Knowledge-Based Systems 272 (2023) 110601

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Structural regularization based discriminative multi-view unsupervised feature selection

Shixuan Zhou [a], Peng Song [a,*], Yanwei Yu [b], Wenming Zheng [c]

[a] School of Computer and Control Engineering, Yantai University, Yantai 264005, China
[b] College of Computer Science and Technology, Ocean University of China, Qingdao 266400, China
[c] Key Laboratory of Child Development and Learning Science (Ministry of Education), Southeast University, Nanjing 210096, China

ARTICLE INFO

Article history:
Received 28 February 2023
Received in revised form 25 April 2023
Accepted 26 April 2023
Available online 29 April 2023

Keywords:
Multi-view learning
Graph learning
Latent representation
Feature selection

ABSTRACT

Multi-view unsupervised feature selection (MUFS) has recently aroused considerable attention, which can select the compact representative feature subset from original multi-view data. Despite the promising preliminary performance, most previous MUFS methods fail to explore the discriminative ability of multi-view data. In addition, they usually utilize spectral analysis to maintain the geometrical structure, which will inevitably increase the difficulty of parameter selection. To address these issues, we present a novel MUFS method, named structural regularization based discriminative multi-view unsupervised feature selection (SDFS). Specifically, we calculate the similarity matrix of sample space from different views and automatically weight each view-specific graph to learn a consensus similarity graph, in which these two types of graphs can promote each other. Further, we treat the learned latent representation as the cluster indicator, and employ a graph regularization without introducing additional parameters to maintain the geometrical structure of data. Besides, a simple yet efficient iterative updating algorithm with theoretical convergence property is developed. Extensive experiments on several benchmark datasets verify that the designed model is superior to several state-of-the-art MUFS models.

1. Introduction

Nowadays multi-view data are ubiquitous in real-world applications. For instance, images can be described by different features, e.g., local binary pattern (LBP), global information (GIST), census transform histogram (CENT) and scale-invariant feature transform (SIFT). Videos can be described by different features, e.g., linear predictive coefficient (LPC) and discrete wavelet transform (DWT). Although different views can describe an identical object, each view has unique heterogeneous features. Traditional single-view learning methods often integrate all features into a single view to handle the multi-view data, which might neglect the correlation between different views and will affect the performance to a certain extent. To comprehensively explore the abundant intrinsic information of an object from multi-view data, many multi-view learning methods build the models directly from multi-view data, including active learning [1], transfer learning [2], multi-view representation learning [3], multi-view clustering [4], and multi-view feature selection [5]. Among them, multi-view feature selection is an efficient way to reduce the dimension of high-dimensional multi-view data, which has received considerable attention recently. In addition, since data labeling is labor-intensive and time-consuming, multi-view data are often labeled in practical applications. Thus, the multi-view unsupervised feature selection (MUFS) method is a fundamental and challenging task, which has been employed in various applications, e.g., visual concept recognition [6], activity recognition [7], and human motion retrieval [8].

Recently, numerous MUFS methods have been presented [6,9–12]. The key problem of MUFS is how to model the multi-view information to guide the process of feature selection. To solve this issue, the clustering structure and the geometrical structure of the original data are often considered. Specifically, Feng et al. [6] consider the graph Laplacian [9] of each view, and then linearly combine these graphs to maintain the local geometrical structure. Liu et al. [10] explore the clustering structure by developing a robust multi-view clustering strategy. Tang et al. [11] maintain the local geometrical structure by cross-view similarity graph learning. Fang et al. [12] integrate the learning of the clustering structure and the preservation of the geometrical structure, which can simultaneously explore the clustering structure and the geometrical structure.

* Corresponding author.
E-mail addresses: a-ngie@foxmail.com (S. Zhou), pengsong@ytu.edu.cn (P. Song), yuyanwei@ouc.edu.cn (Y. Yu), wenming_zheng@seu.edu.cn (W. Zheng).

https://doi.org/10.1016/j.knosys.2023.110601
0950-7051/

Although these above-mentioned MUFS methods achieve promising performance, there still exist three key problems. (a) Although existing methods fully utilize the information of clustering structure or geometrical structure, they rarely integrate them into a joint framework. (b) Existing methods usually construct view-specific similarity graphs or consensus similarity graph statically, which might lead to a lower quality of the generated graph. (c) Existing pseudo-label learning methods cannot fully consider the clustering structure information of the original multi-view data.

To address these issues, we develop a novel structural regularization based discriminative multi-view unsupervised feature selection (SDFS) method, which can explore the intrinsic information of clustering structure while maintaining the geometrical structure of data. Specifically, the samples from the same source or different sources usually correlate to each other due to the influence of external conditions, it is crucial to explore the inherent attributes of the original data through the link information. Thus, we measure the intrinsic relations between different samples by using the adjacency matrix, and then learn the latent representation matrix of the adjacency matrix. Since the learned latent representation matrix contains the clustering structure of data samples, it is treated as prior knowledge to guide the process of feature selection. Further, we employ an automatic weighting strategy to jointly learn the view-specific similarity graph and the consensus similarity graph. Moreover, we design a novel parameter-less graph regularization strategy, which can maintain the geometrical structure without introducing additional parameters. Finally, we impose the $\ell_{2,1}$-norm to constrain the row sparsity of feature selection matrices.

The main contributions of this work are summarized as follows:

- The latent representation learning is conducted in data space, in which the learned low-dimensional latent representation matrix is regarded as the cluster indicator matrix to provide prior knowledge for feature selection tasks.
- An automatic weighting strategy is developed to jointly learn the view-specific similarity graph and the consensus similarity graph, in which these two learning processes can promote each other.
- A parameter-less graph regularization strategy is designed to maintain the geometrical structure of sample space without introducing additional parameters.
- An efficient scheme is proffered to optimize the proposed method, and extensive experimental results verify the superiority of the proposed SDFS model.

The remainder of this paper is organized as follows. Section 2 briefly introduces the existing related work. Section 3 describes the details of the proposed SDFS model and the alternating updating scheme. Section 4 provides extensive experiments and corresponding analysis. Finally, the conclusion is included in Section 5.

2. Related work

2.1. Single-view feature selection

Unsupervised feature selection (UFS) is classified into three main types, i.e., filter methods [13–15], wrapper methods [16–18], and embedded methods [19,20]. The filter methods utilize an evaluation index to get the most representative feature subsets from the original features. The typical filter methods include PCA score [13], Laplacian score [14], and spectral feature selection (SPEC) [15]. The wrapper methods evaluate the feature subset according to the accuracy of clustering or classification, and usually obtain better performance in comparison with filter methods [18]. However, the time cost of wrapper methods is very expensive for large-scale data [21]. The embedded methods integrate the model construction and feature selection, so that these two procedures can be simultaneously optimized.

Due to the advantages of lower computing cost and higher performance, embedded methods gain more attention than the other two categories. For instance, in [19], Cai et al. integrate the spectral analysis into the UFS framework to improve performance. In [20], Yang et al. apply the discriminative analysis in UFS tasks to select the most distinguishing features. In [22], Sheng et al. improve the performance of UFS by maintaining the local geometrical structure of both feature and sample space. In [23], Wang et al. develop an unsupervised soft-label feature selection (USFS) method, in which the learned soft-label matrix can enhance the discriminative ability of the algorithm while effectively alleviating the loss of information. In [24], Li et al. propose a self-paced learning and low-redundant regularization (SPLR) method for UFS, which utilizes self-paced learning to remove the outliers in original features. In [25], Miao et al. present a graph regularized local linear embedding (GLLE) method for UFS, which simultaneously employs manifold regularization and locally linear embedding to preserve the local invariance of feature subspace. In [26], we present a soft-label guided non-negative matrix factorization (SLNMF) based UFS method, which utilizes soft-label regression to guide the process of non-negative matrix factorization for obtaining suitable low-dimensional representation of data. In [27], You et al. present a neural networks embedded self-expression (NNSE) based UFS method, which utilizes neural networks to explore the nonlinear mapping correlation between original data and pseudo-labels. Note that the above methods are proposed for single-view learning, which cannot directly handle the multi-view data.

2.2. Multi-view feature selection

To cope with the feature selection for multi-view data, various methods have been developed. For instance, in [6], Feng et al. propose a MUFS method for visual concept recognition. In [8], Wang et al. propose a MUFS method for human motion retrieval. In [10], Liu et al. employ a robust multi-view k-means clustering algorithm to learn robust and high-quality pseudo-labels for feature selection, which reduces the computational complexity of the pseudo-label learning in previous MUFS methods. In [28], Lin et al. employ a maximum margin criterion to learn the inter-class and intra-class structure information of each view, which helps to learn the discriminative transformation matrix. In [29], Kennedy et al. propose a mixed sparsity regularized MUFS (MSMFS) method, which imposes mixed group sparsity regularization to alleviate the effects of outliers and different views. In [30], Lin et al. integrate the locally sparse regularization terms with a shared loss to enhance the sparsity of blocks from views and features.

Previous MUFS methods explore the manifold structure of each view with fixed and predefined similarity matrices separately without considering the common structures. To tackle this problem, in [31], Hou et al. propose an adaptive similarity and view weight (ASVW) method, which learns an adaptive common similarity matrix to characterize the manifold structure from different views. In [32], Dong et al. present an adaptive collaborative similarity learning (ACSL) method, which dynamically learns the desirable collaborative similarity structure and the ideal neighbor assignment. In [33], Bai et al. propose a non-negative structured graph learning (NGSL) method, which imposes the rank constraint on the similarity graph to ensure the ideal structure. In [34], Wan et al. present an adaptive similarity embedding (ASE-UMFS) method for MUFS, which unifies data from different views

Table 1
Notations and descriptions.

Notations	Description
$X^{(m)} \in R^{n\times d^{(m)}}$	The original data matrix in the mth view
$F^{(m)} \in R^{n\times c}$	The pseudo-label matrix in the mth view
$W^{(m)} \in R^{d^{(m)}\times r}$	The selection matrix in the mth view
$H^{(m)} \in R^{r\times c}$	The coefficient matrix in the mth view
$A^{(m)} \in R^{n\times n}$	The adjacency matrix in the mth view
$S^{(m)} \in R^{n\times n}$	The view-specific similarity matrix in the mth view
$D^{(m)} \in R^{n\times n}$	The diagonal matrix of $S^{(m)}$
$\overline{S} \in R^{n\times n}$	The consensus similarity matrix
$\overline{D} \in R^{n\times n}$	The diagonal matrix of $\overline{S}$
$\alpha^{(m)}$	The adaptive view weighting in the mth view
n	The number of samples
$d^{(m)}$	The number of features in the mth view
n_v	The number of views
r	The dimension of latent subspace
c	The number of clusters

to an optimal sparse subspace to maintain the global structure and learns a consensus similarity matrix to maintain the local structure. Similar to these methods, in this work, we focus on learning a unified similarity matrix to maintain consistency by a linear weighting fusion.

