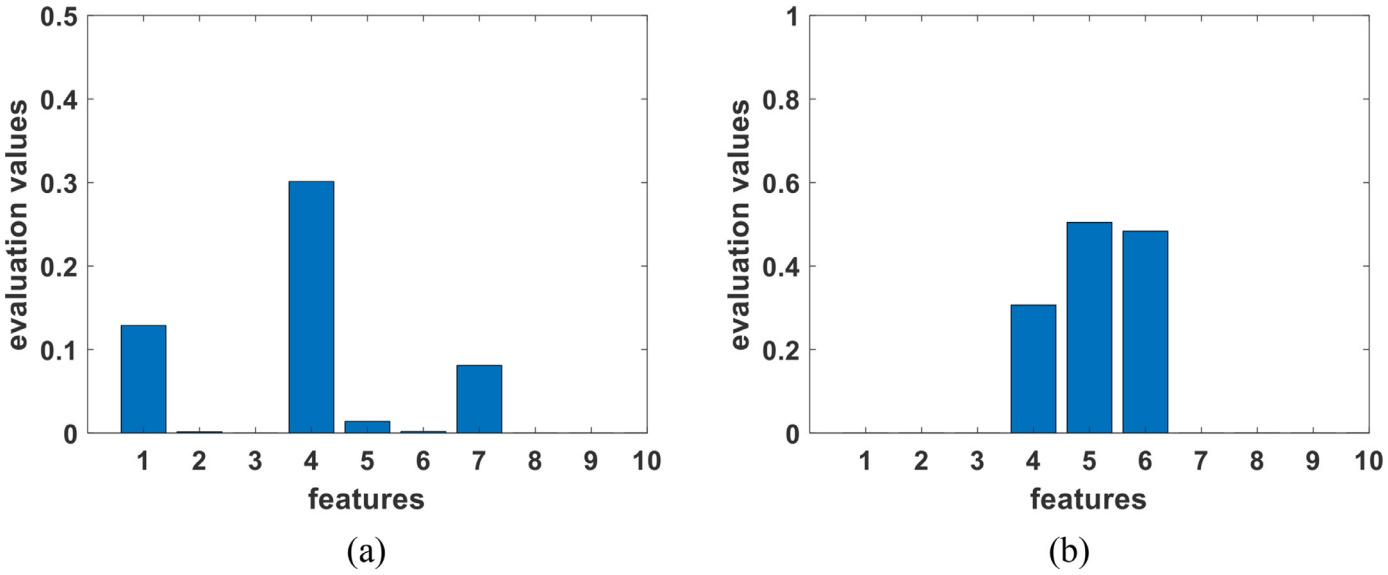

Fig. 5. Feature evaluation value comparison: (a) DSLRL (b) DSLRL without feature latent representation learning

