Information Sciences 649 (2023) 119675

Contents lists available at ScienceDirect

Information Sciences

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

Efficient multi-view semi-supervised feature selection

Chenglong Zhang [a], Bingbing Jiang [a,*], Zidong Wang [b], Jie Yang [c], Yangfeng Lu [a], Xingyu Wu [a], Weiguo Sheng [a,*]

[a] *School of Information Science and Technology, Hangzhou Normal University, Hangzhou 311121, China*
[b] *Department of Computer Science, Brunel University London, Uxbridge, Middlesex UB8 3PH, UK*
[c] *Australian AI Institute, University of Technology Sydney, 15 Broadway Ultimo, NSW 2007, Australia*

ARTICLE INFO

Keywords:
Multi-view learning
Semi-supervised feature selection
Sparse feature projection
Bipartite graph learning

ABSTRACT

Multi-view semi-supervised feature selection can identify a feature subset from heterogeneous feature spaces of data. However, existing methods fail in handling large-scale data since they have to calculate the inverses of high-order dense matrices. Moreover, traditional methods often preconstruct graphs to mine the similarity structure of data, such that the interaction between graph construction and feature selection is directly ignored, degrading their effectiveness in practice. To address these issues, we propose an efficient multi-view feature selection method (EMSFS), which combines graph learning, label propagation as well as multi-view feature selection within a unified framework. Specifically, EMSFS can adaptively learn a bipartite graph between training samples and generated anchors, not only reducing the cost of graph computation but also tactfully avoiding the inverse of a high-order matrix. As a result, the main computational complexity of EMSFS is approximately linear to the number of training samples. Meanwhile, EMSFS simultaneously selects important features and exploits the similarity structure in the projected feature space, which enhances the reliability of the graph and positively facilitates feature selection. To solve the formulated objective function, we developed an alternating optimization, and experiments validate the effectiveness and the efficiency of EMSFS.

1. Introduction

As information acquisition technology continues to develop, multi-view data with heterogeneous feature representations have become increasingly available in many domains [1,2]. Unlike single-view data, multi-view data can describe research objects from diverse perspectives, as the features from different views have partly independent statistical properties [3–5]. However, due to the instability of external environments and collection equipment, multi-view data collected from practical applications are often high-dimensional and typically contain irrelevant features and noisy dimensions [6]. Thus, directly handling such data not only encounters substantial computation and storage costs but also compromises the performance of subsequent tasks [7,8]. As a dimensionality reduction technique, feature selection can eliminate low-quality features from data without altering the original feature space, enabling sound interpretability of high-dimensional data. Consequently, the multi-view feature selection that can identify a discriminative feature subset from heterogeneous feature spaces, has received considerable attention in recent years [9].

[*] Corresponding authors.
E-mail addresses: jiangbb@hznu.edu.cn (B. Jiang), zidong.wang@brunel.ac.uk (Z. Wang), jie.yang-1@uts.edu.au (J. Yang), w.sheng@ieee.org (W. Sheng).

https://doi.org/10.1016/j.ins.2023.119675
Received 22 May 2023; Received in revised form 17 July 2023; Accepted 4 September 2023

Available online 11 September 2023

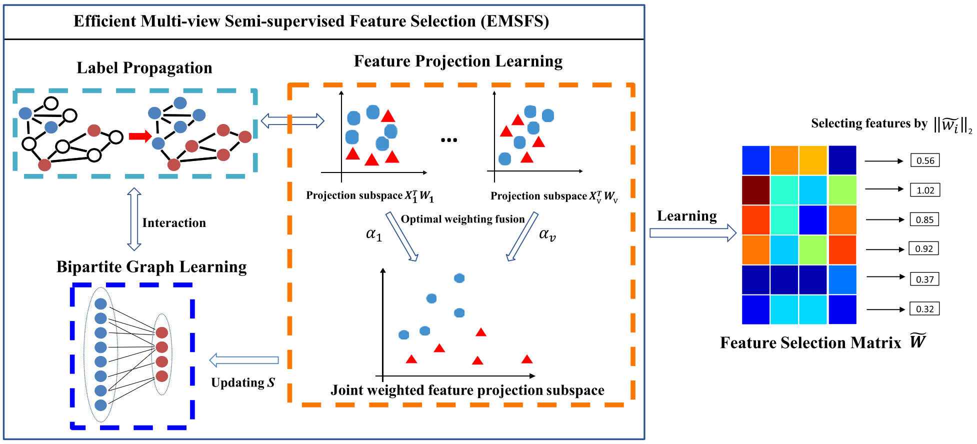

Efficient Multi-view Semi-supervised Feature Selection (EMSFS)
Label Propagation
Feature Projection Learning
Projection subspace $X_1^T W_1$
Projection subspace $X_v^T W_v$
Optimal weighting fusion
α_1
α_v
Interaction
Bipartite Graph Learning
Updating S
Joint weighted feature projection subspace
Learning
Selecting features by $\|\widetilde{w_i}\|_2$
0.56
1.02
0.85
0.92
0.37
0.32
Feature Selection Matrix $\widetilde{W}$

Fig. 1. Schematic illustration of the proposed multi-view semi-supervised feature selection.

According to the availability of label information for training samples, multi-view feature selection can be achieved through supervised, unsupervised, and semi-supervised manners [10–14]. Due to the expensive and time-consuming process for labeling data, there are a limited number of labeled samples in real-world applications [15,16]. As a result, researchers have made efforts on multi-view semi-supervised feature selection. Existing approaches to obtain a feature subset from multi-view data can be classified into two types. The first way directly concatenates different views and then invokes single-view feature selection models, in which the similarity structure of data and sparse constraints are often considered. Typical methods mostly exploit the graph-based label propagation and sparse regression to learn projection matrices of concatenated features [17–20]. Unfortunately, these methods equally treat different views and overlook the distinctions among diverse views, severely restricting their effectiveness and applicability in multi-view scenarios [10,21].

To improve the multi-view feature selection, another strategy introduces a weight for each view to discriminate different views. Representative methods first construct graphs on each view independently, and then perform feature selection guided by the label propagation on different views [22–24]. Nevertheless, they treat the graph construction and feature selection as two separate processes and ignore the information interaction between them, which may affect the reliability of graphs and the effectiveness of selected features. As a result, several methods have been developed to dynamically update similarity graphs during feature selection. For example, the method in [25] dynamically updates the similarity structure in the original feature space of data. The original space of data, however, usually contains low-quality features that might make the learned similarity relations unreliable. To alleviate this problem, Jiang et al. proposed to learn a joint graph for different views according to the similarity structure in projected feature space [26]. Despite making some achievements, existing methods suffer from high computation costs that come from: i) the solution procedure of most methods involves the inverse operations of $n \times n$ dense matrices, taking the computational complexity of $\mathcal{O}(n^3)$; ii) they construct n-order graphs to explore the similarity structure of data, taking $\mathcal{O}(n^2 d)$ (where n and d are the numbers of samples and features, respectively). Therefore, it is difficult to apply these methods to large-scale multi-view semi-supervised feature selection tasks, extensively degrading the computational efficiency and applicability in practice. To the best of our knowledge, relatively few efforts have been made on accelerating the semi-supervised multi-view feature selection.

To address the challenges mentioned above, we propose an efficient multi-view semi-supervised feature selection (EMSFS) method that combines bipartite graph learning with semi-supervised feature selection. Departing from previous works, EMSFS can adaptively learn a bipartite graph between training samples and generated anchors, not only improving the computational efficiency of graph construction but also tactfully replacing the inverse of a high-order matrix with the simple operations on low-order matrices. As a result, the computational complexity of EMSFS is approximately linear to the number of training samples, such that EMSFS can efficiently deal with large-scale multi-view data. Moreover, EMSFS incorporates the graph construction, label propagation as well as multi-view feature selection into a unified framework, such that it selects discriminative features and simultaneously exploits the neighbor relations in the projected space to construct a unified graph for different views, which enhances the quality of the graph and positively facilitates the ultimate feature selection. The schematic illustration of EMSFS is shown in Fig. 1.