To fully take into account the diversity and consistency of data, in [35], Tang et al. assume that the cluster indicator matrix of each view should be as same as possible, and propose a consensus learning guided MUFS (CGMV-UFS) method. In [11], Tang et al. integrate all views of data into a consensus pseudo-label space to explore both consensus information and diversity information. In [36], Yuan et al. construct the graph matrix from each view-specific subspace to explore the complementary information, and employ a low-rank tensor regularization term to ensure the consistency of different views. In [12], Fang et al. propose a joint MUFS and graph learning (JMVFG) method, which utilizes a cross-space locality preserving term to bridge the gap between the global manifold in the original and the local manifold in the projected space. Moreover, to solve the problem of higher computational complexity in traditional graph learning, in [37], Shi et al. propose a multi-view feature selection with binary hashing (MVFS-BH) method, which imposes a binary hash constraint in the process of graph learning to obtain binary hash codes as pseudo-labels. The above methods introduce graph regularization to improve the performance, which will inevitably increase the difficulty of parameter regulation. By contrast, we develop a parameter-less graph regularization strategy, which can efficiently alleviate this issue.

3. Proposed method

In this section, we first introduce the main notations. Then, we elaborate on the proposed SDFS, as well as the iterative optimization algorithm and complexity analysis of SDFS.

3.1. Notations

For an arbitrary matrix Y, y_i is the ith instance of Y, and Y^T and $\mathrm{Tr}(Y)$ represent the transpose and trace of Y, respectively. $\mathbf{1} \in R^{n\times 1}$ denotes the column vector whose elements are all 1, and $I_1 \in R^{r\times r}$ and $I_2 \in R^{c\times c}$ denote the identity matrices. $\|Y\|_{2,1} = \sum_{i=1}^{n}\sqrt{\sum_{j=1}^{d} Y_{ij}^2}$ represents the $\ell_{2,1}$-norm of Y. $\|Y\|_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{d} Y_{ij}^2}$ represents the Frobenius norm of Y. For the sake of clarity, Table 1 summarizes the commonly used notations in our work.

3.2. Latent representation learning

Generally, in multi-view data, the latent representations of different instances affect each other and form linked information accordingly, in which the instances with similar latent representations are more likely to be concatenated than those with different latent representations. Here, we utilize the Gaussian function to explore the relationship between instances, which is formulated as follows:

$$A_{ij}^{(m)} = \begin{cases} \exp\left(\frac{\|x_i^{(m)} - x_j^{(m)}\|_2^2}{-2\sigma^2}\right), & x_i^{(m)} \in \mathcal{N}_k(x_j^{(m)}) or\ x_j^{(m)} \in \mathcal{N}_k(x_i^{(m)}) \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where $A^{(m)}$ is the adjacency matrix of the mth view, $x_i^{(m)}$ and $x_i^{(m)}$ are the ith and jth instance vectors of the mth view, σ is the Gaussian scale parameter, and $\mathcal{N}_k(x_i^{(m)})$ is the k nearest neighbors of $x_i^{(m)}$.

To find the suitable low-dimensional latent representation matrix, we factorize the adjacency matrix $A^{(m)}$ into two non-negative matrices $F^{(m)}$ and $F^{(m)T}$, which is formulated as follows:

$$\begin{aligned} &\min_{F^{(m)}} \sum_{m=1}^{n_v} \|A^{(m)} - F^{(m)}F^{(m)T}\|_F^2 \\ &\text{s.t. } \forall m, \quad F^{(m)} \geq 0 \end{aligned} \tag{2}$$

where $F^{(m)}$ denotes the latent representation matrix in the mth view.

Note that the latent representation matrix F records the clustering structure of data instances, which can be used as a cluster indicator matrix to provide discriminative information for UFS tasks [38,39].

3.3. Pseudo-label sparse regression

Similar to [40], we employ the least square regression to measure the correlation between the pseudo-label space and low-dimensional latent representation space as follows:

$$\begin{aligned} &\min_{F^{(m)},W^{(m)},H^{(m)}} \sum_{m=1}^{n_v} \|F^{(m)} - X^{(m)}W^{(m)}H^{(m)}\|_F^2 \\ &\text{s.t. } \forall m, \quad W^{(m)} > 0, H^{(m)} > 0 \end{aligned} \tag{3}$$

where $W^{(m)}$ and $H^{(m)}$ denote the feature selection matrix and the regression coefficient matrix in the mth view, respectively.

Moreover, we impose an $\ell_{2,1}$-norm on W to ensure the sparsity of features. Besides, we impose the orthogonal constraint on F and W to ensure that each column or row contains at most one non-zero element. Thus, Eq. (3) can be converted as

$$\begin{aligned} &\min_{F^{(m)},W^{(m)},H^{(m)}} \sum_{m=1}^{n_v} \left\{ \|F^{(m)} - X^{(m)}W^{(m)}H^{(m)}\|_F^2 + \gamma\|W^{(m)}\|_{2,1} \right\} \\ &\text{s.t. } \forall m, \quad W^{(m)} > 0, H^{(m)} > 0, W^{(m)T}W^{(m)} = I_1, \\ &\qquad F^{(m)T}F^{(m)} = I_2 \end{aligned} \tag{4}$$

where γ is a balancing parameter to adjust the sparse regularization term.

3.4. View-specific graph regularization

For UFS tasks, it has been proved that the local geometrical structure of data is vital [41]. As an effective way to maintain the local geometrical structure of data, graph learning has been

extensively used. However, existing MUFS methods based on graph learning suffer from two shortcomings: (1) The manually fixed hyper-parameters are hard to tune, which might lead to sub-optimal results. (2) The performance of the fixed graph is greatly affected by noises, redundancy, and outliers contained in the original data. Different from them, we employ the sparse representation algorithm [42] to adaptively measure the similarity weights as follows:

$$\begin{aligned} &\min_{S^{(m)}} \sum_{m=1}^{n_v} \sum_{i,j=1}^{n} \|x_i^{(m)} - x_j^{(m)}\|_2^2 s_{ij}^{(m)} \\ &\text{s.t. } \forall m, i, \quad s_i^{(m)T}\mathbf{1} = 1, 1 \geq s_{i,j}^{(m)} \geq 0, diag(S^{(m)}) = 0 \end{aligned} \tag{5}$$

where $\|x_i^{(m)} - x_j^{(m)}\|_2^2$ denotes the Euclidean distance between $x_i^{(m)}$ and $x_j^{(m)}$, and $s_{ij}^{(m)}$ denotes the similarity between $x_i^{(m)}$ and $x_j^{(m)}$. Moreover, the normalization is imposed on S, i.e., $s_i^{(m)T}\mathbf{1} = 1$, which is equivalent to a sparse constraint, and $diag(S^{(m)}) = 0$ is used to avoid trivial solutions.

By jointly integrating Eqs. (3) and (5), the learning process of F, W, H, and S can promote each other. Thus, the joint learning model is formulated as

$$\begin{aligned} &\min_{F^{(m)},W^{(m)},H^{(m)}} \sum_{m=1}^{n_v} \Bigg\{ \|F^{(m)} - X^{(m)}W^{(m)}H^{(m)}\|_2^2 \\ &+ \lambda_1 \sum_{i,j=1}^{n} \|x_i^{(m)} - x_j^{(m)}\|_F^2 s_{ij}^{(m)} \Bigg\} \\ &\text{s.t. } \forall m, i, \quad s_i^{(m)T}\mathbf{1} = 1, 1 \geq s_{i,j}^{(m)} \geq 0, diag(S^{(m)}) = 0 \end{aligned} \tag{6}$$

where λ_1 is a balancing parameter.

However, the graph regularization inevitably leads to an additional balancing parameter λ_1, which increases the burden of parameter selection [43,44]. To address this issue, we design a parameter-free strategy. Specifically, according to Eq. (3), if $f_i^{(m)}$ and $f_j^{(m)}$ are the nearest neighbors of each other, $x_i^{(m)}W^{(m)}H^{(m)}$ and $x_j^{(m)}W^{(m)}H^{(m)}$ should also the nearest neighbors. Thus, we can derive that $f_i^{(m)} \approx x_i^{(m)}W^{(m)}H^{(m)}$. Then, Eq. (6) is converted as follows:

$$\begin{aligned} &\min_{F^{(m)},W^{(m)},H^{(m)}} \sum_{m=1}^{n_v} \sum_{i,j=1}^{n} \|f_i^{(m)} - x_j^{(m)}W^{(m)}H^{(m)}\|_2^2 s_{ij}^{(m)} \\ &\text{s.t. } \forall m, i, \quad s_i^{(m)T}\mathbf{1} = 1, 1 \geq s_{i,j}^{(m)} \geq 0, diag(S^{(m)}) = 0 \end{aligned} \tag{7}$$

3.5. Consensus graph regularization

To further maintain the local geometrical structure of sample space, we dynamically learn the consensus similarity graph of different views. Through the multi-view similarity graph learned in Section 3.4, the learning strategy of the consensus similarity graph is formulated as

$$\begin{aligned} &\min_{S^{(m)},\bar{S},\alpha^{(m)}} \sum_{m=1}^{n_v} \alpha^{(m)} \|\bar{S} - S^{(m)}\|_F^2 \\ &\text{s.t. } \forall i, \quad \bar{s}_i^T\mathbf{1} = 1, 1 \geq \bar{s}_{i,j} \geq 0, \end{aligned} \tag{8}$$

where $\bar{S}$ represents the consensus similarity matrix, and $\alpha^{(m)}$ represents adaptive view weighting of the mth view.

For two data instances $x_i^{(m)}$ and $x_j^{(m)}$, if they are close to each other, the corresponding pseudo-label vectors $f_i^{(m)}$ and $f_j^{(m)}$ should also be close to each other. Thus, the consensus graph regularization can be formulated as

$$\min_{\bar{S}} \sum_{m=1}^{n_v} \sum_{i,j=1}^{n} \|f_i^{(m)} - f_j^{(m)}\|_2^2 \bar{s}_{ij} \tag{9}$$

By combining Eqs. (2) and (9), the learning process of F and $\bar{S}$ can promote each other. The joint learning model is formulated as

$$\begin{aligned} &\min_{F^{(m)},\bar{S}} \sum_{m=1}^{n_v} \Bigg\{ \|A^{(m)} - F^{(m)}F^{(m)T}\|_F^2 + \lambda_2 \sum_{i,j=1}^{n} \|f_i^{(m)} - f_j^{(m)}\|_2^2 \bar{s}_{ij} \Bigg\} \\ &\text{s.t. } \forall m, \quad F^{(m)} > 0 \end{aligned} \tag{10}$$

where λ_2 is a balancing parameter.