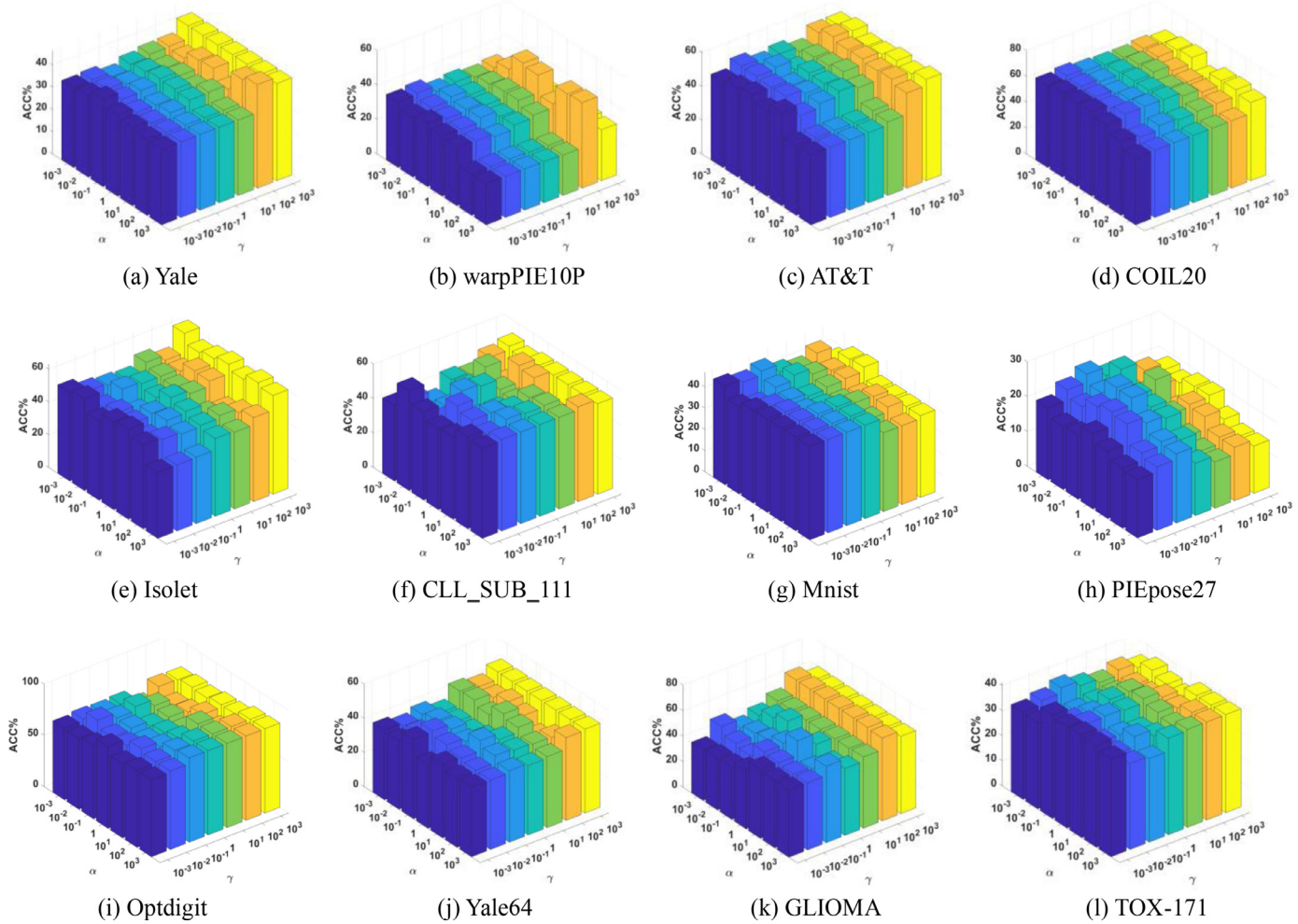

Fig. 6. The ACC of DSLRL on the twelve datasets under values ofαandγ(β= 1 and $\lambda = 1$)

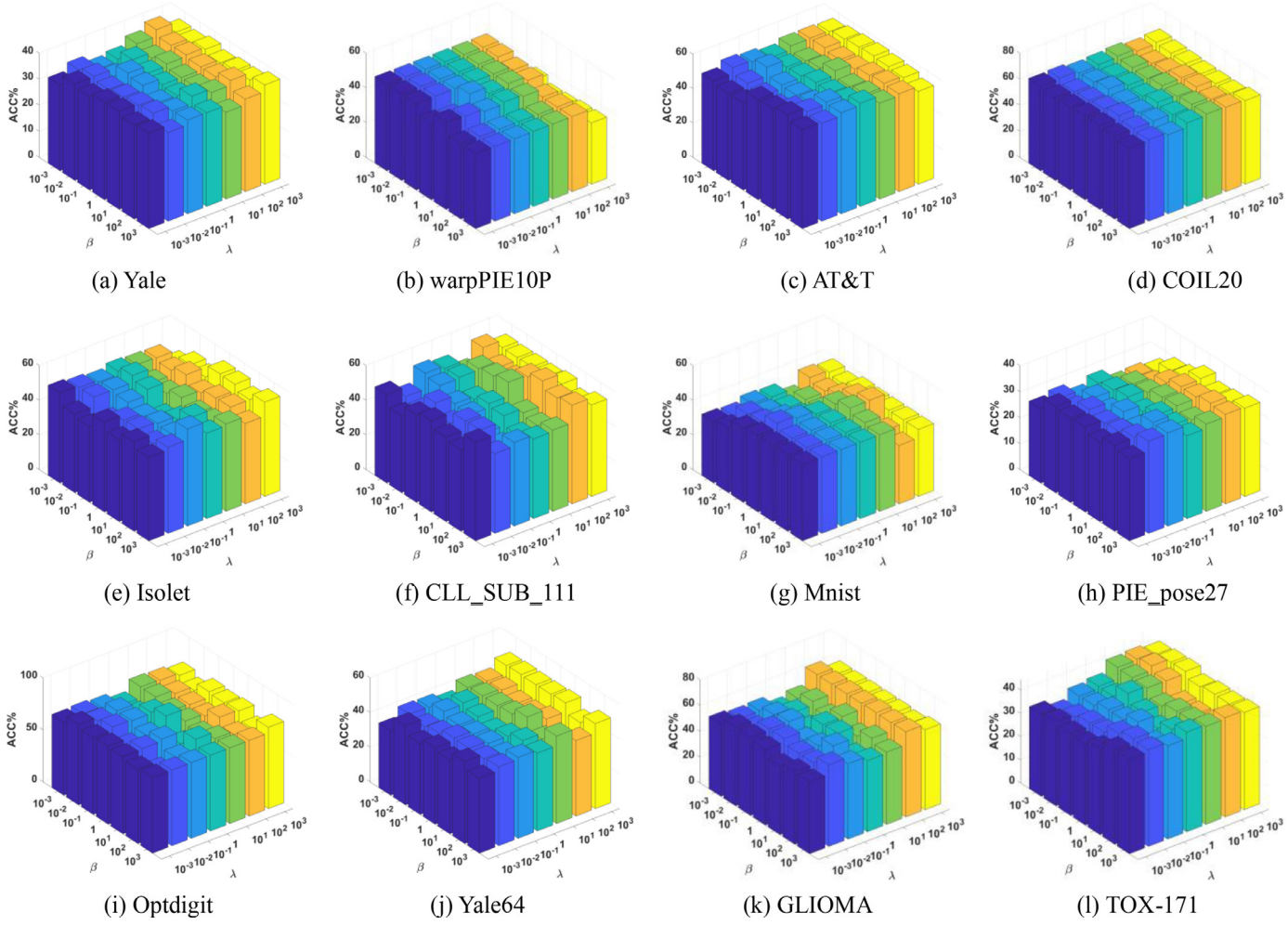

Fig. 7. The ACC of DSLRL on twelve datasets under values of β and $\lambda(\alpha=1$ and $\gamma=1)$

Table 7
The ACC of LRLMR and DSLRL on three datasets with different variance noises (ACC±STD%)

Variance	1		10		20	
Dataset	LRLMR	DSLRL	LRLMR	DSLRL	LRLMR	DSLRL
Yale	38.88±3.06	44.91±3.05	38.27±2.71	42.97±2.10	38.08±2.52	40.85±3.13
AT&T	56.66±2.10	62.02±1.88	5584±1.44	62.41±2.87	56.35±1.98	59.85±1.81
PIE_pose27	29.79±1.18	37.24±0.80	29.69±1.00	37.48±1.17	29.79±1.13	36.50±1.19

which a method performs best compared to other methods. The "Tie" means the number of datasets for which a method does not completely win, but is one of the best performing methods [51]. It can be seen that among all the feature selection methods in Fig. 3, the value of "Win" of DSLRL is the largest. At the same time, considering the sum of "Win" and "Tie", the value of DSLRL is also the largest. Finally, it can be concluded that compared with other methods, DSLRL performs comparatively better based on Score, which shows that the overall performance of DSLRL is good.