2. Notations and related works

2.1. Notations

In this section, we first introduce the notations frequently used in the paper. Specifically, matrices and vectors can be written in boldface with uppercase and lowercase, respectively. Furthermore, for a matrix $\mathbf{M}$, $\mathbf{m}_i$ is the i-th row of $\mathbf{M}$, $\|\mathbf{M}\|_F = \sqrt{\mathrm{Tr}(\mathbf{M}^T\mathbf{M})}$ denotes the matrix Frobenius norm, and $\|\mathbf{M}\|_{2,1} = \sum_{i=1}^{n} \|\mathbf{m}_i\|_2$ denotes the $l_{2,1}$-norm of $\mathbf{M}$, where $\|\mathbf{m}_i\|_2 = \sqrt{\mathbf{m}_i\mathbf{m}_i^T}$ is the l_2-norm of the row vector $\mathbf{m}_i$. Table 1 lists the notations throughout the paper.

Table 1
Table of Notations.

Notation	Description
T	The overall number of samples.
n	The total number of training data
l	The number of labeled data
c	The number of classes
V	The number of views
d_v	The dimension of v-th view
$d = \sum_{v=1}^{V} d_v$	The total dimension of V views
$\mathbf{x}_i^v \in \mathbb{R}^{d_v \times 1}$	The i-th sample in v-th view
$\mathbf{x}_i = [\mathbf{x}_i^1, ..., \mathbf{x}_i^V] \in \mathbb{R}^{d \times 1}$	The i-th sample
$\mathbf{X} = [\mathbf{x}_1, ..., \mathbf{x}_n] \in \mathbb{R}^{d \times n}$	The concatenated feature matrix
$\mathbf{Z} = [\mathbf{z}_1, ..., \mathbf{z}_m] \in \mathbb{R}^{d \times m}$	The generated anchors
$\mathbf{Y}_l \in \mathbb{R}^{l \times c}$	The label matrix of labeled samples
$\mathbf{Y}_n = [\mathbf{Y}_l; \mathbf{0}] \in \mathbb{R}^{n \times c}$	The label matrix
$\mathbf{L} \in \mathbb{R}^{n \times n}$	The graph Laplacian matrix
$\mathbf{F} \in \mathbb{R}^{n \times c}$	The predicted label matrix
$\mathbf{W} \in \mathbb{R}^{d \times c}$	The feature projection matrix
$\mathbf{b} \in \mathbb{R}^{c \times 1}$	The bias vector
$\mathbf{1} \in \mathbb{R}^{c \times 1}$	The all-one vector.

2.2. Single-view and multi-view semi-supervised feature selection

During recent years, sparse projection learning has become a popular model for feature selection [27,28], whose general formulation can be written as:

$$\min_{\mathbf{W}} \; Loss(\mathbf{W}^T\mathbf{X}, \mathbf{F}) + \gamma R(\mathbf{W}), \tag{1}$$

where the first term measures the discrepancy between the projection subspace $\mathbf{W}^T\mathbf{X}$ and the predicted label $\mathbf{F}$. To ensure that the feature projection $\mathbf{W}$ is row-sparse, $R(\mathbf{W})$ is often materialized as a sparse regularization, making $\mathbf{W}$ serve as a feature selection matrix [13]. Accordingly, the importance of features can be evaluated by the $\|\mathbf{w}_i\|_2$, where $\mathbf{w}_i$ denotes the i-th row of the feature projection $\mathbf{W}$. To utilize unlabeled data, Ma et al. [17] incorporated the graph-based label propagation [10] into the sparse projection learning, and proposed a structural feature selection model, formulated as:

$$\begin{aligned} &\min_{\mathbf{F},\mathbf{W},\mathbf{b}} \operatorname{Tr}(\mathbf{F}^T\mathbf{L}\mathbf{F}) + \operatorname{Tr}((\mathbf{F}-\mathbf{Y}_n)^T\mathbf{U}_n(\mathbf{F}-\mathbf{Y}_n)) \\ &+ \mu\|\mathbf{X}^T\mathbf{W} + \mathbf{1}\mathbf{b}^T - \mathbf{F}\|_F^2 + \gamma\|\mathbf{W}\|_{2,1}, \end{aligned} \tag{2}$$

where μ and γ are regularization parameters. $\mathbf{U}_n \in \mathbb{R}^{n \times n}$ is a diagonal matrix, in which the i-th diagonal element will be a large const if $\mathbf{x}_i$ is labeled and 1 otherwise. In Eq. (2), the first term propagates the label information from labeled samples to unlabeled samples, and the second term makes the prediction labels on labeled samples consistent with the given labels. Based on Eq. (2), many forms of semi-supervised feature selection models have been developed [16,18,29]. For example, Shi et al. imposed a binary hash constraint on the predicted labels and proposed a binary label learning strategy for feature selection tasks [18,30]. Recently, Zhang et al. proposed a semi-supervised feature selection method with soft label learning (FSSLL) [19], which employs fuzzy CMeans clustering to construct the initial soft label matrix of data. The optimization objective of FSSLL is formulated as:

$$\begin{aligned} &\min_{\mathbf{W},\mathbf{F},\mathbf{V}} \|\mathbf{F}-\mathbf{U}\|_F^2 + \alpha(\|\mathbf{F}_l\mathbf{V} - \mathbf{Y}_l\|_F^2 + \gamma\|\mathbf{V}\|_F^2) \\ &+ \beta(\sum_{i=1}^{n} \|\mathbf{W}^T\mathbf{x}_i - \mathbf{f}_i\|_2^2 + \lambda\|\mathbf{W}\|_{2,1}) \text{ s.t. } \mathbf{F}\mathbf{1} = \mathbf{1}, \mathbf{F} \geqslant 0, \end{aligned} \tag{3}$$

where $\mathbf{U}$ and $\mathbf{V}$ denote the pre-learned soft labels of training data and the projection matrix of labeled samples, respectively. To reduce computation costs and avoid similarity graphs, FSSLL directly neglects the local neighbor relationships of training data, which is very essential for feature selection especially when a few samples are labeled [31].

Moreover, the methods mentioned above directly concatenate different views and neglect the differences among views. To balance different views, Shi et al. designed the multi-view Laplacian sparse feature selection (MLSFS), which combines the label propagation on different views by introducing view weights [22]. Similarly, Li et al. proposed to fuse similarity graphs across views during the process of feature selection [23]. In [24], the Hessian matrices derived from different views are adopted to replace the Laplacian matrices of MLSFS. However, the performance of these methods extremely depends on the pre-constructed similarity graphs, neglecting the interaction between feature selection and graph learning. To alleviate this problem, the multi-view adaptive semi-supervised feature selection (MASFS) [25] was proposed to update the similarity structures based on the data distance in the original space and the current predicted labels, whose objective function is:

$$\min_{\mathbf{F},\mathbf{W},\boldsymbol{\alpha}} \operatorname{Tr}(\mathbf{F}^T\mathbf{L}\mathbf{F}) + \operatorname{Tr}\Big((\mathbf{F}-\mathbf{Y}_n)^T\mathbf{U}_n(\mathbf{F}-\mathbf{Y}_n)\Big) + \mu\|\mathbf{X}^T\mathbf{W} - \mathbf{F}\|_F^2 + \lambda\operatorname{Tr}(\mathbf{X}^T\mathbf{L}\mathbf{X})$$

$$+\beta\|\mathbf{W}\|_{2,\frac{1}{2}}^{\frac{1}{2}}+f(\mathbf{S}^v;\sigma)\quad \text{s.t. } \mathbf{L}=\sum_{v=1}^{V}\alpha_v^{\gamma}\mathbf{L}^v, \boldsymbol{\alpha}^T\mathbf{1}=1,\ \alpha_v\in[0,1], \tag{4}$$