Similar to Eq. (7), if $a_i^{(m)}$ and $a_j^{(m)}$ are the nearest neighbors to each other, $f_i^{(m)}F^{(m)T}$ and $f_j^{(m)}F^{(m)T}$ should also the nearest neighbor. Thus, we can derive that $a_i^{(m)} \approx f_i^{(m)}F^{(m)T}$. Then, Eq. (10) is converted as follows:

$$\begin{aligned} &\min_{F^{(m)},\bar{S}} \sum_{m=1}^{n_v} \sum_{i,j=1}^{n} \|a_i^{(m)} - f_j^{(m)}F^{(m)T}\|_2^2 \bar{s}_{ij} \\ &\text{s.t. } \forall m, \quad F^{(m)} > 0 \end{aligned} \tag{11}$$

3.6. Objective function

By combining Eqs. (4), (7), (8) and (11), the overall objective model of the proposed SDFS is formulated as follows:

$$\begin{aligned} &\min_{\phi} \sum_{m=1}^{n_v} \Bigg\{ \sum_{i,j=1}^{n} \|f_i^{(m)} - x_j^{(m)}W^{(m)}H^{(m)}\|_2^2 s_{ij}^{(m)} + \alpha^{(m)} \|\bar{S} - S^{(m)}\|_F^2 \\ &\quad + \sum_{i,j=1}^{n} \beta \|a_i^{(m)} - f_j^{(m)}F^{(m)T}\|_2^2 \bar{s}_{ij} + \gamma \|W^{(m)}\|_{2,1} \Bigg\} \\ &\text{s.t. } \forall m, i, \quad F^{(m)} > 0, W^{(m)} > 0, H^{(m)} > 0, W^{(m)T}W^{(m)} = I_1, \\ &\quad F^{(m)T}F^{(m)} = I_2, \bar{s}_i^T\mathbf{1} = 1, 1 \geq \bar{s}_{i,j} \geq 0, \\ &\quad diag(S^{(m)}) = 0, s_i^{(m)T}\mathbf{1} = 1, 1 \geq s_{i,j}^{(m)} \geq 0 \end{aligned} \tag{12}$$

where $\phi = \{F^{(m)}, W^{(m)}, H^{(m)}, S^{(m)}, \bar{S}, \alpha^{(m)}\}$, and β is the balancing parameter.

By solving the above objective function, the weight values of features can be calculated by $\|W_i\|_2$. Then, the features are arranged in descending order, and the top l features are chosen to form the new data matrix X_{new}.

3.7. Optimization

The optimization of SDFS is a non-convex problem. Thus, we proffer an alternating scheme of the augmented Lagrange multiplier (ALM) method [45] to solve this problem. Specifically, we alternately solve one variable by fixing other variables, and repeat this step until the objective function is converged.

We introduce five Lagrangian operators η, ω, θ, ϵ and τ, where θ, ϵ and τ are used to ensure F, W and H be non-negative. As reported by $\ell_{2,1}$-norm in [46], we impose a fairly tiny constant σ to avoid overflow. For the mth view, we define a diagonal matrix $Q^{(m)} \in R^{d^{(m)} \times d^{(m)}}$, in which the ith factor on its main diagonal is defined as

$$Q_{ii}^{(m)} = \frac{1}{2\max(\|q_i^{(m)}\|_2, \sigma)} \tag{13}$$

where $q_i^{(m)}$ denotes the ith row of $W^{(m)}$.

The optimization procedures are listed as follows.

Update $F^{(m)}$: Fix $W^{(m)}$, $H^{(m)}$, $S^{(m)}$, $\bar{S}$ and $\alpha^{(m)}$, and retain the

Algorithm 1 The proposed SDFS algorithm

Input: The original data matrix $X^{(1)}, ..., X^{(n_v)}$, the balancing parameters β and γ, the number of selected features l, the dimension of subspace r, and the number of categories c.
Initialize: Define matrices $F^{(1)}, ..., F^{(n_v)}$, $W^{(1)}, ..., W^{(n_v)}$, $H^{(1)}, ..., H^{(n_v)}$, $Q^{(1)}, ..., Q^{(n_v)}$, and $\overline{V}$. Calculate the similarity matrices $A^{(1)}, ..., A^{(n_v)}$, and $S^{(1)}, ..., S^{(n_v)}$, and initialize $\overline{S}$ by connecting S.
Update:
repeat
For each view, update $F^{(m)}$ with Eq. (14);
For each view, update $W^{(m)}$ with Eq. (19);
For each view, update $H^{(m)}$ with Eq. (24);
For each view, update $\alpha^{(m)}$ with Eq. (29);
For each view, update $S^{(m)}$ with Eq. (32);
Update $\overline{S}$ with Eq. (40);
until convergence
Output: Feature selection matrix W.
Feature selection: Calculate the evaluation values of all features through $\|W_i\|_2$, and arrange them in descending order. Then, the top l features are chosen to form the new data matrix X_{new}.

relevant terms containing $F^{(m)}$, Eq. (12) can be converted as

$$\begin{aligned} &\min \sum_{i,j=1}^{n} \{\|f_i^{(m)} - x_j^{(m)} W^{(m)} H^{(m)}\|_2^2 s_{ij}^{(m)} + \beta \|a_i^{(m)} - f_j^{(m)} F^{(m)T}\|_2^2 \bar{s}_{ij}\} \\ &\text{s.t.} \quad F^{(m)} > 0, F^{(m)T} F^{(m)} = I_2 \end{aligned} \tag{14}$$

By converting Eq. (14) into a trace form, we can get

$$\begin{aligned} \mathcal{L}(F^{(m)}) = \ & \mathrm{Tr}(F^{(m)T} D^{(m)} F^{(m)}) - 2\mathrm{Tr}(F^{(m)T} S^{(m)} Z^{(m)}) \\ & + \frac{\eta}{2} \mathrm{Tr}(F^{(m)T} F^{(m)} - I_2)(F^{(m)T} F^{(m)} - I_2)^T \\ & + \beta \mathrm{Tr}(F^{(m)} F^{(m)T} \overline{D} F^{(m)} F^{(m)T}) + \mathrm{Tr}(\theta F^{(m)T}) \\ & - 2\beta \mathrm{Tr}(A^{(m)T} \overline{S} F^{(m)} F^{(m)T}) \end{aligned} \tag{15}$$

where $Z^{(m)} = X^{(m)} W^{(m)} H^{(m)}$.

Solving the partial derivative of $\mathcal{L}(F^{(m)})$ w.r.t. $F^{(m)}$, we can get

$$\begin{aligned} \frac{\partial \mathcal{L}(F^{(m)})}{\partial F^{(m)}} = \ & D^{(m)} F^{(m)} - S^{(m)} Z^{(m)} - 2\beta A^{(m)T} \overline{S} F^{(m)} + \beta \overline{D} E_1^{(m)} \\ & + \beta F^{(m)} F^{(m)T} \overline{D} F^{(m)} + \eta (E_1^{(m)} - F^{(m)}) + \theta \end{aligned} \tag{16}$$

where $E_1^{(m)} = F^{(m)} F^{(m)T} F^{(m)}$.

By employing the Karush-Kuhn–Tucker (KKT) conditions [47,48], i.e., $\theta_{ij} F_{ij}^{(m)} = 0$, we can get

$$\begin{aligned} &\Big((D^{(m)} F^{(m)} - S^{(m)} Z^{(m)} - 2\beta A^{(m)T} \overline{S} F^{(m)} + \beta \overline{D} E_1^{(m)} \\ &+ \beta F^{(m)} F^{(m)T} \overline{D} F^{(m)} + \eta E_1^{(m)} - \eta F^{(m)}) \Big)_{ij} F_{ij}^{(m)} = 0 \end{aligned} \tag{17}$$

Then, the iteration rule of $F^{(m)}$ is obtained as follows:

$$F_{ij}^{(m)} \leftarrow F_{ij}^{(m)} \frac{\left(2\beta A^{(m)T} \overline{S} F^{(m)} + S^{(m)} Z^{(m)} + \eta F^{(m)}\right)_{ij}}{\left(\beta F^{(m)} F^{(m)T} \overline{D} F^{(m)} + (\beta \overline{D} + \eta) E_1^{(m)} + D^{(m)} F^{(m)}\right)_{ij}} \tag{18}$$

Update $W^{(m)}$: Fix $F^{(m)}$, $H^{(m)}$, $S^{(m)}$, $\overline{S}$ and $\alpha^{(m)}$, and retain the relevant terms containing $W^{(m)}$, Eq. (12) can be converted as

$$\begin{aligned} &\min \sum_{i,j=1}^{n} \|f_i^{(m)} - x_j^{(m)} W^{(m)} H^{(m)}\|_2^2 s_{ij}^{(m)} + \gamma \|W^{(m)}\|_{2,1} \\ &\text{s.t.} \quad W^{(m)} > 0, W^{(m)T} W^{(m)} = I_1 \end{aligned} \tag{19}$$

By converting Eq. (19) into a trace form, we can get

$$\begin{aligned} \mathcal{L}(W^{(m)}) = \ & \mathrm{Tr}(Z^{(m)T} D^{(m)} Z^{(m)}) - 2\mathrm{Tr}(F^{(m)T} S^{(m)} Z^{(m)}) \\ & + \frac{\omega}{2} \mathrm{Tr}(W^{(m)T} W^{(m)} - I_1)(W^{(m)T} W^{(m)} - I_1)^T \\ & + \gamma \mathrm{Tr}(W^{(m)T} Q^{(m)} W^{(m)}) + \mathrm{Tr}(\epsilon W^{(m)T}) \end{aligned} \tag{20}$$

Solving the partial derivative of $\mathcal{L}(W^{(m)})$ w.r.t. $W^{(m)}$, we can get

$$\begin{aligned} \frac{\partial \mathcal{L}(W^{(m)})}{\partial W^{(m)}} = \ & X^{(m)T} D^{(m)} X^{(m)} W^{(m)} H^{(m)} H^{(m)T} + \gamma Q^{(m)} W^{(m)} \\ & - X^{(m)T} S^{(m)T} F^{(m)} H^{(m)T} + \omega (E_2^{(m)} - W^{(m)}) + \epsilon \end{aligned} \tag{21}$$

where $E_2^{(m)} = W^{(m)} W^{(m)T} W^{(m)}$.