4.5. Noise test and low redundancy test

In order to verify that DSLRL can reduce the negative impact of noise and redundant information, we designed some small tests, such as noise test and redundancy test. In the noise test, Yale, AT&T and PIE_pose27 datasets are used. Gaussian noises with variances of 1, 10, and 20 are added into these datasets respectively, and nine datasets with noise are obtained. The clustering results of LRLMR and DSLRL on nine datasets are recorded in Tables 7 and 8, respectively.

It can be seen from the Tables 7 and 8 that the ACC and NMI of DSLRL are higher than LRLMR, indicating the effectiveness of DSLRL. Meanwhile, the better clustering results of DSLRL on the noise-added datasets also prove that the algorithm is robust to noise.

In the low redundancy test, we choose the page-blocks dataset as the test dataset, which contains 5473 samples and 10 features. The Pearson correlation coefficient is used to evaluate the correlation between features, and the heat map of the correlation coefficient matrix is shown in the Fig. 4.

DSLRL performs feature selection on the page-blocks dataset, and the evaluation values of all the features are shown in the Fig. 5(a). As a comparison, we remove the latent representation learning of the feature space and re-select the features, and the obtained feature evaluation values are shown in the Fig. 5(b).

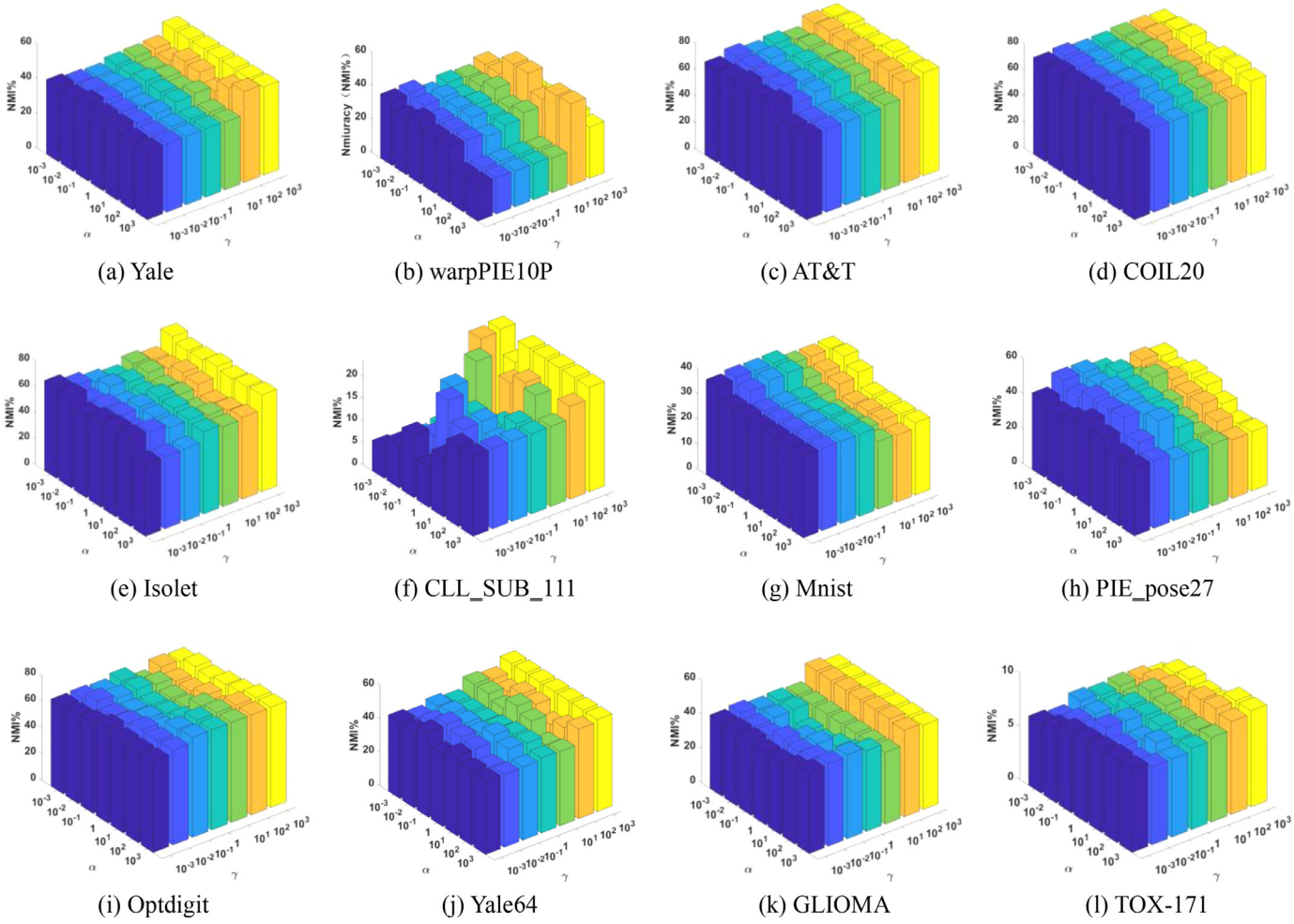

Fig. 8. The NMI of DSLRL on twelve datasets under values of α and $\gamma(\beta = 1$ and $\lambda = 1)$