where α_v and $\mathbf{L}^v$ are the view weight and the Laplacian matrix of the v-th view, respectively. The exponential parameter $\gamma > 1$ controls the distribution of $\{\alpha_v\}_{v=1}^{V}$. The self-paced function $f(\mathbf{S}^v;\sigma)$ enables MASFS to update the single-view graph $\mathbf{S}^v$, in which σ is the self-paced learning parameter [32]. Recently, Jiang et al. proposed a multi-view semi-supervised feature selection model (SMFS) that mines the similarity structures of data from the original space and the projected feature space, whose graph learning model is formulated as:

$$\sum_{v=1}^{V} q_v\|\mathbf{S}-\mathbf{S}^v\|_F^2+\mu\sum_{i,j=1}^{n}s_{ij}\|\widehat{\mathbf{W}}^T\mathbf{x}_i-\widehat{\mathbf{W}}^T\mathbf{x}_j\|_2^2\quad \text{s.t. } \mathbf{S1}=\mathbf{1},\mathbf{S}\geqslant 0, \tag{5}$$

where $q_v=\frac{1}{2\|\mathbf{S}-\mathbf{S}^v\|_F}$, the single-view graphs $\{\mathbf{S}^v\}_{v=1}^{V}$ are constructed from the original feature space. $\widehat{\mathbf{W}}\in\mathbb{R}^{d\times c}$ denotes the joint feature projection of multiple views. In SMFS, although the undesirable effects of noisy features can be weakened, it takes $\mathcal{O}(n^2d)$ to update the $n\times n$ graph $\mathbf{S}$ in each iteration. Besides, SMFS has to calculate the inverse of an $n\times n$ dense matrix in solving $\mathbf{F}$, which costs $\mathcal{O}(n^3)$, making it impractical for large-scale problems.

2.3. A brief on graph learning

Graph-based semi-supervised learning/feature selection has attracted considerable attention in recent years [31]. Most methods typically adopt a two-step strategy of constructing graphs and propagating the label information from labeled samples to unlabeled samples, which aim to solve the following minimization problem:

$$\min_{\mathbf{F}}\sum_{i,j=1}^{n}\|\mathbf{f}_i-\mathbf{f}_j\|_2^2 s_{ij}+\mathrm{Tr}\left((\mathbf{F}-\mathbf{Y})^T\mathbf{U}(\mathbf{F}-\mathbf{Y})\right), \tag{6}$$

where s_{ij} denotes the similarity between the i-th and j-th samples. It can be observed that the quality of constructed graphs has a direct influence on the performance of graph-based methods. Previous graph construction models use the kernel-based strategy (e.g., Gaussian kernel $s_{ij}=\frac{exp(-\|\mathbf{x_i}-\mathbf{x}_j\|_2^2)}{2\sigma^2}$) to build graphs [17]. However, this manner needs to manually tune the kernel parameter (e.g., the σ in the Gaussian kernel), degrading the applicability of models in practice. Apart from the kernel parameter involved in constructing graphs, these methods learn the similarity structure from all samples, hindering the effective utilization of neighbor information [33]. To address these issues, a graph construction model that learns similarity graph based on the distance in the original feature space was proposed in [34], as follows

$$\min_{\mathbf{S1}=\mathbf{1},\mathbf{S}\geqslant 0}\sum_{i=1}^{n}\sum_{j=1}^{n}s_{ij}\|\mathbf{x}_i-\mathbf{x}_j\|_2^2+\eta\|\mathbf{S}\|_F^2. \tag{7}$$

Compared with the kernel-based strategy, the above model learns the similarity information from the k-nearest neighbor for each sample. Nevertheless, the construction of an n-order graph requires the computational complexity of $\mathcal{O}(n^2d)$, making it impractical for large-scale problems.

Recently, several researchers have devoted to learning a bipartite graph rather than the n-order graph to reduce the computational costs of graph construction. For example, He et al. proposed to learn the similarity between the samples and anchor points [35], whose objective function is formulated as:

$$\min_{\mathbf{S1}=\mathbf{1},\mathbf{S}\geqslant 0}\sum_{i=1}^{n}\sum_{j=1}^{m}s_{ij}\|\mathbf{x}_i-\mathbf{z}_j\|_2^2+\eta\|\mathbf{S}\|_F^2. \tag{8}$$

Based on [35], many variants of bipartite graph learning models have been developed. In [36], the distance in the original feature space and the current prediction label information were utilized to update the bipartite graph. In [37], the graphs derived from different views are fused to learn a unified bipartite graph for multi-view semi-supervised scenarios. Despite making some progress, the performance of the above methods is still unsatisfying. One of the limitations is that the graphs are directly constructed from the original feature space and remain constant in the learning procedures for most methods. As a result, the graphs might be susceptible to low-quality features since the original feature space usually contains irrelative features and noisy dimensions, impairing the reliability of the graph, To capture the intrinsic neighbor relations of data accurately, the sample similarity structure in the projected feature space should be taken into consideration.

3. Proposed method

3.1. Adaptive bipartite graph learning

Most existing methods suffer from high computational costs issue due to the use of all samples for previously constructing similarity graphs [37,38]. Thus, we plan to use k-means to obtain m centers of data as anchors $\mathbf{z}_j=[\mathbf{z}_j^1,...,\mathbf{z}_j^V]\in\mathbb{R}^{d\times 1}$, and then learn an $n\times m$ bipartite graph $\mathbf{S}$ that compatibly crosses multiple views. Meanwhile, feature selection and bipartite graph learning are performed simultaneously to fully utilize the interaction information between them, which can effectively alleviate the adverse effects of noisy features and positively facilitate the subsequent tasks. To this end, two aspects should be considered for bipartite graph construction: i) the similarity relations derived from the original feature space are easily affected by redundant and noisy features, leading to an inaccurate similarity graph; ii) different views contain complementary information and share a consistent local structure. According to the principle that the smaller distance between samples and anchors in the projection subspace, the larger similarity between them, we propose the feature projection-based adaptive bipartite graph learning, formulated as:

$$\min_{\mathbf{S1}=\mathbf{1},\mathbf{S}\geqslant 0}\sum_{i=1}^{n}\sum_{j=1}^{m}s_{ij}\|\sum_{v=1}^{V}\alpha_v\mathbf{W}_v^T\mathbf{x}_i^v-\sum_{v=1}^{V}\alpha_v\mathbf{W}_v^T\mathbf{z}_j^v\|_2^2+\eta\|\mathbf{S}\|_F^2, \tag{9}$$

where s_{ij} measures the similarity between sample $\mathbf{x}_i$ and anchor $\mathbf{z}_j$ across all of the views, and $\eta > 0$ is the regularization parameter. The view-specific feature projection $\mathbf{W}_v \in \mathbb{R}^{d_v\times c}$ can map the original features $\mathbf{X}_v \in \mathbb{R}^{d_v\times n}$ into the corresponding subspace, and α_v is the view weight factor that can discriminate different feature projection subspaces. Thus, Eq. (9) can mine and balance the similarity structures among multiple views. By introducing the feature projection $\mathbf{W}_v$, the similarity structure in the weighted projection subspace (i.e., $\alpha_v\mathbf{X}_v^T\mathbf{W}_v$) can be exploited to learn a unified bipartite graph $\mathbf{S}$ for multiple views. In this way, bipartite graph learning and feature selection can benefit from each other in a mutual reinforcement manner. Denoting the $\mathbf{u}_i$ is a row vector with $u_{ij} = \|\sum_{v=1}^{V}\alpha_v\mathbf{W}_v^T\mathbf{x}_i^v-\sum_{v=1}^{V}\alpha_v\mathbf{W}_v^T\mathbf{z}_j^v\|_2^2$ sorted from small to large, and assuming that each sample has k nearest neighbors (i.e., $\mathbf{s}_i$ has k nonzero elements), the parameter η can be determined adaptively as: $\eta=\sum_{i=1}^{n}\frac{ku_{i,k+1}-\sum_{j=1}^{k}u_{i,j}}{2n}$, and the solution of Eq. (9) is derived as:

$$s_{ij}=(\frac{u_{i,k+1}-u_{i,j}}{ku_{i,k+1}-\sum_{j=1}^{k}u_{ij}})_+. \tag{10}$$

Thus, this manner not only enhances the reliability of the learned graph (i.e., the similarity relation learns from k nearest projected neighbors are more accurate [39]) but also releases the model from an extra parameter.