Through the KKT conditions, i.e., $\epsilon_{ij} W_{ij}^{(m)} = 0$, we can get

$$\begin{aligned} &\Big((X^{(m)T} D^{(m)} X^{(m)} W^{(m)} H^{(m)} H^{(m)T} - X^{(m)T} S^{(m)T} F^{(m)} H^{(m)T} \\ &+ \gamma Q^{(m)} W^{(m)} + \omega (E_2^{(m)} - W^{(m)})) \Big)_{ij} W_{ij}^{(m)} = 0 \end{aligned} \tag{22}$$

Then, the iteration rule of $W^{(m)}$ is obtained as follows:

$$W_{ij}^{(m)} \leftarrow W_{ij}^{(m)} \frac{(X^{(m)T} S^{(m)T} F^{(m)} H^{(m)T} + \omega W^{(m)})_{ij}}{(X^{(m)T} D^{(m)} Z^{(m)} H^{(m)T} + \gamma Q^{(m)} W^{(m)} + \omega E_2^{(m)})_{ij}} \tag{23}$$

Update $H^{(m)}$: Fix $F^{(m)}$, $W^{(m)}$, $S^{(m)}$, $\overline{S}$ and $\alpha^{(m)}$, and retain the relevant terms containing $H^{(m)}$, Eq. (12) can be converted as

$$\begin{aligned} &\min \sum_{i,j=1}^{n} \|f_i^{(m)} - x_j^{(m)} W^{(m)} H^{(m)}\|_2^2 s_{ij}^{(m)} \\ &\text{s.t.} \quad H^{(m)} > 0 \end{aligned} \tag{24}$$

By converting Eq. (24) into a trace form, we can get

$$\begin{aligned} \mathcal{L}(H^{(m)}) = \ & \mathrm{Tr}(Z^{(m)T} D^{(m)} Z^{(m)}) + \mathrm{Tr}(\tau H^{(m)T}) \\ & - 2\mathrm{Tr}(F^{(m)T} S^{(m)} Z^{(m)}) \end{aligned} \tag{25}$$

Solving the partial derivative of $\mathcal{L}(H^{(m)})$ w.r.t. $H^{(m)}$, we can get

$$\frac{\partial \mathcal{L}(H^{(m)})}{\partial H^{(m)}} = W^{(m)T} X^{(m)T} D^{(m)} Z^{(m)} - W^{(m)T} X^{(m)T} S^{(m)T} F^{(m)} + \tau \tag{26}$$

Through the KKT conditions, i.e., $\tau_{ij} H_{ij}^{(m)} = 0$, we can get

$$\Big((W^{(m)T} X^{(m)T} D^{(m)} Z^{(m)} - W^{(m)T} X^{(m)T} S^{(m)T} F^{(m)}) \Big)_{ij} H_{ij}^{(m)} = 0 \tag{27}$$

Then, the iteration rule of $H^{(m)}$ is obtained as follows:

$$H_{ij}^{(m)} \leftarrow H_{ij}^{(m)} \frac{(W^{(m)T} X^{(m)T} S^{(m)T} F^{(m)})_{ij}}{(W^{(m)T} X^{(m)T} D^{(m)} Z^{(m)})_{ij}} \tag{28}$$

Update $\alpha^{(m)}$: Fix $F^{(m)}$, $W^{(m)}$, $H^{(m)}$, $S^{(m)}$ and $\overline{S}$, and retain the relevant terms containing $\alpha^{(m)}$. According to [49], if the weighting $\alpha^{(m)}$ are fixed, we can derive

$$\min \alpha^{(m)} \|\overline{S} - S^{(m)}\|_F^2 = \min \sqrt{\|\overline{S} - S^{(m)}\|_F^2} \tag{29}$$

Solving the partial derivative of the above function w.r.t. $\overline{S}$, we can get

$$\frac{\partial \sqrt{\|\overline{S} - S^{(m)}\|_F^2}}{\partial \overline{S}} = \frac{\partial \|\overline{S} - S^{(m)}\|_F^2}{(2\sqrt{\|\overline{S} - S^{(m)}\|_F^2}) \partial \overline{S}} \tag{30}$$

According to Eqs. (29) and (30), we can derive the iteration rule of $\alpha^{(m)}$ as follows:

$$\alpha^{(m)} = \frac{1}{(2\sqrt{\|\overline{S} - S^{(m)}\|_F^2})} \tag{31}$$

Update $S^{(m)}$: Fix $F^{(m)}$, $W^{(m)}$, $H^{(m)}$, $\bar{S}$ and $\alpha^{(m)}$, and retain the relevant terms containing $S^{(m)}$, Eq. (12) can be converted as

$$\begin{aligned} &\min \sum_{i,j=1}^{n} \|f_i^{(m)} - x_j^{(m)} W^{(m)} H^{(m)}\|_2^2 s_{ij}^{(m)} + \alpha^{(m)} \|\bar{S} - S^{(m)}\|_F^2 \\ &\text{s.t.} \quad \forall i, \quad diag(S^{(m)}) = 0, s_i^{(m)T} \mathbf{1} = 1, 1 \geq s_{i,j}^{(m)} \geq 0 \end{aligned} \tag{32}$$

Note that Eq. (32) is independent between different i. Thus, we can divide the Lagrangian function of Eq. (32) into n sub-problems as follows:

$$\begin{aligned} \min \sum_{j=1}^{n} &\frac{1}{2} \|f_i^{(m)} - x_j^{(m)} W^{(m)} H^{(m)}\|_2^2 s_{ij}^{(m)} + \frac{\alpha^{(m)}}{2} \|\bar{S} - S^{(m)}\|_F^2 \\ &+ \rho \|s_i^{(m)}\|_2^2 - \xi (s_i^{(m)} \mathbf{1} - 1) - \zeta^T s_i^{(m)} \end{aligned} \tag{33}$$

where ρ is a regularization parameter, ξ is a Lagrangian coefficient scalar, and ζ is a Lagrangian coefficient vector.

Here, we define $d_{ij}^{(m)} = \|f_i^{(m)} - x_j^{(m)} W^{(m)} H^{(m)}\|_2^2$, and $d_j^{(m)}$ is the jth vector of $d_{ij}^{(m)}$. Thus, Eq. (33) is rewritten as follows:

$$\begin{aligned} \mathcal{L}(s_i^{(m)}, \xi, \zeta) = &\frac{1}{2} \|s_i^{(m)} + \frac{d_i^{(m)}}{2\rho}\|_2^2 + \frac{\alpha^{(m)}}{2\rho} \|\bar{s}_i - s_i^{(m)}\|_2^2 \\ &- \xi (s_i^{(m)} \mathbf{1} - 1) - \zeta^T s_i^{(m)} \end{aligned} \tag{34}$$

For the jth entry of $s_i^{(m)}$, the partial derivative of $\mathcal{L}(s_i^{(m)}, \xi, \zeta)$ w.r.t. $s_i^{(m)}$ can be written as

$$\frac{\partial \mathcal{L}(s_{ij}^{(m)}, \xi, \zeta)}{\partial s_{ij}^{(m)}} = s_{ij}^{(m)} + \frac{d_{ij}^{(m)}}{2\rho} - \frac{\alpha^{(m)} (\bar{s}_{ij} - s_{ij}^{(m)})}{\rho} - \xi - \zeta_j \tag{35}$$

Through the KKT conditions, i.e., $s_{ij}^{(m)} \zeta_j = 0$, we can get

$$s_{ij}^{(m)} = \left(\frac{2\alpha^{(m)} \bar{s}_{ij} + 2\rho\xi - d_{ij}^{(m)}}{2\rho + 2\alpha^{(m)}} \right)_+ \tag{36}$$

where $(a)_+ = \max(a, 0)$.

Assume $\{d_{i1}^{(m)}, \ldots, d_{in}^{(m)}\}$ are arranged in ascending order. Assuming $s_i^{(m)}$ has k non-zero entries, we have $s_{ik}^{(m)} > 0$ and $s_{i(k+1)}^{(m)} = 0$. Besides, by combining Eq. (36) and the constraint $s_i^{(m)} \mathbf{1} = 1$. Then, we can get

$$\begin{cases} 2\alpha^{(m)} \bar{s}_{ik} + 2\rho\xi - d_{ik}^{(m)} > 0 \\ 2\alpha^{(m)} \bar{s}_{i(k+1)} + 2\rho\xi - d_{i(k+1)}^{(m)} \leqslant 0 \\ \xi = \frac{2\rho + \alpha^{(m)} + 2\sum_{h=1}^{k} d_{ih}^{(m)}}{2\rho k} \end{cases} \tag{37}$$

According to Eq. (37), to ensure that the optimal solution for $s_i^{(m)}$ has k non-zero entries, ρ is defined as

$$\rho = \frac{k d_{i(k+1)}^{(m)} - \sum_{h=1}^{k} d_h^{(m)} - 2k\alpha^{(m)} \bar{s}_{i(k+1)} - 2\alpha^{(m)}}{2} \tag{38}$$

Then, the iteration rule of $s_{ij}^{(m)}$ is obtained as follows:

$$s_{ij}^{(m)} = \begin{cases} \frac{d_{i(k+1)}^{(m)} - d_{ij}^{(m)} + 2\alpha^{(m)} \bar{s}_{ij} - 2\alpha^{(m)} \bar{s}_{i(k+1)}}{k d_{i(k+1)}^{(m)} - \sum_{h=1}^{k} d_{ih}^{(m)} + 2\sum_{h=1}^{k} \alpha^{(m)} \bar{s}_{ih} - 2k\alpha^{(m)} \bar{s}_{i(k+1)}}, & j \leqslant k \\ 0, & j > k \end{cases} \tag{39}$$

Update $\bar{S}$: Fix $F^{(m)}$, $W^{(m)}$, $H^{(m)}$, $S^{(m)}$ and $\alpha^{(m)}$, and retain the relevant terms containing $\bar{S}$, Eq. (12) can be converted as

$$\begin{aligned} &\min \sum_{m=1}^{n_v} \left\{ \alpha^{(m)} \|\bar{S} - S^{(m)}\|_F^2 + \sum_{i,j=1}^{n} \beta \|a_i^{(m)} - f_j^{(m)} F^{(m)T}\|_2^2 \bar{s}_{ij} \right\} \\ &\text{s.t.} \quad \forall i, \quad \bar{s}_i^T \mathbf{1} = 1, 1 \geq \bar{s}_{i,j} \geq 0 \end{aligned} \tag{40}$$

Similar to Eq. (32), we can divide the Lagrangian function of Eq. (40) into n sub-problems as follows:

$$\begin{aligned} \min \sum_{m=1}^{n_v} &\left\{ \frac{\alpha^{(m)}}{2} \|\bar{S} - S^{(m)}\|_F^2 + \sum_{j=1}^{n} \frac{\beta}{2} \|a_i^{(m)} - f_j^{(m)} F^{(m)T}\|_2^2 \bar{s}_{ij} \right\} \\ &- \varsigma (\bar{s}_i^T \mathbf{1} - 1) - \vartheta^T \bar{s}_i \end{aligned} \tag{41}$$

where ς represents a Lagrangian coefficient scalar, ϑ represents a Lagrangian coefficient vector.

Define $p_{ij}^{(m)} = \|a_i^{(m)} - f_j^{(m)} F^{(m)T}\|_2^2$, and $p_{ij}^{(m)}$ is the jth element of vector $p_i^{(m)}$. Eq. (41) can be rewritten as

$$\mathcal{L}(\bar{s}_i, \varsigma, \vartheta) = \sum_{m=1}^{n_v} \|\bar{s}_i - s_i^{(m)} + \frac{\beta p_i^{(m)}}{2 n_v \alpha^{(m)}}\|_2^2 - \varsigma (\bar{s}_i^T \mathbf{1} - 1) - \vartheta^T \bar{s}_i \tag{42}$$

Through the KKT conditions, we can get

$$\begin{cases} \forall j, \quad \sum_{m=1}^{n_v} (\bar{s}_{ij} - g_j^{(m)}) - \varsigma - \vartheta_j = 0 \\ \forall j, \quad \bar{s}_{ij} \geq 0, \quad \vartheta_j \geq 0, \quad \bar{s}_{ij} \vartheta_j = 0 \end{cases} \tag{43}$$

where $g^{(m)} = s_i^{(m)} + \frac{\beta p_i^{(m)}}{2 n_v \alpha^{(m)}}$.