Table 8
The NMI of LRLMR and DSLRL on three datasets with different variance noises (NMI±STD%)

Variance	1		10		20	
Dataset	LRLMR	DSLRL	LRLMR	DSLRL	LRLMR	DSLRL
Yale	46.27±1.96	51.40±2.12	45.99±2.31	49.09±2.58	45.25±2.25	47.38±2.80
AT&T	74.76±1.29	78.85±1.22	73.77±1.02	79.08±1.39	74.55±1.30	76.61±1.03
PIE_pose27	53.92±0.70	64.05±0.50	53.23±0.65	63.33±0.68	53.77±0.69	62.97±0.56

We can see from the Fig. 5(a) that DSLRL considers features 4, 1 and 7 to be the most representative, and after removing the latent representation learning of the feature space, the selected three features are 5, 6 and 4 from the Fig. 5(b). In Fig. 4, the correlation coefficients of features selected by the former are 0.094, 0.029 and 0.135 with the average value 0.086, while the latter are 0.128, 0.066 and 0.515 with the average value 0.236. The correlation of the three features selected by DSLRL is lower than the latter. It can be concluded that the feature selected by DSLRL without feature latent representation learning is high-redundant, while the features selected by DSLRL have low redundancy.

Table 9
Computational complexity analysis

Algorithms	Computational complexity
LapScor	$O(dn^2)$
MCFS	$O(dn^2)$
JELSR	$O(dn^2+t(n^3+mdn))$
SOGFS	$O(t(d^3+mn^2))$
$ADGCF_{FS}$	$O(n^2d+nd^2+t(n^2m))$
URAFS	$O(t(d^3+dn^2))$
LRLMR	$O(dn^2+t(d^3))$
DSLRL	$O(dn^2+d^2n+t(d^2n))$

4.6. Computational complexity analysis

The computational complexity of the seven comparison algorithms and the proposed algorithm has been shown in Table 9, where n is the number of samples, d is the number of features, m represents the dimension of the low-dimensional space, l represents the number of the selected features, and t is the number of iterations.

Next, the computational complexity of DSLRL will be analyzed. According to the procedure of DSLRL, the running time is mainly used to construct the affinity matrices and iteratively optimize $\mathbf{W}$ and $\mathbf{V}$. The computational complexity of constructing the affinity