In Eq. (9), the anchor points can be considered as the newly unlabeled samples, and its corresponding prediction label matrix is denoted as $\mathbf{G} \in \mathbb{R}^{m\times c}$. To propagate the label information from labeled samples to unlabeled samples (including the anchors $\mathbf{Z}$) according to the similarity relations of data, we can define an augmented matrix $\widetilde{\mathbf{S}} = \begin{bmatrix}\mathbf{0} & \mathbf{S}\\ \mathbf{S}^T & \mathbf{0}\end{bmatrix} \in \mathbb{R}^{(n+m)\times(n+m)}$. With the augmented matrix of the bipartite graph $\mathbf{S}$, the label propagation can be formulated as:

$$\mathrm{Tr}\left(\begin{bmatrix}\mathbf{F}\\ \mathbf{G}\end{bmatrix}^T\mathbf{L}_{\widetilde{\mathbf{S}}}\begin{bmatrix}\mathbf{F}\\ \mathbf{G}\end{bmatrix}\right)+\mathrm{Tr}\left(\begin{bmatrix}\mathbf{F}-\mathbf{Y}_n\\ \mathbf{G}\end{bmatrix}^T\mathbf{U}\begin{bmatrix}\mathbf{F}-\mathbf{Y}_n\\ \mathbf{G}\end{bmatrix}\right), \tag{11}$$

where $\mathbf{U} = \begin{bmatrix}\mathbf{U}_n & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix} \in \mathbb{R}^{(n+m)\times(n+m)}$, and $\mathbf{L}_{\widetilde{\mathbf{S}}}$ denotes the Laplacian matrix of $\widetilde{\mathbf{S}}$, which is calculated as follows:

$$\mathbf{L}_{\widetilde{\mathbf{S}}}=\begin{bmatrix}\mathbf{D}_s & \mathbf{0}\\ \mathbf{0} & \mathbf{\Lambda}\end{bmatrix}-\begin{bmatrix}\mathbf{0} & \mathbf{S}\\ \mathbf{S}^T & \mathbf{0}\end{bmatrix}=\begin{bmatrix}\mathbf{I}_n & -\mathbf{S}\\ -\mathbf{S}^T & \mathbf{\Lambda}\end{bmatrix}, \tag{12}$$

where $\mathbf{D}_s$ and $\mathbf{\Lambda}$ are diagonal matrices whose elements are row sums and column sums of the bipartite graph $\mathbf{S}$, respectively.

3.2. EMSFS with bipartite graph

We propose an efficient multi-view semi-supervised feature selection (EMSFS) that combines bipartite graph learning, label propagation, and multi-view feature selection within a unified learning framework. To increase the robustness of feature projection learning against outliers, the $l_{2,1}$-norm constraint is imposed on the regression loss. Therefore, the objective function of EMSFS is formulated as follows:

$$\begin{aligned}&\min_{\mathbf{W}_v,\mathbf{R},\mathbf{S},\boldsymbol{\alpha}}\|\mathbf{F}-\sum_{v=1}^{V}\alpha_v\mathbf{X}_v^T\mathbf{W}_v\|_{2,1}+\lambda\sum_{v=1}^{V}\|\mathbf{W}_v\|_{2,1}+\gamma\mathrm{Tr}(\mathbf{R}^T\mathbf{L}_{\widetilde{\mathbf{S}}}\mathbf{R})\\ &+\mathrm{Tr}\big((\mathbf{R}-\mathbf{Y})^T\mathbf{U}(\mathbf{R}-\mathbf{Y})\big)+\beta\Big(\sum_{i=1}^{n}\sum_{j=1}^{m}s_{ij}\|\sum_{v=1}^{V}\alpha_v\mathbf{W}_v^T\mathbf{x}_i^v-\sum_{v=1}^{V}\alpha_v\mathbf{W}_v^T\mathbf{z}_j^v\|_2^2+\eta\|\mathbf{S}\|_F^2\Big)\\ &\text{s.t. } \boldsymbol{\alpha}\geqslant 0, \boldsymbol{\alpha}^T\mathbf{1}=1,\ \mathbf{S1}=\mathbf{1},\mathbf{S}\geqslant 0,\end{aligned} \tag{13}$$

where $\mathbf{R} = \begin{bmatrix}\mathbf{F}\\ \mathbf{G}\end{bmatrix}$, $\mathbf{Y} = \begin{bmatrix}\mathbf{Y}_n\\ \mathbf{0}\end{bmatrix} \in \mathbb{R}^{(n+m)\times c}$. To select informative and discriminative features in the case of relatively limited labeled samples, EMSFS uses label propagation to embed the multi-view sparse projection learning into the bipartite graph construction, making them benefit from each other in the learning phase. Specifically, the graph structure among training samples can be better constructed in the feature projection subspace, such that the label information of training samples will be enriched via the label propagation on the learned bipartite graph. In this way, sparse projection learning can learn a discriminative feature selection/projection matrix guided by the prediction label $\mathbf{F}$.

However, the coupled relationship between $\mathbf{F}$ and the projection subspaces in the $l_{2,1}$-norm regression loss makes the original optimization problem in Eq. (13) difficult to be optimized directly. To make Eq. (13) separable, an auxiliary variable $\mathbf{E}$ is introduced, which can transform the original optimization problem into the following equivalent problem:

$$\begin{aligned}
&\min_{\mathbf{W}_v,\mathbf{R},\mathbf{S},\boldsymbol{\alpha},\mathbf{E}} \|\mathbf{E}\|_{2,1} + \lambda \sum_{v=1}^{V} \|\mathbf{W}_v\|_{2,1} + \gamma \mathrm{Tr}(\mathbf{R}^T \mathbf{L}_{\widetilde{\mathbf{S}}} \mathbf{R}) + \mathrm{Tr}\big((\mathbf{R}-\mathbf{Y})^T \mathbf{U}(\mathbf{R}-\mathbf{Y})\big) \\
&+ \beta\Big(\sum_{i=1}^{n}\sum_{j=1}^{m} s_{ij} \|\sum_{v=1}^{V} \alpha_v \mathbf{W}_v^T \mathbf{x}_i^v - \sum_{v=1}^{V} \alpha_v \mathbf{W}_v^T \mathbf{z}_j^v\|_2^2 + \eta \|\mathbf{S}\|_F^2\Big) \\
&\text{s.t. } \boldsymbol{\alpha} \geqslant 0, \boldsymbol{\alpha}^T \mathbf{1} = 1,\ \mathbf{S1} = \mathbf{1}, \mathbf{S} \geqslant 0, \mathbf{E} = \mathbf{F} - \sum_{v=1}^{V} \alpha_v \mathbf{X}_v^T \mathbf{W}_v.
\end{aligned} \tag{14}$$

Eq. (14) can be efficiently solved by the augmented Lagrangian multiplier (ALM) method [40]. Thus, the original optimization problem in Eq. (13) is transformed into the following ALM problem:

$$\begin{aligned}
&\min_{\mathbf{W}_v,\mathbf{R},\mathbf{S},\boldsymbol{\alpha},\mathbf{E}} \|\mathbf{E}\|_{2,1} + \lambda \sum_{v=1}^{V} \|\mathbf{W}_v\|_{2,1} + \gamma \mathrm{Tr}(\mathbf{R}^T \mathbf{L}_{\widetilde{\mathbf{S}}} \mathbf{R}) + \mathrm{Tr}\big((\mathbf{R}-\mathbf{Y})^T \mathbf{U}(\mathbf{R}-\mathbf{Y})\big) \\
&+ \beta\Big(\sum_{i=1}^{n}\sum_{j=1}^{m} s_{ij} \|\sum_{v=1}^{V} \alpha_v \mathbf{W}_v^T \mathbf{x}_i^v - \sum_{v=1}^{V} \alpha_v \mathbf{W}_v^T \mathbf{z}_j^v\|_2^2 + \eta \|\mathbf{S}\|_F^2\Big) + \frac{\mu}{2} \|\mathbf{E} - \mathbf{F} + \sum_{v=1}^{V} \alpha_v \mathbf{X}_v^T \mathbf{W}_v + \frac{\mathbf{\Pi}}{\mu}\|_F^2 \\
&\text{s.t. } \boldsymbol{\alpha} \geqslant 0, \boldsymbol{\alpha}^T \mathbf{1} = 1,\ \mathbf{S1} = \mathbf{1}, \mathbf{S} \geqslant 0,
\end{aligned} \tag{15}$$

where $\mu \in \mathbb{R}^{1\times 1}$ is a penalty parameter, and $\mathbf{\Pi} \in \mathbb{R}^{n\times c}$ denotes the Lagrange multipliers. To obtain the solutions of all variables, we design an optimization strategy that alternately solves each variable of Eq. (15) with others fixed. The detailed solution procedures are as follows.