Then, according to the constraint, i.e., $\bar{s}_i^T \mathbf{1} - 1 = 0$, we get

$$\varsigma = \frac{n_v - \sum_{m=1}^{n_v} g^{(m)T} \mathbf{1} - \vartheta^T \mathbf{1}}{n} \tag{44}$$

For $\forall j$, we can derive the optimal solution for $\bar{s}_{ij}$ as follows:

$$\bar{s}_{ij} = \frac{n_v + n \sum_{m=1}^{n_v} g_j^{(m)} + n\vartheta_j - \sum_{m=1}^{n_v} \mathbf{1}^T g_j^{(m)T} \mathbf{1} - \mathbf{1}^T \vartheta_j^T \mathbf{1}}{n_v n} \tag{45}$$

According to Eqs. (43) and (45), we can derive

$$\vartheta^* = \frac{\sum_{j=1}^{n} (\bar{s}_{ij} + \vartheta^* - g_j^*)}{n} = \frac{\sum_{j=1}^{n} (\vartheta^* - g_j^*)_+}{n} \tag{46}$$

where $g_j^* = \frac{n_v + n \sum_{m=1}^{n_v} g_j^{(m)} - \sum_{m=1}^{n_v} \mathbf{1}^T g_j^{(m)T} \mathbf{1}}{n_v n}$ and $\vartheta^* = \frac{\mathbf{1}^T \vartheta_j^T}{n_v n}$.

Here, we first define a function of χ, i.e. $f(\chi) = \frac{\sum_{j=1}^{n} (\chi - g_j^*)_+}{n} - \chi$. Then, we utilize the Newton method to solve the root finding problem $f(\vartheta^*) = 0$, which is written as follows:

$$\vartheta_{t+1}^* = \vartheta_t^* - \frac{f(\vartheta_t^*)}{f'(\vartheta_t^*)} \tag{47}$$

Similar to Eq. (46), the iteration rule without additional parameters for $\bar{s}_{ij}$ is obtained as follows:

$$\bar{s}_{ij} = g_j^* - \vartheta^* + \frac{\vartheta_j}{n_v} = (g_j^* - \vartheta^*)_+ \tag{48}$$

3.8. Computational complexity

In this section, we analyze the time computational complexity of SDFS. Specifically, the optimization of SDFS can be divided into six sub-processes. Thus, the computational complexity of updating $F^{(m)}$ is $O(n^3 + n^2 d^{(m)} + n d^{(m)} r)$, where $c \ll n$ and $r \ll d$. The computational complexity of optimizing $W^{(m)}$ is $O(n^2 d^{(m)} + n d^{(m)2} + d^{(m)2} r + n d^{(m)} r)$. The computational complexity of optimizing $H^{(m)}$ is $O(n^2 r^{(m)} + n d^{(m)} r + d^{(m)} r^2 + r^2 c)$. The computational complexity of optimizing $\alpha^{(m)}$ is $O(n^2 d^{(m)})$. The computational complexity of optimizing $S^{(m)}$ is $O(n d^{(m)} k)$, where k is the number of neighbors and $k \ll n$. The computational complexity of optimizing $\bar{S}$ is $O(nc)$. Thus, the total time complexity is $O\left(t n_k (n^3 + n^2 d^{(m)} + n d^{(m)2} + d^{(m)2} r) \right)$, where t is the number of iterations.

Table 2
Details of datasets.

Dataset	Instances	Views	Classes	Features
MSRC-v1	210	5	7	1302, 48, 512, 256, 210
Mfeat	2000	6	10	216, 76, 64, 6, 240, 47
Calthch101-7	1474	6	7	48, 40, 254, 1984, 512, 928
Coil20	1440	3	20	944, 324, 512
ORL	400	3	40	4096, 3304, 6750
Youtube	1596	2	11	750, 750

4. Experiments

In this section, we compare the proposed SDFS with several other state-of-the-art methods on six benchmark datasets to validate its efficacy.

4.1. Datasets

To evaluate the effectiveness of SDFS, we use six benchmark datasets in our experiments, including MSRC-v1 [50], Mfeat [51], Calthch101-7 [52], Coil20 [53], ORL [54], and Youtube [55]. For clarity, Table 2 provides the statistical information of these datasets, and the description of these datasets is given as follows:

- **MSRC-v1** [50]: MSRC-v1 composes of 240 images with 8 object classes, in which each image is associated with five types of views. According to the experimental setup in [56], we select 210 images and seven object classes, i.e., tree, bicycle, airplane, face, building, and cow.
- **Mfeat** [51]: Mfeat is a widely-used handwritten digital dataset, which consists of 2000 instances corresponding to 10 digits, i.e., $(0 \sim 9)$. It is gathered from two handwritten digital datasets, i.e., the MNIST dataset and the USPS dataset. Each sample is described by six types of features.
- **Caltech101-7** [52]: Caltech101-7 composes of 1474 images with 101 object classes, in which each image is captured for object recognition problems and associated with six types of views. According to the experimental setup in [57], we select seven object classes, i.e., faces, Garfield, motorbikes, Dolla-bill, Windsor-chair, stop-sign, and Snoopy.
- **Coil20** [53]: Coil20 composes of 1440 images from the Columbia object image library. According to the experimental setup in [58], we extracted three types of features as different views, i.e., histogram of oriented gradients (HOG), local binary pattern (LBP), and global information features.
- **ORL** [54]: ORL composes of 400 face images captured from 40 humans with varying facial expressions, facial wears, illuminations, taking times, and angles. Each sample is described by three views, i.e., intensity, LBP, and texture.
- **Youtube** [55]: Youtube composes of 1596 video sequences in 11 actions, in which each video sequence is extracted from two views, i.e., global information features and scale-invariant feature transform (SIFT) features.

4.2. Compared algorithams

To demonstrate the superiority of SDFS, we compare it with several state-of-the-art algorithms, and the details are given as follows:

- **Baseline** : Baseline uses all features to cluster.
- **UDFS** [20]: UDFS is a classic representative single-view UFS algorithm. In multi-view tasks, it combines all features into a single view as input to perform the process of feature selection.
- **DGSLFS** [22]: DGSLFS is a single-view UFS algorithm. In multi-view tasks, it combines all features into a single view as input to perform the process of feature selection.
- **SLNMF** [26]: SLNMF is a new single-view UFS algorithm. In multi-view tasks, it combines all features into a single view as input to perform the process of feature selection.
- **CGMV-UFS** [35]: CGMV-UFS restricts the clustering index matrix of each view by learning the consensus matrix.
- **ACSL** [32]: ACSL dynamically learns the ideal collaborative similarity structure and the desirable neighbor assignment.
- **NGSL** [33]: NGSL employs the rank constraint to ensure that the adaptive similarity graph has an ideal structure.
- **TLR-MFS** [36]: TLR-MFS imposes the low-rank tensor regularization on the similarity graph to capture consistent information across views.

4.3. Experimental setup

To achieve good performance of each algorithm, we adjust the balancing parameters of each comparison method according to the original papers and record the best results under the optimal parameters. For SDFS, we set the balancing parameters from the grid $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2, 10^3\}$. Besides, we set the values of selected feature ratio varying from 3% to 15% with 3% intervals. Finally, for each method, we perform K-means clustering algorithm 40 times and record the average clustering results with standard deviation (std).

Two widely used clustering evaluation criteria, i.e., clustering accuracy (ACC) and normalized mutual information (NMI), are employed in our experiments [59,60].

4.4. Experimental results

In this section, we set the percentage of selected features to 9% for all methods. For SDFS and other compared methods, we report the mean results with std in Tables 3 and 4. In addition, to clearly show the performance, we make the best values bold and underline the second-best values. Meanwhile, we report the mean results under different percentages of selected features in Figs. 1 and 2, in which the x-axis and y-axis denote the ratio of selected features and the values of ACC or NMI. From these results, we have the following observations:

(1) First, SDFS achieves better performance on almost all benchmark datasets in comparison with other methods. The reason lies in that the proposed SDFS can make fully utilize the discriminative information and geometrical structure information in multi-view data.
(2) Second, the performance of MUFS algorithms is superior to that of single-view UFS algorithms on most datasets. The reason is that the MUFS algorithms consider the correlation between different views.
(3) Third, compared with the linear regression based MUFS algorithms, i.e., ACSL and NGSL, the proposed SDFS achieves higher performance. The reason might be that both ACSL and NGSL fail to fully consider the discriminative information of the original data.
(4) Fourth, it is worth noting that the proposed SDFS obtains better performance than the Baseline on all datasets excluding Caltech101-7, which verifies that the selected feature subset using SDFS can significantly improve the clustering performance.
(5) Last, for the Caltech101-7 dataset, the NMI value of Baseline is higher than that of other methods. The reason might be that $3\% \sim 15\%$ are not the optimal percentages of selected features for Caltech101-7. Nonetheless, the proposed SDFS achieves higher ACC results compared with

Table 3
Clustering performance (ACC ± std%) of different methods on six datasets.

Dataset	Baseline	UDFS	DGSLFS	SLNMF	CGMV-UFS	ACSL	NGSL	TLR-MFS	SDFS
MSRC-v1	77.86 ± 4.58	48.45 ± 3.62	63.26 ± 3.49	67.02 ± 2.42	62.67 ± 1.36	75.21 ± 4.28	65.76 ± 2.98	78.92 ± 3.52	**83.67 ± 5.10**
Mfeat	80.76 ± 5.19	69.34 ± 0.57	69.54 ± 1.19	78.86 ± 4.53	72.74 ± 2.14	88.51 ± 1.89	86.61 ± 2.09	89.78 ± 0.96	**91.42 ± 1.17**
Caltech101-7	55.86 ± 2.79	39.39 ± 1.06	42.92 ± 4.05	50.21 ± 3.44	50.40 ± 2.25	52.77 ± 2.11	51.20 ± 1.92	55.99 ± 2.79	**61.45 ± 2.38**
Coil20	67.42 ± 2.80	61.79 ± 3.74	67.22 ± 3.07	69.91 ± 2.81	65.96 ± 2.07	70.56 ± 3.34	73.08 ± 2.92	70.52 ± 3.39	**73.67 ± 2.66**
ORL	59.54 ± 2.36	46.15 ± 2.61	50.25 ± 2.98	51.39 ± 2.17	51.88 ± 2.65	51.75 ± 3.35	55.94 ± 3.34	57.18 ± 3.35	**61.16 ± 3.53**
Youtube	14.36 ± 0.11	22.52 ± 2.19	20.19 ± 1.67	21.64 ± 0.74	19.47 ± 0.53	28.29 ± 2.11	18.97 ± 1.62	27.73 ± 1.04	**29.84 ± 0.30**

Table 4
Clustering performance (NMI ± std%) of different methods on six datasets.