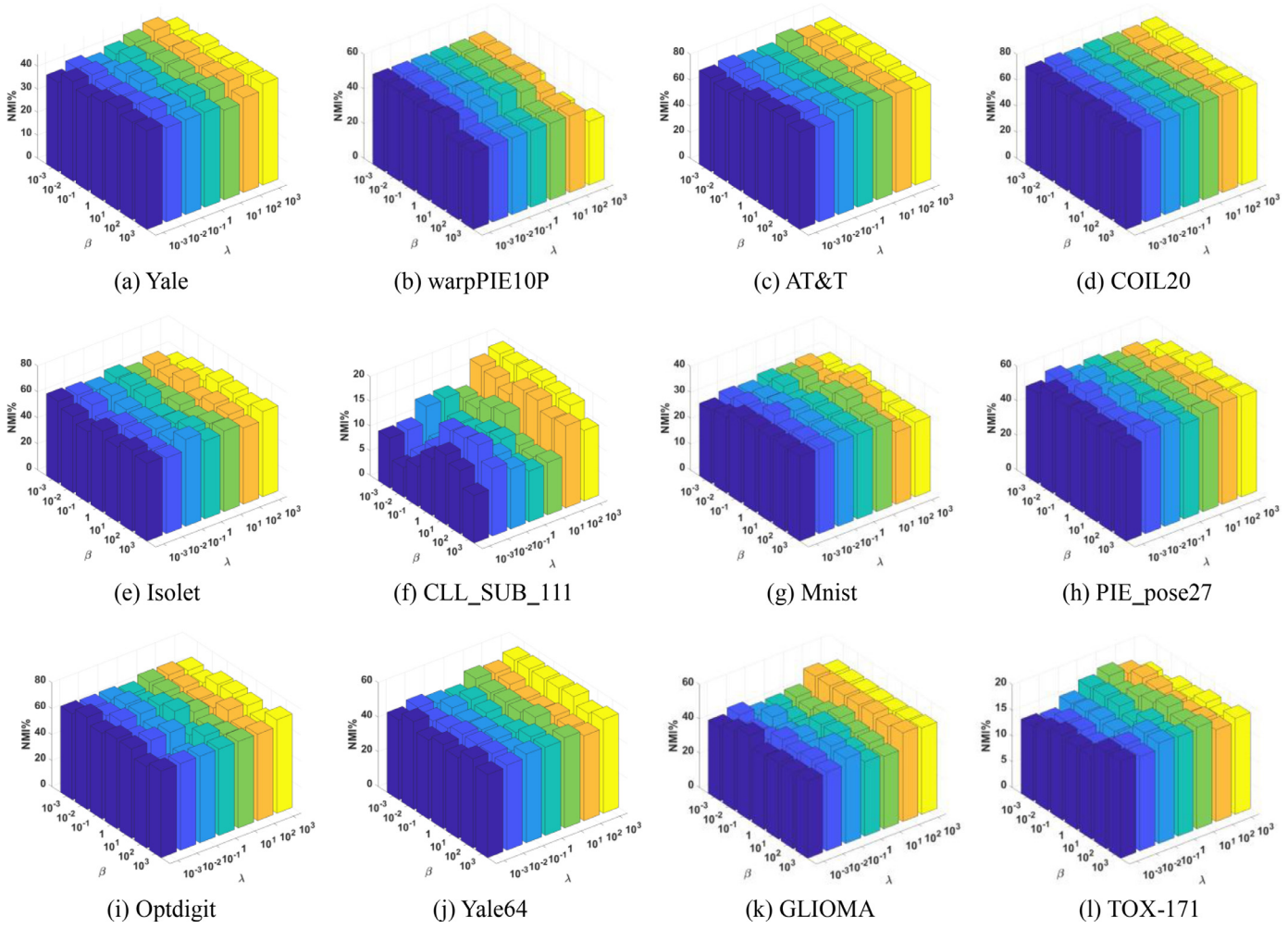

Fig. 9. The NMI of DSLRL on the twelve datasets under values of β and λ(α=1 and γ=1)

Algorithm 1
The procedure of DSLRL.

Input: Data matrix $\mathbf{X} \in \mathbb{R}^{n\times d}$; Maximum iteration number *Niter*; Balance parameters $\alpha, \beta, \gamma, \lambda$; Selected feature number l;
Step 1. Construct the affinity matrices $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{d\times d}$;
Step 2. Initialize *iter*=0, $\mathbf{H}$= **I,** $\mathbf{W} = rand(d, c)$, $\mathbf{V} = rand(n, c)$;
Step 3. Update $\mathbf{W}$ and $\mathbf{V}$ according to the iterative updating formulas (18) and (21), until the convergence conditions are satisfied;
Step 4. Calculate the weight of all the features according to $||\mathbf{W}_i||_2$, and sort them in descending order, then select the top l ranked features as a new data matrix $\mathbf{X}_{new}$.
Output: Index of selected features *index*; New data matrix $\mathbf{X}_{new}$.

matrices $\mathbf{A}$ and $\mathbf{B}$ is $O(dn^2)$ and $O(d^2n)$, respectively. Next the computational complexity of each iteration is $O(d^2n)$. Therefore, the total computational complexity of DSLRL is $O(dn^2+d^2n+t(d^2n))$. From Table 7, the computational complexity relationship between DSLRL and others is determined by the relationship between d and n.

4.7. Parameters sensitivity analysis

The parameters of DSLRL include balance parameters $\alpha, \beta, \gamma, \lambda$ and Gaussian kernel bandwidth parameters σ_1, σ_2. In this paper, we only discuss the sensitivity of the balance parameters $\alpha, \beta, \gamma, \lambda$. Because $0 < A_{ij} \leq 1$ and $0 < \mathbf{B}_{ij} \leq 1$, both σ_1 and σ_2 will have a fixed value on each dataset. Fixing other parameters, we record the changes of ACC and NMI as balance parameters vary. The search ranges of parameters α, β, γ and λ are both $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10^{1}, 10^{2}, 10^{3}\}$. Figs. 6 and 8 illustrate the changes of ACC and NMI on twelve datasets under different values of α and γ when β= 1 and λ = 1. Figs. 7 and 9 illustrate the changes of ACC and NMI on twelve datasets under different values of β and λ when α=1 and γ=1.

Fig. 6 shows that when the parameters α and γ vary, the ACC of DSLRL does not change obviously on most datasets. In particular, the performance of DSLRL is very stable on the COIL20 and Optdigit dataset. On the Yale, AT&T, warpPIE10P, Isolet, Minist, Yale64, GLIOMA and TOX-171 datasets, the ACC shows a steady upward trend, and the range of fluctuations relatively small. On the remaining datasets, the ACC occasionally fluctuates, but in general, most of the ACC values are in a stable state. It can be concluded from Fig. 6 that the ACC of DSLRL is relatively stable with the change of parameters α and γ.

Fig. 7 displays that when α and γ are fixed, with the changes of β and λ, the trend of ACC is relatively stable on most datasets, especially on the Yale, AT&T and COIL20 datasets. Overall, compared