Update E: By fixing other variables, we can update $\mathbf{E}$ by addressing the following subproblem:

$$\min_{\mathbf{E}} \frac{1}{\mu} \|\mathbf{E}\|_{2,1} + \frac{1}{2} \|\mathbf{E} - \mathbf{J}\|_F^2, \tag{16}$$

where $\mathbf{J} = \mathbf{F} - \sum_{v=1}^{V} \alpha_v \mathbf{X}_v^T \mathbf{W}_v - \frac{\mathbf{\Pi}}{\mu}$. According to [41], the optimal solution of $\mathbf{E}$ is:

$$\mathbf{e}^i = \begin{cases} (1 - \frac{1}{\mu \|\mathbf{j}^i\|_2})\mathbf{j}^i, & \text{if } \|\mathbf{j}^i\|_2 > \frac{1}{\mu}; \\ 0, & \text{otherwise}, \end{cases} \tag{17}$$

where $\mathbf{e}^i$ and $\mathbf{j}^i$ are the i-th columns of $\mathbf{E}$ and $\mathbf{J}$, respectively.

Update F, G and $\mathbf{W}_v$: When other variables are fixed except $\mathbf{F}$, $\mathbf{G}$ and $\mathbf{W}_v$, the view-wise weight α_v can be merged into the corresponding feature projection $\mathbf{W}_v$ as $\alpha_v \mathbf{W}_v = \widetilde{\mathbf{W}}_v$, where $\widetilde{\mathbf{W}}_v$ plays the role of a weighted feature projection. Thus, the problem in Eq. (15) becomes:

$$\begin{aligned}
&\min_{\mathbf{F},\mathbf{G},\widetilde{\mathbf{W}}} \lambda \sum_{v=1}^{V} \frac{\|\widetilde{\mathbf{W}}_v\|_{2,1}}{\alpha_v} + \gamma \mathrm{Tr}(\begin{bmatrix} \mathbf{F} \\ \mathbf{G} \end{bmatrix}^T \mathbf{L}_{\widetilde{\mathbf{S}}} \begin{bmatrix} \mathbf{F} \\ \mathbf{G} \end{bmatrix}) + \mathrm{Tr}\big((\begin{bmatrix} \mathbf{F} \\ \mathbf{G} \end{bmatrix} - \mathbf{Y})^T \mathbf{U} (\begin{bmatrix} \mathbf{F} \\ \mathbf{G} \end{bmatrix} - \mathbf{Y})\big) \\
&+ \beta \sum_{i=1}^{n}\sum_{j=1}^{m} s_{ij} \|\widetilde{\mathbf{W}}^T \mathbf{x}_i - \widetilde{\mathbf{W}}^T \mathbf{z}_j\|_2^2 + \frac{\mu}{2} \|\mathbf{E} - \mathbf{F} + \mathbf{X}^T \widetilde{\mathbf{W}} + \frac{\mathbf{\Pi}}{\mu}\|_F^2,
\end{aligned} \tag{18}$$

where $\widetilde{\mathbf{W}} = [\alpha_1 \mathbf{W}_1, \cdots, \alpha_V \mathbf{W}_V]^T \in \mathbb{R}^{d\times c}$ denotes the joint feature projection of all the views. Obviously, $\{\boldsymbol{W}_v\}_{v=1}^{V}$ can be determined via solving the joint projection $\widetilde{\boldsymbol{W}}$ with fixed $\{\alpha_v\}_{v=1}^{V}$. Furthermore, we can prove that the objective function in Eq. (18) is jointly convex w.r.t. $\widetilde{\mathbf{W}}$, $\mathbf{F}$ and $\mathbf{G}$.

Proof. We first denote the objective function in Eq. (18) as $g(\widetilde{\mathbf{W}}, \mathbf{F}, \mathbf{G})$, then define a matrix $\mathbf{M}$ as:

$$\mathbf{M} = \begin{bmatrix} \frac{\mu}{2}\mathbf{I} + \gamma \mathbf{I} + \mathbf{U}_n & -\gamma \mathbf{S} & -\frac{\mu}{2}\mathbf{X} \\ -\gamma \mathbf{S}^T & \gamma \mathbf{\Lambda} & \mathbf{0} \\ -\frac{\mu}{2}\mathbf{X}^T & \mathbf{0} & \beta \mathbf{H} + \lambda \mathbf{A} + \frac{\mu}{2}\mathbf{X}\mathbf{X}^T \end{bmatrix} \in \mathbb{R}^{(n+m+d)\times(n+m+d)}, \tag{19}$$

where $\mathbf{H} = \mathbf{X}\mathbf{X}^T + \mathbf{Z}\mathbf{\Lambda}\mathbf{Z}^T - 2\mathbf{X}\mathbf{S}\mathbf{Z}^T$ and $\mathbf{A} = [\mathbf{A}^1, \cdots, \mathbf{A}^V]$ is a diagonal matrix with $\mathbf{A}^v = \frac{1}{2\alpha_v}\mathrm{diag}(\frac{1}{\|\widetilde{\mathbf{w}}_1^v\|_2}, \cdots, \frac{1}{\|\widetilde{\mathbf{w}}_{d_v}^v\|_2})$. Accordingly, $g(\widetilde{\mathbf{W}}, \mathbf{F}, \mathbf{G})$ can be rewritten in the matrix form:

$$g(\widetilde{\mathbf{W}}, \mathbf{F}, \mathbf{G}) = \mathrm{Tr}\left(\begin{bmatrix} \mathbf{F} \\ \mathbf{G} \\ \widetilde{\mathbf{W}} \end{bmatrix}^T \mathbf{M} \begin{bmatrix} \mathbf{F} \\ \mathbf{G} \\ \widetilde{\mathbf{W}} \end{bmatrix}\right) - \mathrm{Tr}\left(\begin{bmatrix} \mathbf{F} \\ \mathbf{G} \\ \widetilde{\mathbf{W}} \end{bmatrix}^T \begin{bmatrix} 2\mathbf{U}_n \mathbf{Y}_n + \mu \mathbf{E} + \mathbf{\Pi} \\ \mathbf{0} \\ -\mu \mathbf{X}\mathbf{E} - \mathbf{X}\mathbf{\Pi} \end{bmatrix}\right). \tag{20}$$