Dataset	Baseline	UDFS	DGSLFS	SLNMF	CGMV-UFS	ACSL	NGSL	TLR-MFS	SDFS
MSRC-v1	68.76 ± 4.27	34.51 ± 3.09	53.65 ± 2.58	58.26 ± 2.59	52.46 ± 1.18	68.54 ± 2.34	55.83 ± 2.91	69.75 ± 3.62	**77.04 ± 3.85**
Mfeat	80.17 ± 0.95	61.05 ± 0.63	68.89 ± 0.71	76.98 ± 0.97	67.76 ± 1.08	82.27 ± 0.82	82.70 ± 1.78	82.81 ± 0.79	**84.82 ± 0.87**
Caltech101-7	**50.38 ± 1.09**	33.42 ± 0.61	40.27 ± 0.64	42.49 ± 2.15	35.48 ± 1.07	45.94 ± 2.16	44.74 ± 0.92	45.16 ± 1.94	47.36 ± 2.28
Coil20	79.95 ± 1.49	73.95 ± 2.23	77.92 ± 1.73	79.49 ± 0.93	77.48 ± 1.16	78.46 ± 1.49	**82.67 ± 1.44**	79.17 ± 1.68	81.74 ± 1.32
ORL	76.97 ± 0.80	61.70 ± 2.77	67.11 ± 1.46	70.18 ± 1.18	71.55 ± 1.25	68.17 ± 2.93	71.82 ± 2.43	74.74 ± 2.07	**77.91 ± 2.30**
Youtube	3.24 ± 0.01	13.62 ± 1.84	10.41 ± 1.18	14.01 ± 1.21	9.42 ± 0.79	22.92 ± 1.21	9.66 ± 1.62	20.79 ± 1.38	**24.04 ± 0.23**

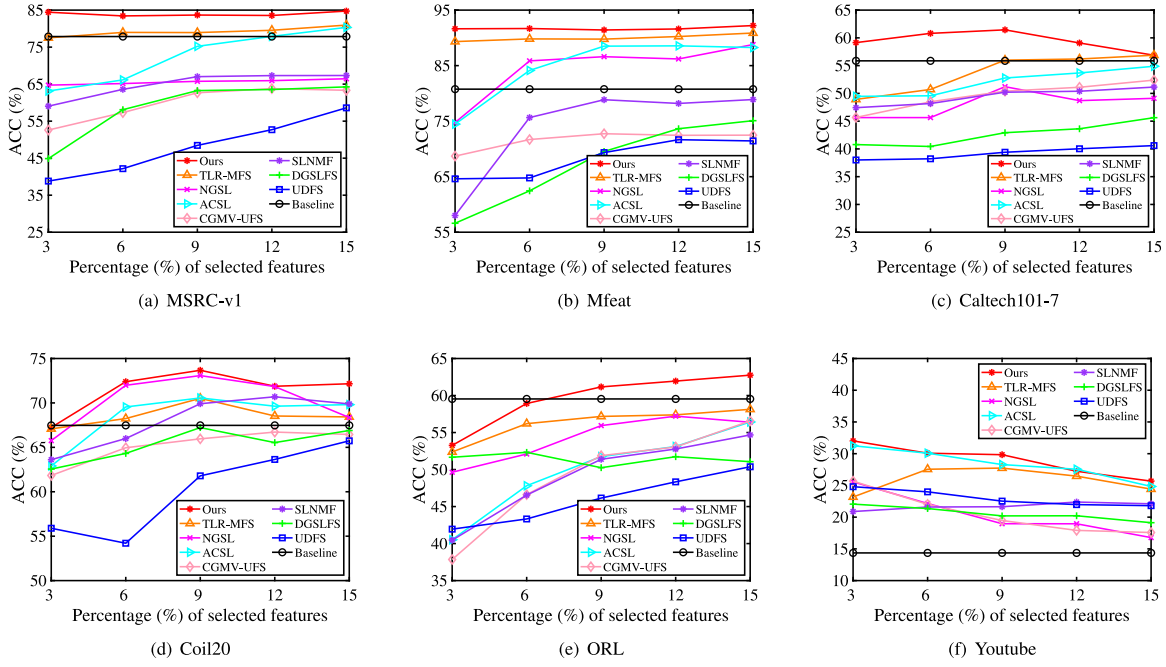


Fig. 1. ACC of different feature selection algorithms on eight datasets.

Baseline. As can be seen from Fig. 3, when the selected feature ratio increases, the NMI of the proposed method fluctuates between 44.21% ∼ 52.84%. And when selecting a 35% feature ratio, it reaches the maximum value of 52.84%, which exceeds the Baseline by 2.46%. This indicates the effectiveness of the proposed method.

Furthermore, to intuitively illustrate the effects of the proposed method, we visualize the clustering results using t-SNE in Fig. 4. It can be observed from the figure, the data points of X_{new} are divided into different clusters with relatively clear borders. In contrast, the clusters of original data X are blended with each other.

4.5. Parameter sensitivity analysis

In this section, we analyze the impacts of β and γ. Since there is no prior information about these parameters, we determine it by grid search in a heuristic strategy as previous works [6,26,31]. Specifically, the values of β and γ are chosen in the range of $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^{2}, 10^{3}\}$. The results are reported in Figs. 5 and 6.

It can be observed from the figures that when the values of β and γ are adjusted, the ACC and NMI values of the proposed SDFS do not change obviously on most datasets, especially on Mfeat, Coil20 and ORL. For Youtube, when $10^{-3} \leq \beta \leq 1$, the ACC and NMI results of the proposed SDFS are relatively good, and the performance is stable under other parameter combinations. For Caltech101-7, the ACC values of the proposed SDFS are stable for γ, and it only fluctuates within a small range for β. For the remaining cases, the ACC and NMI values occasionally fluctuate, but they are relatively smooth in general.

The experimental results show that the proposed SDFS is insensitive to the two balancing parameters β and γ on most datasets, which demonstrates that the proposed SDFS model is robust. Although some combinations of β and γ may cause fluctuations in clustering performance, SDFS can still achieve stable performance within a broad range.

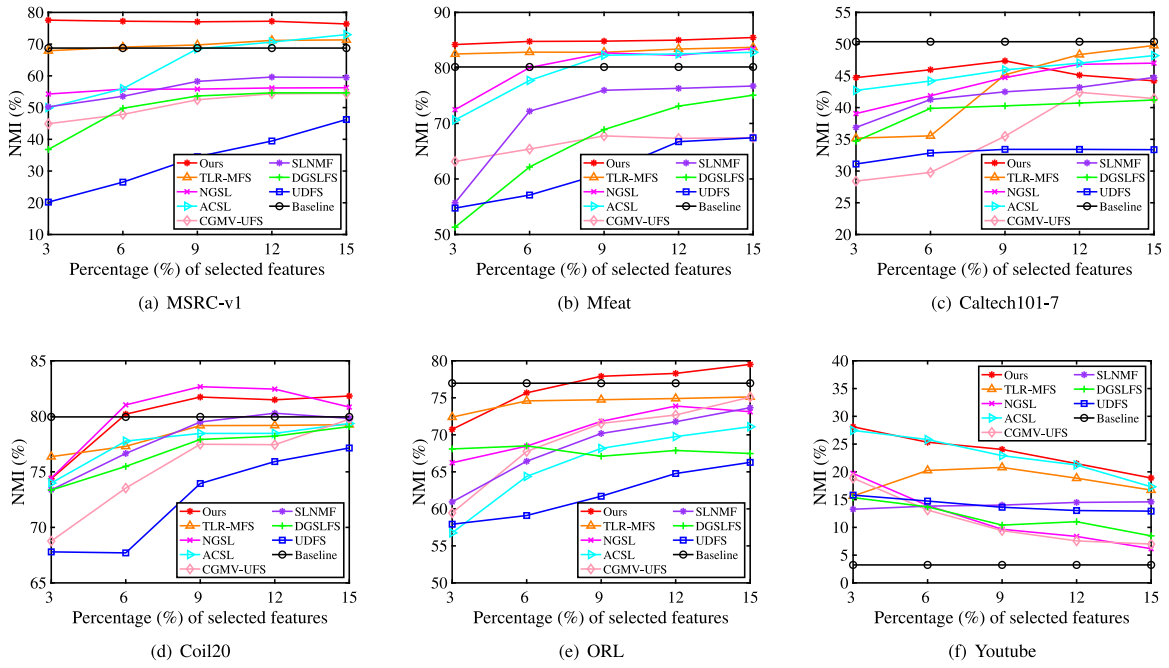


Fig. 2. NMI of different feature selection algorithms on eight datasets.

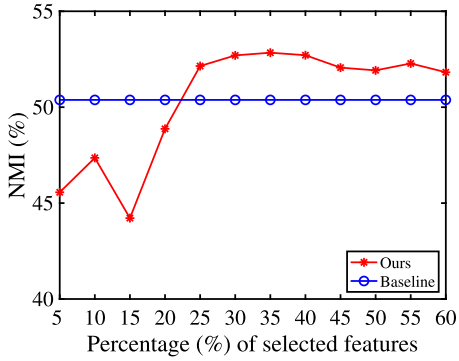


Fig. 3. NMI of the proposed SDFS with different selected feature ratios on Caltech101-7 dataset.

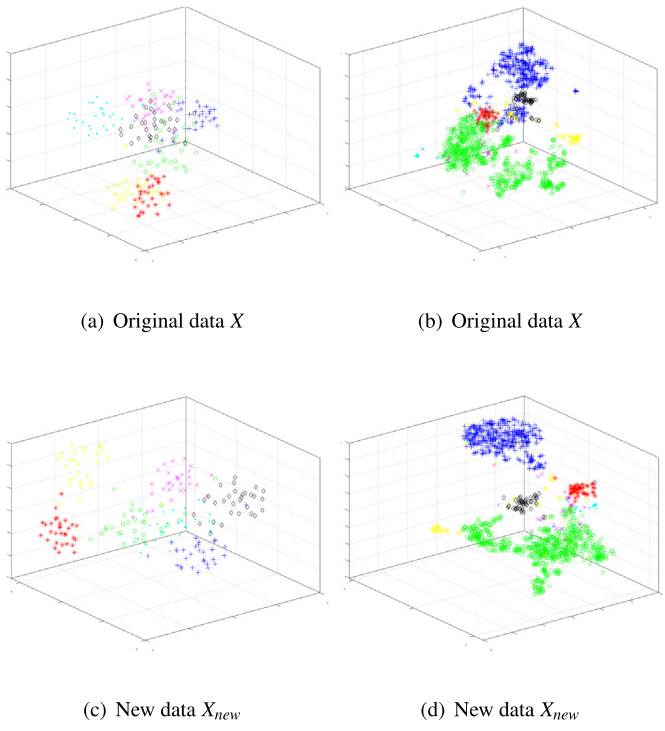


Fig. 4. t-SNE visualization of clustering results by SDFS and Baseline on the MSRC-v1 and Caltech101-7 datasets, where (a) and (c) are the results on MSRC-v1, and (b) and (d) are the results on Caltech101-7.