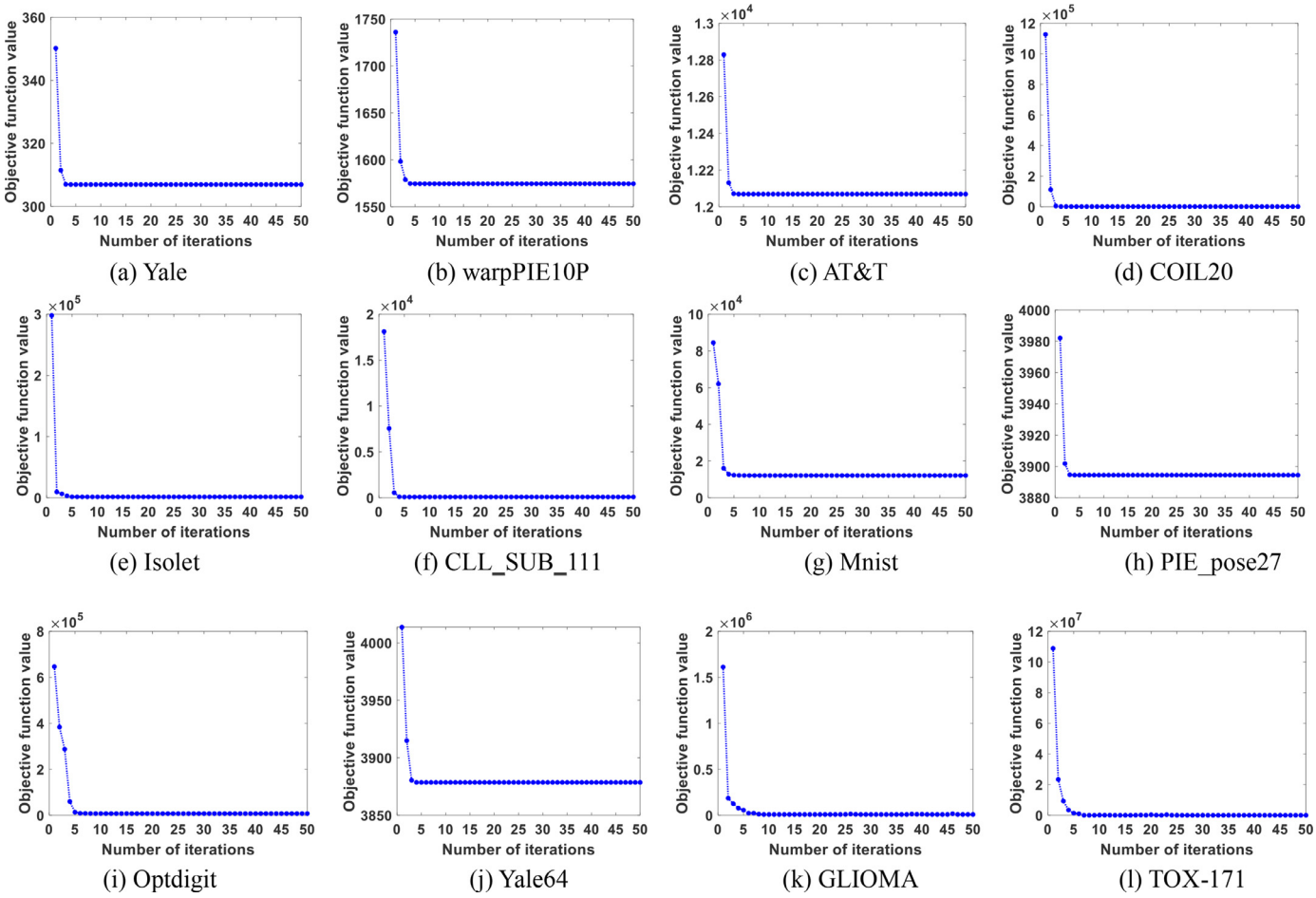

Fig. 10. The convergence curves of DSLRL on twelve datasets (α=1000, β=0.001 and γ= 0.001)

to Fig. 6, we find that the ACC of DSLRL is less sensitive to parameters β and λ.

In Fig. 8, the NMI rises slightly, and the performance is stable on the Yale, AT&T, COIL20, and Optdigit datasets. While on the warpPIE10P, Isolet, Mnist, PIEpose27, Yale64, GLIOMA and TOX-171 datasets, the NMI slowly rises in steps and it is a little bit sensitive to the parameters α and γ.

Fig. 9 demonstrates that the value of NMI given by DSLRL when parameters α=1 and γ=1. From these pictures, we can see that when λ is large, the NMI of DSLRL rapidly increases to a peak, and when $10^{-1} \leq \beta \leq 10^{1}$, the NMI shows a better value on CLL_SUB_111 dataset. For other datasets, the NMI of DSLRL is relatively stable.

Combined with Figs. 6-9, the performance of ACC and NMI on the CLL_SUB_111 dataset is not as stable as their performance on other datasets. This is due to the properties of CLL_SUB_111 dataset. The large number of feature and high similarity among features make it is more difficult for feature selection. It can be seen from Figs. 6-9 that the results of DSLRL are gradually increasing. Although DSLRL performs better than others, the results are still not very stable due to the characteristics of CLL_SUB_111 dataset. Therefore, it is our future research work to improve the stability of DSLRL for such datasets.

4.8. Convergence study

The convergence analysis of the proposed algorithm has been given in the previous section. Here, we show the convergence curves of DSLRL on different datasets to intuitively illustrate the convergence properties of the designed algorithm.

The convergence curves of DSLRL for different iterations on twelve datasets are shown in Fig. 10.

The vertical axis represents the value of the objective function, and the horizontal axis is the number of iterations. It can be seen that as the number of iterations increases, the value of the objective function decreases rapidly and converges on each dataset. Fig. 10 verifies the convergence of DSLRL.

5. Conclusions

This paper proposes an unsupervised feature selection algorithm called DSLRL, which exploits the inherent association information in data space and feature space to improve the effect of feature selection. The proposed algorithm combines the advantages of latent representation learning and sparse learning, hence the performance of feature selection is improved. We propose latent representation learning based on dual space, which characterizes the intrinsic structure of data space and feature space, respectively. We make the latent representation matrix of data space close to the ideal label matrix, and unify the sparse transformation matrix with the latent representation matrix of feature space, so that the internal information of dual space is fully utilized to optimize the feature selection. Then, the $l_{2,1}$-*norm* constraint is used to ensure the row sparseness of the matrix. In the optimization process, we employ the alternating method to obtain the updating rules of **W** and **V**. Finally, this paper compares DSLRL with seven comparison algorithms on several datasets. The results of cluster-

ing experiments show that DSLRL has good clustering results on most datasets, while the results of parameter sensitivity experiments verify the robustness of DSLRL. Overall, the experimental results show that DSLRL outperforms the other comparison algorithms.