Algorithm 1 : Optimization procedures for EMSFS.

```
Input: Training data $\mathbf{X}$, given labels of labeled data, single-view graphs $\{S^v\}_{v=1}^V$, the number of anchors $m$, and parameters $\lambda$, $\beta$ and $\gamma$;
Initialize $\alpha_v = 1/V$ $(v = 1, \cdots, V)$, $\mathbf{S} = \sum_{v=1}^V \mathbf{S}^v/V$, $\mathbf{A}$ as the identity matrix, $\mathbf{\Pi} = \mathbf{0}$, $\mu = 10^{-4}$ and $\rho = 1.1$; Generate $m$ anchors by $k$-means;
repeat
    Update $\mathbf{E}$ by Eq. (16);
    Update $\mathbf{G}$ by Eq. (22);
    Update $\mathbf{F}$ by Eq. (24);
    repeat
        With current $\mathbf{A}$, update $\widetilde{\mathbf{W}}$ by Eq. (26);
        With current $\widetilde{\mathbf{W}}$, calculate the diagonal matrix $\mathbf{A}$;
    until Eq. (26) converges;
    Update each row of $\mathbf{S}$ by solving Eq. (28);
    Update $\boldsymbol{\alpha}$ by Eq. (29);
    Update $\mathbf{\Pi}$ and $\mu$ by Eq. (30);
until Eq. (15) converges;
Output: The selected $r$ features with the highest scores (i.e. $\{\|\widetilde{\mathbf{w}}_i\|_2\}_{i=1}^d$).
```

To prove that $g(\widetilde{\mathbf{W}}, \mathbf{F}, \mathbf{G})$ is jointly convex, we have to prove that the matrix $\mathbf{M}$ is positive semi-definite. Defining an arbitrary vector $\mathbf{v} = [\mathbf{v}_1^T, \mathbf{v}_2^T, \mathbf{v}_3^T]^T \in \mathbb{R}^{(n+m+d)\times 1}$, where $\mathbf{v}_1 \in \mathbb{R}^{n\times 1}$, $\mathbf{v}_2 \in \mathbb{R}^{m\times 1}$ and $\mathbf{v}_3 \in \mathbb{R}^{d\times 1}$, we have

$$\begin{aligned}\mathbf{v}^T\mathbf{M}\mathbf{v} &= \mathbf{v}_1^T\mathbf{U}_n\mathbf{v}_1 + \mathbf{v}_3^T(\beta\mathbf{H} + \lambda A)\mathbf{v}_3 + \frac{\mu}{2}(\mathbf{v}_1 - \mathbf{X}^T\mathbf{v}_3)^T(\mathbf{v}_1 - \mathbf{X}^T\mathbf{v}_3)\\ &+ \gamma\begin{bmatrix}\mathbf{v}_1\\ \mathbf{v}_2\end{bmatrix}^T\begin{bmatrix}\mathbf{I} & -\mathbf{S}\\ -\mathbf{S}^T & \mathbf{\Lambda}\end{bmatrix}\begin{bmatrix}\mathbf{v}_1\\ \mathbf{v}_2\end{bmatrix}.\end{aligned} \tag{21}$$

As $\mathbf{U}_n$ and $\mathbf{A}$ are nonnegative diagonal matrices, and $\mathbf{H}$ and $\begin{bmatrix}\mathbf{I} & -\mathbf{S}\\ -\mathbf{S}^T & \mathbf{\Lambda}\end{bmatrix}$ are positive semi-definite [42], it can be concluded that $\mathbf{v}^T\mathbf{M}\mathbf{v} \geqslant 0$ is true for all possible $\mathbf{v}$. Thus, $\mathbf{M}$ is positive semi-definite, and Eq. (18) should be jointly convex w.r.t. $\widetilde{\mathbf{W}}$, $\mathbf{F}$ and $\mathbf{G}$.

Taking the derivative of Eq. (18) w.r.t. $\mathbf{G}$ and setting it to zero, we have:

$$\mathbf{G} = \mathbf{\Lambda}^{-1}\mathbf{S}^T\mathbf{F}. \tag{22}$$

Then, we can substitute $\mathbf{G}$ of Eq. (22) into Eq. (18) and set its derivative w.r.t. $\mathbf{F}$ to zero, the solution of $\mathbf{F}$ is obtained as follows:

$$\mathbf{F} = (\mathbf{P} - \gamma\mathbf{S}\mathbf{\Lambda}^{-1}\mathbf{S}^T)^{-1}\mathbf{Q}, \tag{23}$$

where $\mathbf{P} = (\gamma + \frac{\mu}{2})\mathbf{I} + \mathbf{U}_n$ is a diagonal matrix, and $\mathbf{Q} = \mathbf{U}_n\mathbf{Y}_n + \frac{\mu}{2}(\mathbf{E} + \mathbf{X}^T\widetilde{\mathbf{W}} + \frac{\mathbf{\Pi}}{\mu})$. Although the solution of $\mathbf{F}$ in Eq. (23) seems simple, it involves the inverse operation of an $n \times n$ dense matrix (i.e., $\mathbf{P} - \gamma\mathbf{S}\mathbf{\Lambda}^{-1}\mathbf{S}^T$), which takes the computational complexity of $\mathcal{O}(n^3)$ at least. Instead of it, we further exploit the matrix identity[1] to simplify the solution of $\mathbf{F}$ as:

$$\mathbf{F} = \mathbf{P}^{-1}\mathbf{Q} + \mathbf{P}^{-1}\mathbf{S}(\frac{\mathbf{\Lambda}}{\gamma} - \mathbf{S}^T\mathbf{P}^{-1}\mathbf{S})^{-1}\mathbf{S}^T\mathbf{P}^{-1}\mathbf{Q}, \tag{24}$$

where $\frac{\mathbf{\Lambda}}{\gamma} - \mathbf{S}^T\mathbf{P}^{-1}\mathbf{S}$ is an $m \times m$ matrix. Therefore, the inverse operation of the n-order dense matrix in Eq. (23) is equivalently substituted by the inverses of the n-order diagonal matrix (i.e., $\mathbf{P}$) and the m-order matrix (i.e., $\frac{\mathbf{\Lambda}}{\gamma} - \mathbf{S}^T\mathbf{P}^{-1}\mathbf{S}$), as well as several matrix multiplications. By computing the terms in Eq. (24) one by one from right to left, the computational complexity of solving $\mathbf{F}$ can be reduced from $\mathcal{O}(n^3)$ to $\mathcal{O}(nm^2 + ndc + m^3)$, making the proposed EMSFS scale well with the number of training samples.

When other variables are fixed, we substitute the optimal solutions of $\mathbf{G}$ and $\mathbf{F}$ into Eq. (18), then obtain the following subproblem:

$$\begin{aligned}&\min_{\widetilde{\mathbf{W}}} \frac{\mu}{2}\mathrm{Tr}\Big(\widetilde{\mathbf{W}}^T\mathbf{X}(2\mathbf{E} + \mathbf{X}^T\widetilde{\mathbf{W}} + \frac{2}{\mu}\mathbf{\Pi})\Big) - \mathrm{Tr}\Big(\mathbf{Q}^T(\mathbf{P} - \gamma\mathbf{S}\mathbf{\Lambda}^{-1}\mathbf{S}^T)^{-1}\mathbf{Q}\Big)\\ &+ \lambda\mathrm{Tr}(\widetilde{\mathbf{W}}^T\mathbf{A}\widetilde{\mathbf{W}}) + \beta\sum_{i=1}^n\sum_{j=1}^m \widetilde{s}_{ij}\|\widetilde{\mathbf{W}}^T\mathbf{x}_i - \widetilde{\mathbf{W}}^T\mathbf{z}_j\|_2^2.\end{aligned} \tag{25}$$

Substituting $\mathbf{Q} = \mathbf{U}_n\mathbf{Y}_n + \frac{\mu}{2}(\mathbf{E} + \mathbf{X}^T\widetilde{\mathbf{W}} + \frac{\mathbf{\Pi}}{\mu})$ into Eq. (25) and setting its derivative w.r.t.$\widetilde{\mathbf{W}}$ to zero, we have:

$$\widetilde{\mathbf{W}} = (\lambda\mathbf{A} + \beta\mathbf{B} + \mathbf{C})^{-1}\mathbf{D}, \tag{26}$$

where $\mathbf{B} = \mathbf{X}\mathbf{X}^T + \mathbf{Z}\mathbf{\Lambda}\mathbf{Z}^T - \mathbf{X}\mathbf{S}\mathbf{Z}^T - \mathbf{Z}\mathbf{S}^T\mathbf{X}^T$, $\mathbf{C} = \frac{\mu}{2}\mathbf{X}\mathbf{X}^T - \frac{\mu^2}{4}\mathbf{X}(\mathbf{P} - \gamma\mathbf{S}\mathbf{\Lambda}^{-1}\mathbf{S}^T)^{-1}\mathbf{X}^T$, and $\mathbf{D} = \Big(\frac{\mu}{2}\mathbf{X}(\mathbf{P} - \gamma\mathbf{S}\mathbf{\Lambda}^{-1}\mathbf{S}^T)^{-1}(\mathbf{U}_n\mathbf{Y}_n + \frac{\mu}{2}(\mathbf{E} + \frac{\mathbf{\Pi}}{\mu})) - \frac{\mu}{2}\mathbf{X}(\mathbf{E} + \frac{\mathbf{\Pi}}{\mu})\Big)$. Since $\mathbf{A}$ is also unknown and depends on $\widetilde{\mathbf{W}}$, thus we can alternately update $\mathbf{A}$ and $\widetilde{\mathbf{W}}$.