4.6. Ablation analysis

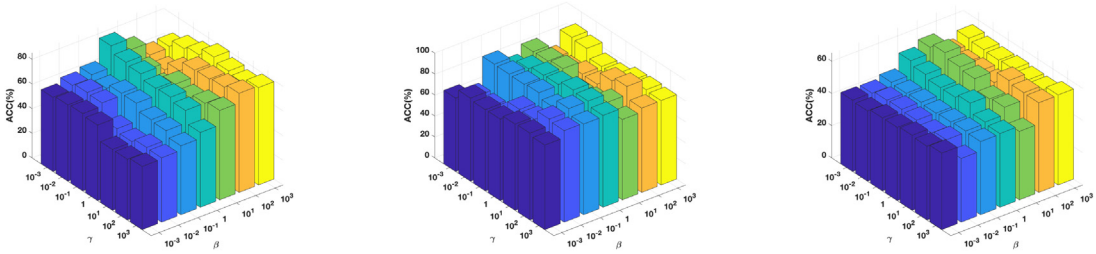
In this section, we analyze the effectiveness of consensus graph learning and latent representation learning. Specifically, we consider two special cases of SDFS:

- **SDFS**$_1$: We set the balancing parameter β to zero.
- **SDFS**$_2$: We set the adaptive weighting parameter α and balancing parameter β to zero, simultaneously.

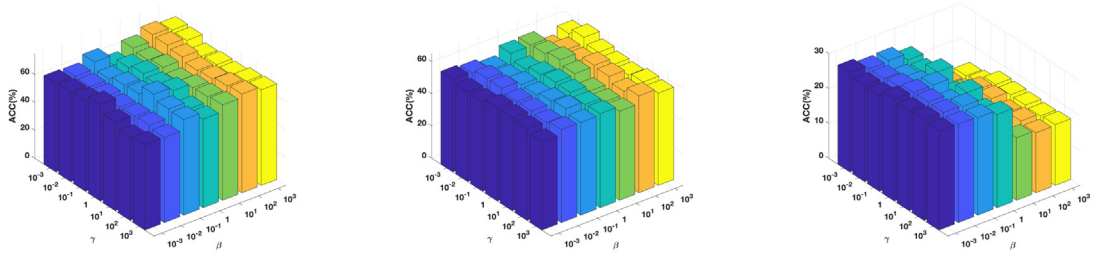
Fig. 7 shows the average results of SDFS and the special cases, i.e., SDFS$_1$ and SDFS$_2$, on all datasets. From this figure, we can observe that the performance of SDFS is higher than SDFS$_1$ and SDFS$_2$ in all datasets. Based on the above analysis, we can conclude that both consensus graph learning and latent representation learning are effective for the proposed SDFS model.

4.7. Convergence analysis

According to the discussion in Section 3.7, the proposed SDFS is a non-convex problem, and an alternating scheme of ALM method is developed to optimize it. In this section, we conduct the experiments to validate the convergence of SDFS, and the curves of convergence are shown in Fig. 8. From this figure, we can observe that the objective curves can converge quickly, which verifies the effectiveness of the developed optimization algorithm.

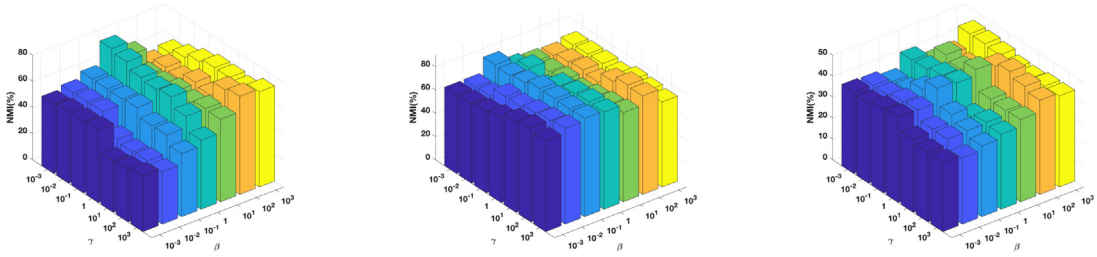


(a) MSRC-v1 (b) Mfeat (c) Caltech101-7

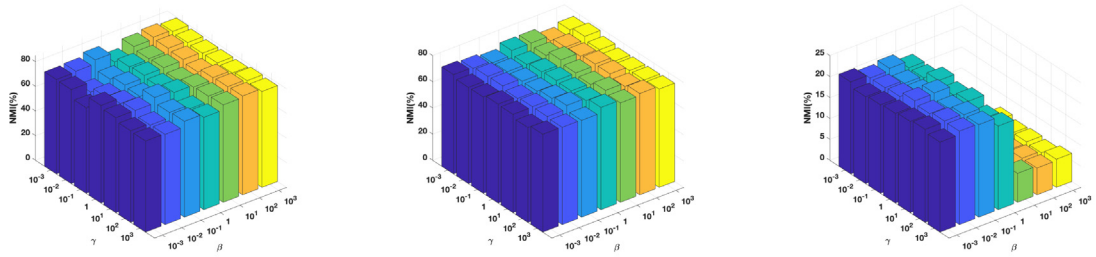


(d) Coil20 (e) ORL (f) Youtube

Fig. 5. ACC results of SDFS with different values of α and β on eight datasets.



(a) MSRC-v1 (b) Mfeat (c) Caltech101-7



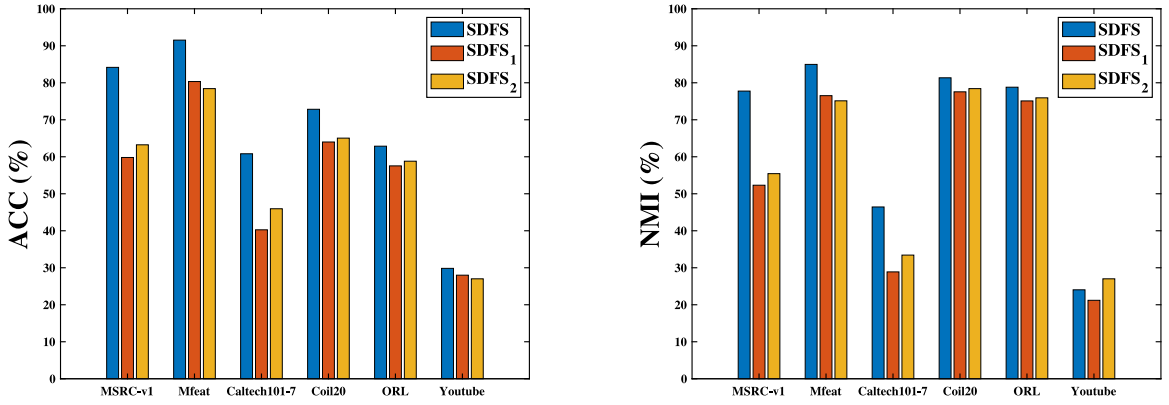
(d) Coil20 (e) ORL (f) Youtube

Fig. 6. NMI results of SDFS with different values of α and β on eight datasets.

5. Conclusion

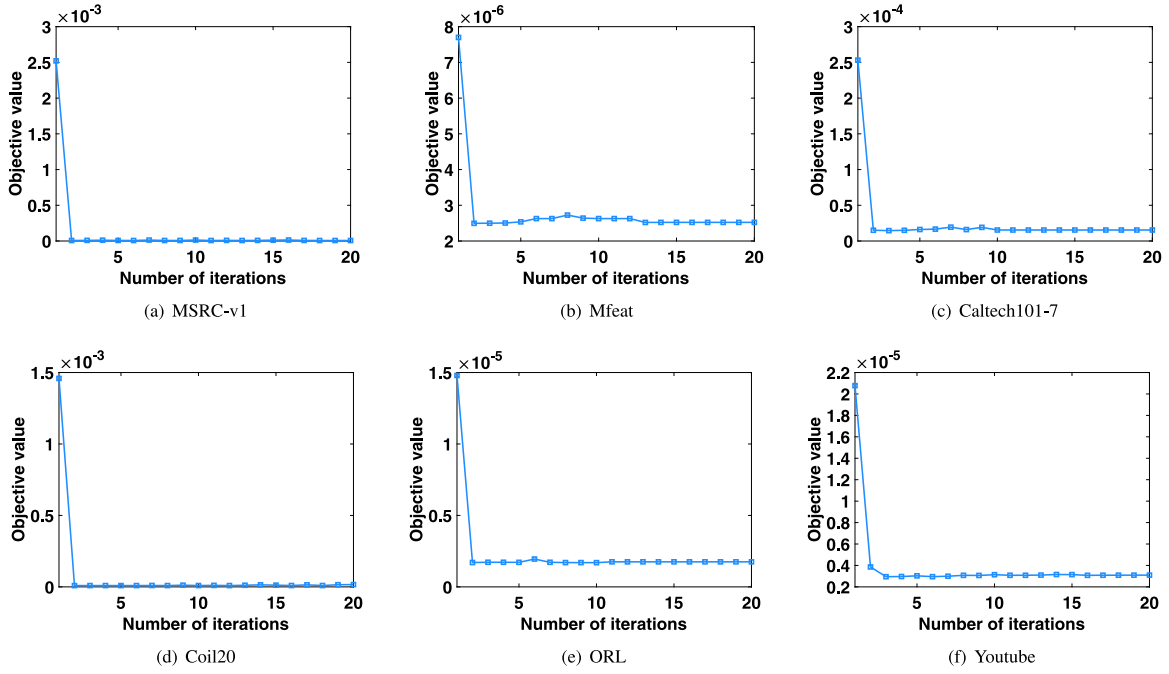
This article presents a novel MUFS method called structural regularization based discriminative multi-view unsupervised feature selection (SDFS). The proposed method can discover the relations between samples by learning the adjacency matrix and latent representation, in which the latent representation matrix is regarded as prior knowledge to guide the feature selection. Further, a novel graph regularization strategy is imposed on the view-specific graphs and the consensus graph to maintain the geometrical structure of data without introducing additional parameters, in which the consensus graph is learned by an automatic weighting strategy. An efficient iterative updating scheme is proffered to optimize the proposed method. Experimental results on six benchmark datasets validate the superiority of SDFS for MUFS tasks.

One limitation of the proposed SDFS lies in that it requires a relatively high complexity for graph construction, which leads to low efficiency when it is applied to large-scale datasets. To tackle this problem, in the future, we would like to develop an anchor graph-based MUFS model to reduce the computational complexity. Another limitation lies in that its optimization requires updating six variables separately. In the future, we would like to investigate designing a new optimizing method that can simultaneously optimize two or more variables.



(a) ACC (b) NMI

Fig. 7. ACC and NMI results of SDFS and two special cases on eight datasets.



(a) MSRC-v1 (b) Mfeat (c) Caltech101-7

(d) Coil20 (e) ORL (f) Youtube

Fig. 8. Convergence curves of SDFS on six datasets.

CRediT authorship contribution statement

Shixuan Zhou: Methodology, Writing – original draft. **Peng Song:** Funding acquisition, Methodology, Supervision, Writing – review & editing. **Yanwei Yu:** Writing – review & editing. **Wenming Zheng:** Writing – review & editing.

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.

Data availability

Data will be made available on request.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China under Grant 61703360, the Natural Science Foundation of Shandong Province, China under Grant ZR2022MF314, and the Graduate Innovation Foundation of Yantai University (GIFYTU).