The disadvantage of DSLRL is that it has many parameters that need to be adjusted, which leads to a large search range. In the future research, we will further explore the method of parameter adaptation to reduce the cost of the algorithm and find the most suitable parameter combination. In addition, we hope to study a novel optimization method that can simultaneously optimize the variables **W** and **V**.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

We would like to express our sincere appreciation to the editors and the anonymous reviewers for their insightful comments, which have greatly helped us in improving the quality of the paper. This work was partially supported by the National Natural Science Foundation of China under Grants 61773304, 61871306, 61772399, 61836009, and U1701267, the Fund for Foreign Scholars in University Research and Teaching Programs (the 111 Project) under Grants No. B07048, and the Program for Cheung Kong Scholars and Innovative Research Team in University under Grant IRT1170.

References

[1] A. Jain, D. Zongker, Feature selection: Evaluation, application, and small sample performance, IEEE Trans. Pattern Anal. Mach. Intell. 19 (2) (1997) 153–158.
[2] F. Nie, H. Huang, X. Cai, C.H. Ding, Efficient and robust feature selection via joint $l_{2,1}$-norms minimization, Adv. Neural Inf. Process. Syst. (2010) 1813–1821.
[3] R. Shang, W. Wang, R. Stolkin, L. Jiao, Non-negative spectral learning and sparse regression-based dual-graph regularized feature selection, IEEE Trans. Cybern. 48 (2) (2017) 793–806.
[4] H. Yan, J. Yang, Sparse discriminative feature selection, Pattern Recognit. 48 (5) (2015) 1827–1835.
[5] R. Shang, Y. Meng, F. Shang, L. Jiao, Local discriminative based sparse subspace learning for feature selection, Pattern Recognit. 92 (2019) 219–230.
[6] M. Sikonja, I. Kononenko, Theoretical and empirical analysis of relief and relief, Mach. Learn. 53 (1-2) (2003) 23–69.
[7] Z. Xu, I. King, M. Lyu, R. Jin, Discriminative semi-supervised feature selection via manifold regularization, IEEE Trans. Neural Netw. 21 (7) (2010) 1033–1047.
[8] Y. Wang, J. Wang, H. Liao, H. Chen, An efficient semi-supervised representatives feature selection algorithm based on information theory, Pattern Recognit. 61 (2017) 511–523.
[9] J. Dy, C. Brodley, Feature selection for unsupervised learning, J. Mach Learn. Res. 5 (2004) 845–889.
[10] M. Banerjee, N. Pal, Unsupervised Feature Selection with Controlled Redundancy (UFeSCoR), IEEE Trans. Knowl. Data Eng. 27 (12) (2015) 3390–3403.
[11] Z. Zhao, H. Liu, Spectral feature selection for supervised and unsupervised learning, Proc. Int. Conf. Mach. Learn. (2007) 1151–1157.
[12] X. He, D. Cai, P. Niyogi, Laplacian score for feature selection, Adv. Neural Inf. Process. Syst. (2005) 507–514.
[13] P. Mitra, C. Murthy, S. Pal, Unsupervised feature selection using feature similarity, IEEE Trans. Pattern Anal. Mach. Intell. 24 (3) (2004) 301–302.
[14] M. Law, M. Figueiredo, A. Jain, Simultaneous feature selection and clustering using mixture models, IEEE Trans. Pattern Anal. Mach. Intell. 26 (9) (2004) 1154–1165.
[15] J. Dy, C. Brodley, Feature Selection for Unsupervised Learning, J. Mach. Learn. Res. 5 (2004) 845–889.
[16] C. Hou, F. Nie, D. Yi, Y. Wu, Feature selection via joint embedding learning and sparse regression, IJCAI Proc.-Int. Joint Conf. Artif. Intell. (2011) 1324–1329.
[17] X. Liu, L. Wang, J. Zhang, J. Yin, H. Liu, Global and local structure preservation for feature selection, IEEE Trans. Neural Netw. Learn. Syst. 25 (6) (2013) 1083–1095.
[18] H. Yuan, J. Li, L. Lai, Y. Tang, Joint sparse matrix regression and nonnegative spectral analysis for two-dimensional unsupervised feature selection, Pattern Recognit. 89 (2019) 119–133.
[19] R. Zhang, F. Nie, X. Li, X. Wei, Feature selection with multi-view data: A survey, Inf. Fusion 50 (2019) 158–167.
[20] D. Cai, C. Zhang, X. He, Unsupervised feature selection for multi-cluster data, Proc. 16th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min. (2010) 333–342.
[21] Z. Zhao, L. Wang, H. Liu, Efficient spectral feature selection with minimum redundancy, Proc. Twenty-Fourth AAAI Conf. Artif. Intell. (2010) 673–678.
[22] C. Hou, F. Nie, X. Li, D. Yi, Y. Wu, Joint embedding learning and sparse regression: a framework for unsupervised feature selection, IEEE Trans. Cybern. 44 (6) (2014) 2168–2267.
[23] F. Nie, W. Zhu, X. Li, Unsupervised feature selection with structured graph optimization, Proc. Thirtieth AAAI Conf. Artif. Intell. (2016) 1302–1308.
[24] X. Li, H. Zhang, R. Zhang, Y. Liu, F. Nie, Generalized uncorrelated regression with adaptive graph for unsupervised feature selection, IEEE Trans. Neural Netw. Learn. Syst. 30 (5) (2019) 1587–1595.
[25] R. Shang, J. Chang, L. Jiao, Y. Xue, Unsupervised feature selection based on self-representation sparse regression and local similarity preserving, Int. J. Mach. Learn. Cybern. 10 (4) (2019) 757–770.
[26] W. He, X. Cheng, R. Hu, Y. Zhu, G. Wen, Feature self-representation based hypergraph unsupervised feature selection via low-rank representation, Neurocomputing 253 (2017) 127–134.