[1] $(\boldsymbol{A} + \boldsymbol{B}\boldsymbol{C}\boldsymbol{B}^T)^{-1} = \boldsymbol{A}^{-1} - \boldsymbol{A}^{-1}\boldsymbol{B}(\boldsymbol{C}^{-1} + \boldsymbol{B}^T\boldsymbol{A}^{-1}\boldsymbol{B})^{-1}\boldsymbol{B}^T\boldsymbol{A}^{-1}$.

Table 2
The computational complexity of multi-view methods.

Methods	Constructing graph **S**	Solving **F**	Total computational complexity
MLSFS	$\mathcal{O}(n^2 d)$	$\mathcal{O}(n^3 + ndc)$	$\mathcal{O}(n^3 + n^2 d + d^3)$
MASFS	$\mathcal{O}(n^2 d)$	$\mathcal{O}(n^3 + ndc)$	$\mathcal{O}(n^3 + n^2 d + d^3)$
SMFS	$\mathcal{O}(n^2 d)$	$\mathcal{O}(n^3 + ndc)$	$\mathcal{O}(n^3 + d^3)$
EMSFS	$\mathcal{O}(nmd + nm \log m)$	$\mathcal{O}(nm^2 + ndc + m^3)$	$\mathcal{O}(nm^2 + nd^2 + nmd + m^3 + d^3)$

Update S: By fixing other variables, the subproblem of **S** becomes:

$$\min_{\mathbf{S1}=\mathbf{1},\mathbf{S}\geqslant 0} \gamma \sum_{i=1}^{n}\sum_{j=1}^{m} \|\mathbf{f}_i - \mathbf{g}_j\|_2^2 s_{ij} + \beta\eta \sum_{i=1}^{n}\sum_{j=1}^{m} s_{ij}^2$$
$$+ \beta \sum_{i=1}^{n}\sum_{j=1}^{m} s_{ij} \|\widetilde{\mathbf{W}}^T \mathbf{x}_i - \widetilde{\mathbf{W}}^T \mathbf{z}_j\|_2^2, \tag{27}$$

where $\mathbf{f}_i$ is the i-th row of **F**, and $\mathbf{g}_j$ is the j-th row of **G**. Eq. (27) is independent for different rows, such that each row of **S** (i.e., $\mathbf{s}_i$) can be separately optimized:

$$\min_{\mathbf{s}_i \mathbf{1}=1, \mathbf{s}_i \geqslant 0} \|\mathbf{s}_i + \frac{1}{2\eta}\mathbf{u}_i\|_2^2, \tag{28}$$

where $\mathbf{u}_i$ is a row vector with $u_{ij} = \frac{\gamma}{\beta}\|\mathbf{f}_i - \mathbf{g}_j\|_2^2 + \|\widetilde{\mathbf{W}}^T \mathbf{x}_i - \widetilde{\mathbf{W}}^T \mathbf{z}_j\|_2^2$. Eq. (28) can be efficiently optimized with a closed-form solution, and the regularization parameter η can be automatically determined via the k nearest anchors [34].

Update $\boldsymbol{\alpha}$: By fixing other variables, $\boldsymbol{\alpha}$ can be solved as follows (see [26]):

$$\alpha_v = \frac{\|\widetilde{\mathbf{W}}_v\|_{2,1}^{\frac{1}{2}}}{(\sum_{v=1}^{V} \|\widetilde{\mathbf{W}}_v\|_{2,1}^{\frac{1}{2}})}. \tag{29}$$

Update $\boldsymbol{\Pi}$ and μ: In each iteration, the Lagrange multipliers $\boldsymbol{\Pi}$ and penalty parameter μ are respectively updated as follows:

$$\boldsymbol{\Pi} = \boldsymbol{\Pi} + \mu(\mathbf{E} - \mathbf{F} + \mathbf{X}^T \widetilde{\mathbf{W}})$$
$$\mu = \rho\mu, \tag{30}$$

where $1 < \rho < 2$ denotes the updated rate, which is a constant.

The main procedures for solving Eq. (15) are summarized in Algorithm 1. The proposed EMSFS is iteratively optimized, whose main computational complexity comes from the generation of anchors and the updates of **E**, **G**, $\widetilde{\mathbf{W}}$, **F** and **S**. Firstly, using k-means to generate m anchors takes $\mathcal{O}(nmd)$. Then, in each iteration, updating **G**, **F** and $\widetilde{\mathbf{W}}$ costs $\mathcal{O}(nmc)$, $\mathcal{O}(nm^2 + ndc + m^3)$ and $\mathcal{O}(nm^2 + nd^2 + nmd + ndc + m^3 + d^3)$, respectively. Besides, solving **E** takes $\mathcal{O}(ndc)$, and calculating **S** needs $\mathcal{O}(nmd + nm\log m + nkd)$, where k is the number of neighbors. Since c and k are small constants, EMSFS approximately costs the computational complexity of $\mathcal{O}(nm^2 + nd^2 + nmd + m^3 + d^3)$, which is linearly related to the training data size n, making EMSFS more efficient than the state-of-the-art competitors. We summarize the computational costs of EMSFS and other multi-view semi-supervised feature selection methods in Table 2, which includes the complexity of constructing graph **S** and solving **F**, as well as their total computational costs.

4. Experiments

In this section, we present a series of experiments that aim to demonstrate the superiority of our proposed EMSFS method. The experiments are divided into two parts. In the first part, we evaluate the effectiveness and efficiency of EMSFS on real-world multi-view datasets. In the second part, we conduct a comprehensive analysis of EMSFS from various perspectives to gain a better understanding of its performance and characteristics.

4.1. Experimental datasets and settings

To comprehensively assess the effectiveness of EMSFS, we conduct a comparative study with five state-of-the-art feature selection methods, namely FSSLL [19], MLSFS [22], MASFS [25], SMFS [26], and multi-view sparse feature selection (MSFS) [43]. Our experiments employ eight widely used multi-view datasets, with data sizes ranging from 1440 to 70000 and feature sizes ranging from 192 to 2801, including COIL20, Leaves, Handwritten, SCENE, COIL100, ALOI, NUS-WIDE and MNIST. Further details on each dataset are provided in Table 3.

For each dataset except MNIST, we randomly select 80% of samples for training. On the MNIST dataset, we use 30% of samples for training and the rest for testing. With different labeled ratios, the training data is randomly divided into the labeled and unlabeled sample sets. The number of anchors m is determined by the training sample sizes on different datasets (e.g., $m = 1\% \times n$ for

Table 3
Detailed information on multi-view datasets.