References

[1] B. Settles, Active learning, Synth. Lect. Artif. Intell. Mach. Learn. 6 (1) (2012) 1–114.
[2] K. Weiss, T.M. Khoshgoftaar, D. Wang, A survey of transfer learning, J. Big Data 3 (1) (2016) 1–40.
[3] Y. Li, M. Yang, Z. Zhang, A survey of multi-view representation learning, IEEE Trans. Knowl. Data Eng. 31 (10) (2018) 1863–1883.
[4] G. Chao, S. Sun, J. Bi, A survey on multi-view clustering, 2017, arXiv preprint arXiv:1712.06246.
[5] R. Zhang, F. Nie, X. Li, X. Wei, Feature selection with multi-view data: A survey, Inf. Fusion 50 (2019) 158–167.

[6] Y. Feng, J. Xiao, Y. Zhuang, X. Liu, Adaptive unsupervised multi-view feature selection for visual concept recognition, in: Asian Conference on Computer Vision, Springer, 2012, pp. 343–357.
[7] W. Yang, Y. Gao, L. Cao, M. Yang, Y. Shi, mPadal: A joint local-and-global multi-view feature selection method for activity recognition, Appl. Intell. 41 (3) (2014) 776–790.
[8] Z. Wang, Y. Feng, T. Qi, X. Yang, J.J. Zhang, Adaptive multi-view feature selection for human motion retrieval, Signal Process. 120 (2016) 691–701.
[9] M. Belkin, P. Niyogi, Laplacian eigenmaps for dimensionality reduction and data representation, Neural Comput. 15 (6) (2003) 1373–1396.
[10] H. Liu, H. Mao, Y. Fu, Robust multi-view feature selection, in: 2016 IEEE 16th International Conference on Data Mining, ICDM, IEEE, 2016, pp. 281–290.
[11] C. Tang, X. Zheng, X. Liu, W. Zhang, J. Zhang, J. Xiong, L. Wang, Cross-view locality preserved diversity and consensus learning for multi-view unsupervised feature selection, IEEE Trans. Knowl. Data Eng. (2021).
[12] S.-G. Fang, D. Huang, C.-D. Wang, Y. Tang, Joint multi-view unsupervised feature selection and graph learning, 2022, arXiv preprint arXiv:2204.08247.
[13] C.M. Bishop, et al., Neural Networks for Pattern Recognition, Oxford University Press, 1995.
[14] X. He, D. Cai, P. Niyogi, Laplacian score for feature selection, Adv. Neural Inf. Process. Syst. 18 (2005).
[15] Z. Zhao, H. Liu, Spectral feature selection for supervised and unsupervised learning, in: Proceedings of the 24th International Conference on Machine Learning, 2007, pp. 1151–1157.
[16] L. Wolf, A. Shashua, D. Geman, Feature selection for unsupervised and supervised inference: The emergence of sparsity in a weight-based approach, J. Mach. Learn. Res. 6 (11) (2005).
[17] H. Zeng, Y.-m. Cheung, Feature selection and kernel learning for local learning-based clustering, IEEE Trans. Pattern Anal. Mach. Intell. 33 (8) (2010) 1532–1547.
[18] L. Jiang, G. Kong, C. Li, Wrapper framework for test-cost-sensitive feature selection, IEEE Trans. Syst. Man Cybern. Syst. 51 (3) (2019) 1747–1756.
[19] D. Cai, C. Zhang, X. He, Unsupervised feature selection for multi-cluster data, in: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2010, pp. 333–342.
[20] Y. Yang, H.T. Shen, Z. Ma, Z. Huang, X. Zhou, L2, 1-norm regularized discriminative feature selection for unsupervised, in: Twenty-Second International Joint Conference on Artificial Intelligence, 2011, pp. 1589–1883.
[21] I. Guyon, A. Elisseeff, An introduction to variable and feature selection, J. Mach. Learn. Res. 3 (Mar) (2003) 1157–1182.
[22] C. Sheng, P. Song, W. Zhang, D. Chen, Dual-graph regularized subspace learning based feature selection, Digit. Signal Process. 117 (2021) 103175.
[23] F. Wang, L. Zhu, J. Li, H. Chen, H. Zhang, Unsupervised soft-label feature selection, Knowl.-Based Syst. 219 (2021) 106847.
[24] W. Li, H. Chen, T. Li, J. Wan, B. Sang, Unsupervised feature selection via self-paced learning and low-redundant regularization, Knowl.-Based Syst. 240 (2022) 108150.
[25] J. Miao, T. Yang, L. Sun, X. Fei, L. Niu, Y. Shi, Graph regularized locally linear embedding for unsupervised feature selection, Pattern Recognit. 122 (2022) 108299.
[26] S. Zhou, P. Song, Z. Song, L. Ji, Soft-label guided non-negative matrix factorization for unsupervised feature selection, Expert Syst. Appl. 216 (2023) 119468.
[27] M. You, A. Yuan, D. He, X. Li, Unsupervised feature selection via neural networks and self-expression with adaptive graph constraint, Pattern Recognit. 135 (2023) 109173.
[28] Q. Lin, L. Yang, P. Zhong, H. Zou, Robust supervised multi-view feature selection with weighted shared loss and maximum margin criterion, Knowl.-Based Syst. 229 (2021) 107331.
[29] K.W. Wangila, K. Gao, P. Zhu, Q. Hu, C. Zhang, Mixed sparsity regularized multi-view unsupervised feature selection, in: 2017 IEEE International Conference on Image Processing, ICIP, IEEE, 2017, pp. 1930–1934.
[30] Q. Lin, M. Men, L. Yang, P. Zhong, A supervised multi-view feature selection method based on locally sparse regularization and block computing, Inform. Sci. 582 (2022) 146–166.
[31] C. Hou, F. Nie, H. Tao, D. Yi, Multi-view unsupervised feature selection with adaptive similarity and view weight, IEEE Trans. Knowl. Data Eng. 29 (9) (2017) 1998–2011.
[32] X. Dong, L. Zhu, X. Song, J. Li, Z. Cheng, Adaptive collaborative similarity learning for unsupervised multi-view feature selection, 2019, arXiv preprint arXiv:1904.11228.
[33] X. Bai, L. Zhu, C. Liang, J. Li, X. Nie, X. Chang, Multi-view feature selection via nonnegative structured graph learning, Neurocomputing 387 (2020) 110–122.
[34] Y. Wan, S. Sun, C. Zeng, Adaptive similarity embedding for unsupervised multi-view feature selection, IEEE Trans. Knowl. Data Eng. 33 (10) (2020) 3338–3350.
[35] C. Tang, J. Chen, X. Liu, M. Li, P. Wang, M. Wang, P. Lu, Consensus learning guided multi-view unsupervised feature selection, Knowl.-Based Syst. 160 (2018) 49–60.
[36] H. Yuan, J. Li, Y. Liang, Y.Y. Tang, Multi-view unsupervised feature selection with tensor low-rank minimization, Neurocomputing 487 (2022) 75–85.
[37] D. Shi, L. Zhu, J. Li, Z. Zhang, X. Chang, Unsupervised adaptive feature selection with binary hashing, IEEE Trans. Image Process. (2023).
[38] R. Shang, L. Wang, F. Shang, L. Jiao, Y. Li, Dual space latent representation learning for unsupervised feature selection, Pattern Recognit. 114 (2021) 107873.
[39] R. Shang, J. Kong, J. Feng, L. Jiao, Feature selection via non-convex constraint and latent representation learning with Laplacian embedding, Expert Syst. Appl. 208 (2022) 118179.
[40] C. Sheng, P. Song, Graph regularized virtual label regression for unsupervised feature selection, Digit. Signal Process. 123 (2022) 103393.
[41] 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.
[42] J. Wright, A.Y. Yang, A. Ganesh, S.S. Sastry, Y. Ma, Robust face recognition via sparse representation, IEEE Trans. Pattern Anal. Mach. Intell. 31 (2) (2008) 210–227.
[43] J. Wen, Z. Zhang, Z. Zhang, L. Fei, M. Wang, Generalized incomplete multiview clustering with flexible locality structure diffusion, IEEE Trans. Cybern. 51 (1) (2020) 101–114.
[44] X. Liu, P. Song, Incomplete multi-view clustering via virtual-label guided matrix factorization, Expert Syst. Appl. 210 (2022) 118408.
[45] D.P. Bertsekas, Constrained Optimization and Lagrange Multiplier Methods, Academic Press, 2014.
[46] R. He, T. Tan, L. Wang, W.-S. Zheng, $\ell_{2,1}$ Regularized correntropy for robust feature selection, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2012, pp. 2504–2511.
[47] D. Cai, X. He, J. Han, Locally consistent concept factorization for document clustering, IEEE Trans. Knowl. Data Eng. 23 (6) (2010) 902–913.
[48] 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.
[49] H. Wang, Y. Yang, B. Liu, GMC: Graph-based multi-view clustering, IEEE Trans. Knowl. Data Eng. 32 (6) (2019) 1116–1129.
[50] J. Winn, N. Jojic, Locus: Learning object classes with unsupervised segmentation, in: Tenth IEEE International Conference on Computer Vision, Vol. 1, ICCV'05 Volume 1, IEEE, 2005, pp. 756–763.
[51] K.-Y. Lin, L. Huang, C.-D. Wang, H.-Y. Chao, Multi-view proximity learning for clustering, in: International Conference on Database Systems for Advanced Applications, Springer, 2018, pp. 407–423.
[52] L. Fei-Fei, R. Fergus, P. Perona, Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories, in: 2004 Conference on Computer Vision and Pattern Recognition Workshop, IEEE, 2004, p. 178.
[53] S.A. Nene, S.K. Nayar, H. Murase, et al., Columbia object image library (coil-100), Technical Report, Citeseer, 1996.
[54] H. Zhang, D. Wu, F. Nie, R. Wang, X. Li, Multilevel projections with adaptive neighbor graph for unsupervised multi-view feature selection, Inf. Fusion 70 (2021) 129–140.
[55] J. Liu, Y. Yang, M. Shah, Learning semantic visual vocabularies using diffusion distance, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2009, pp. 461–468.
[56] J.C. Niebles, H. Wang, L. Fei-Fei, Unsupervised learning of human action categories using spatial-temporal words, Int. J. Comput. Vis. 79 (3) (2008) 299–318.
[57] H.-J. Lee, E.-J. Jeong, H. Kim, M. Czosnyka, D.-J. Kim, Morphological feature extraction from a continuous intracranial pressure pulse via a peak clustering algorithm, IEEE Trans. Biomed. Eng. 63 (10) (2015) 2169–2176.
[58] N. Liang, Z. Yang, L. Li, Z. Li, S. Xie, Incomplete multi-view clustering with cross-view feature transformation, IEEE Trans. Artif. Intell. (2021).
[59] C.H. Papadimitriou, K. Steiglitz, Combinatorial Optimization: Algorithms and Complexity, Courier Corporation, 1998.
[60] A. Strehl, J. Ghosh, Cluster ensembles—a knowledge reuse framework for combining multiple partitions, J. Mach. Learn. Res. 3 (Dec) (2002) 583–617.