[27] C. Tang, X. Liu, M. Li, P. Wang, J. Chen, L. Wang, W. Li, Robust Unsupervised Feature selection via dual self-representation and manifold regularization, Knowl.-Based Syst. 145 (2018) 109–120.
[28] Y. Fan, J. Dai, Q. Zhang, Latent Space Embedding for Unsupervised Feature Selection via Joint Dictionary Learning, in: 2019 International Joint Conference on Neural Networks (IJCNN), 2019, pp. 1–8.
[29] C. Tang, M. Bian, X. Liu, M. Li, H. Zhou, P. Wang, H. Yin, Unsupervised feature selection via latent representation learning and manifold regularization, Neural Netw. 117 (2019) 163–178.
[30] Z. He, S. Xie, R. Zdunek, G. Zhou, A. Cichocki, Symmetric nonnegative matrix factorization: algorithms and applications to probabilistic clustering, IEEE Trans. Neural Netw. 22 (12) (2011) 2117–2131.
[31] P. Luo, J. Peng, Z. Guan, J. Fan, Dual-regularized multi-view non-negative matrix factorization, Neurocomputing 294 (2018) 1–11.
[32] J. Ye, Z. Jin, Dual-graph regularized concept factorization for clustering, Neurocomputing 138 (2014) 120–130.
[33] J. Ye, Z. Jin, Feature selection for adaptive dual-graph regularized concept factorization for data representation, Neural Process. Lett. 45 (2) (2017) 667–668.
[34] X Du, F Nie, W Wang, Y Yang, X Zhou, Exploiting combination effect for unsupervised feature selection by $l_{2,0}$ norm, IEEE Trans. Neural Netw. Learn. Syst. 30 (1) (2019) 1–14.
[35] Y. Jacob, L. Denoyer, P. Gallinari, Learning latent representations of nodes for classifying in heterogeneous social networks, Proc. 7th ACM Int. Conf. Web Search Data Min. (2014) 373–382.
[36] L. Tang, H. Liu, Relational learning via latent social dimensions, Proc. 15th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min. (2009) 817–826.
[37] J. Li, X. Hu, L. Wu, Robust unsupervised feature selection on networked data, in: Proceedings of the 2016 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics, 2016, pp. 387–395.
[38] D. Ding, X. Yang, F. Xia, T. Ma, H. Liu, C Tang, Unsupervised feature selection via adaptive hypergraph regularized latent representation learning, Neurocomputing 378 (2020) 79–97.
[39] J. Cui, Q. Zhu, D. Wang, Z. Li, Learning robust latent representation for discriminative regression, Pattern Recognit. Lett. 117 (2019) 193–200.
[40] D. Kuang, C. Ding, H. Park, Symmetric nonnegative matrix factorization for graph clustering, Proc. 2012 SIAM Int. Conf. Data Min. (2012) 106–117.
[41] F. Shang, L. Jiao, F. Wang, Graph dual regularization non-negative matrix factorization for co-clustering, Pattern Recognit. 45 (6) (2012) 2237–2250.
[42] W. Xu, Y. Gong, Document clustering by concept factorization, in: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, 2004, pp. 202–209.
[43] D. Lee, Seung H, Algorithms for non-negative matrix factorization, Adv. Neural Inf. Process. Syst. (2001) 556–562.
[44] A. Rakhlin, A. Caponnetto, Stability of k-means clustering, Adv. Neural Inf. Process. Syst. (2007) 1121–1128.
[45] S. Wang, J. Chen, W. Guo, G. Liu, Structured learning for unsupervised feature selection with high-order matrix factorization, Expert Syst. Appl. 140 (2020) 112878.
[46] R. Shang, Z. Zhang, L. Jiao, W. Wang, S. Yang, Global discriminative-based nonnegative spectral clustering, Pattern Recognit. 55 (2016) 172–182.
[47] N. Zhou, Y. Xu, H. Cheng, Z. Yuan, Maximum correntropy criterion-based sparse subspace learning for unsupervised feature selection, IEEE Trans. Circ. Syst. Video Technol. 29 (2) (2017) 404–417.
[48] A. Strehl, J. Ghosh, Cluster ensembles-a knowledge reuse framework for combining multiple partitions, J. Mach Learn. Res. 3 (2002) 583–617.
[49] C. Papadimitriou, K. Steiglitz, Combinatorial optimization: Algorithms and Complexity, Dover, New York, NY, USA, 1998.
[50] G. Brown, A. Pocock, M. Zhao, M. Lujan, Conditional likelihood maximisation: a unifying framework for information theoretic feature selection, J. Mach. Learn. Res. 13 (1) (2012) 27–66.
[51] S. Sharmin, M. Shoyaib, A. Ali, M. Khan, O. Chae, Simultaneous feature selection and discretization based on mutual information, Pattern Recognit. 91 (2019) 162–174.

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

Lujuan Wang received the B.E. degree in School of Computer Science and Technology, Tiangong University, Tianjin, China in 2018. She is now pursuing the M.S. degree in School of Artificial Intelligence from Xidian University, Xi'an, China. Her current research interests include data mining and machine learning.

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

Yangyang Li (M'08-SM'18) received the B.S. and M.S. degrees in computer science and technology and the Ph.D. degree in pattern recognition and intelligent system from Xidian University, Xi'an, China, in 2001, 2004, and 2007, respectively. She is currently a Professor with the School of Artificial Intelligence, Xidian University. Her research interests include quantum-inspired evolutionary computation, artificial immune systems, and deep learning.