View	COIL20	Leaves	Handwritten	SCENE	COIL100	ALOI	NUS-WIDE	MNIST
#1	GIST(512)	SD(64)	PIX(240)	GIST(512)	PCA(200)	CS(77)	HIST(64)	GIST(256)
#2	HOG(420)	FSM(64)	FOU(76)	HOG(432)	NPE(200)	Haralick(13)	BCM(225)	HOG(144)
#3	LBP(1239)	TH(64)	FAC(216)	LBP(256)	ISOP(200)	HIST(64)	ACG(144)	LBP(59)
#4	SIFT(630)	-	ZER(47)	GABOR(48)	-	HSV(125)	EDH(73)	-
#5	-	-	KAR(64)	-	-	-	WT(128)	-
#6	-	-	MOR(6)	-	-	-	-	-
Feature size	2801	192	649	1248	600	279	634	459
Classes	20	100	10	8	100	100	12	10
Data size	1440	1600	2000	2688	7200	10800	12000	70000

MNIST, and $m = 5\% \times n$ for other datasets). To ensure fair comparisons, the parameters of all methods are turned from the range of $\{10^{-3}, 10^{-2}, \cdots, 10^{3}\}$, and each method is run 20 times on different training and testing partitions independently. Following the conventions of feature selection, different feature selection methods are first implemented on the training samples to select relevant features, then the Regularized Least Square Classification (RLSC) with a fixed regularization parameter is employed to train a classifier on the selected features. Finally, the results of each method on the testing samples are reported to evaluate the effectiveness of selected features.

4.2. The effectiveness and efficiency for multi-view feature selection

To assess the effectiveness of the selected features, we performed feature selection with a fixed number of labeled samples. Specifically, we used a 3% labeled ratio for the MNIST dataset and a 30% labeled ratio for the other datasets. Table 4 reports the classification results obtained with a varying number of selected features, where "OM" denotes an out-of-memory error encountered during the experiments. Additionally, we fixed the number of selected features at 25% of the total number of features (d) and evaluated the impact of labeled samples on feature selection. The corresponding results are presented in Table 5, where the ratios of labeled samples are varied from $\{1\%, 2\%, 3\%, 4\%\}$ for MNIST and $\{10\%, 20\%, 30\%, 40\%\}$ for the other datasets. Based on the experimental results presented in Table 4 and Table 5, we can draw the following conclusions:

- EMSFS consistently achieves competitive or superior results with the different number of selected features, highlighting the effectiveness and superiority of EMSFS over state-of-the-art methods.
- EMSFS achieves superior results compared to the supervised method (i.e., MSFS) and the single-view method (i.e., FSSLL), which verifies that mining the similarity structure of unlabeled data and taking into account the correlations and distinctions among views indeed benefit the multi-view feature selection.
- EMSFS outperforms MLSFS and MASFS in most situations, highlighting the effectiveness of adaptively assigning appropriate weights to different views and exploiting the similarity relations in the projected feature space for feature selection. Additionally, EMSFS can avoid the memory overflow problem associated with handling large-scale data, whereas MLSFS, MASFS, and SMFS encounter the out-of-memory error on the MNIST dataset. These findings demonstrate the efficiency and scalability of EMSFS compared to other methods.
- As the number of labeled samples increases, EMSFS exhibits stable performance improvement and outperforms most other methods, indicating its ability to select discriminative features with a sufficient number of labeled samples.

To comprehensively analyze the performance between EMSFS and other methods with statistical significance, we further adopt the Friedman test combining with the two-tailed Bonferroni-Dunn test [44] to make a statistical test. Specifically, the results of two methods are significantly different if their average ranks[2] on all datasets differ by at least the critical difference (CD): $CD = q_\pi \sqrt{\frac{p(p+1)}{6N}}$, where p is the number of methods, N is the number of datasets, π denotes the significance level, and q_π is the critical value. According to the experimental results in Table 5 of the revised manuscript, three multi-view semi-supervised feature selection methods, MLSFS, MASFS and SMFS are chosen to compare with the proposed EMSFS. Choosing $\pi = 5\%$ and $q_\pi = 2.39$ ($p = 4$), the critical difference becomes $CD = 1.55$. Fig. 2 demonstrates the Friedman test results of the proposed EMSFS and other multi-view semi-supervised feature selection methods with different labeled ratios. We can observe that the differences between EMSFS and the method (i.e., MLSFS) that uses fixed similarity graphs during feature selection, and the differences between EMSFS and the method (i.e., MASFS) that updates graphs according to the similarity structure in the original feature space, are greater than CD, which means that EMSFS is significantly better than MLSFS, and MASFS. This demonstrates that it is effective to integrate similarity graph learning into the feature selection process and adaptively learn the similarity structure in the projected feature subspace. Although the differences between EMSFS and the method (i.e., SMFS) that explores the structure information in the original data space and the projected

[2] We rank EMSFS with other multi-view semi-supervised feature selection methods and record their ranks as 1, 2 and so on. Average ranks are assigned in the case of ties. The average rank of each method is obtained by averaging over all of the datasets.

Table 4
Results (ACC± STD%) on the testing data with various numbers of selected features. The best results are in bold, “OM” denotes an out-of-memory error and * denotes the results are not significantly worse than the best using the paired t-test at the 5% significance level.

Datasets	Feature ratio	MSFS	FSSLL	MLSFS	MASFS	SMFS	**EMSFS**
COIL20	10%	95.71±1.20	95.96±1.98	96.98±1.16	96.92±1.01	97.79±0.87*	**98.16**±0.65
	15%	96.88±1.16	97.79±1.10	98.30±0.75	98.37±1.19	**98.85**±0.41	98.58±0.53*
	20%	98.06±0.52	98.35±1.01	99.02±0.64	99.13±0.78	**99.44**±0.48	99.27±0.42*
	25%	98.47±0.87	98.65±1.08	99.27±0.39	99.34±0.61*	99.51±0.38*	**99.59**±0.34
	30%	98.78±0.80	98.89±0.73	99.21±0.42	99.41±0.53	99.62±0.35*	**99.63**±0.33
	35%	98.99±0.69	99.18±0.69	99.40±0.38	99.51±0.43*	99.57±0.37*	**99.62**±0.37
	40%	99.03±0.64	99.35±0.52	99.34±0.36	99.54±0.42*	99.57±0.40*	**99.65**±0.31
Leaves	10%	35.15±3.31	41.92±1.12	41.78±3.80	41.96±3.84	42.73±3.32	**48.60**±3.84
	15%	49.69±2.44	58.56±1.17	60.55±3.99	61.11±5.67	61.23±4.08	**67.70**±3.68
	20%	59.96±3.24	66.59±3.02	68.33±4.26	69.56±3.76	69.75±3.33	**74.98**±3.01
	25%	65.73±3.40	71.94±1.23	74.02±3.54	75.67±3.50	76.89±2.30	**79.09**±2.13
	30%	71.66±2.75	75.28±1.80	78.73±3.04	79.83±2.10	79.38±2.82	**83.01**±2.23
	35%	75.62±2.90	78.25±1.14	81.92±2.61	82.84±2.77	83.14±2.76*	**85.31**±1.82
	40%	78.02±3.32	79.97±1.09	82.94±2.40	84.42±1.81	84.42±2.53*	**86.39**±1.93
Handwritten	10%	95.14±0.98	96.30±1.29	97.61±0.88*	**97.63**±0.81	97.56±0.80*	97.60±0.61*
	15%	96.17±0.91	97.40±1.46	97.96±0.62*	**98.06**±0.46	97.90±0.82*	97.79±0.55*
	20%	96.36±1.08	97.65±1.54	98.10±0.50*	98.16±0.55*	**98.21**±0.72	97.84±0.70*
	25%	96.69±0.93	97.70±1.04	97.99±0.50*	98.06±0.47*	**98.08**±0.64	98.00±0.60*
	30%	96.86±0.80	97.62±0.97*	97.93±0.53*	97.99±0.51*	97.94±0.68*	**98.01**±0.82
	35%	96.74±0.71	97.55±1.17*	97.88±0.50*	97.88±0.70*	97.90±0.78*	**97.97**±0.84
	40%	96.89±0.65	97.65±0.95*	97.84±0.57*	97.90±0.50*	97.81±0.57*	**97.92**±0.56
SCENE	10%	48.16±2.76	49.41±2.10	49.61±1.70	50.03±2.37*	50.61±2.32*	**51.97**±2.38
	15%	45.78±2.11	47.14±1.62	47.45±1.84	47.58±1.55	48.18±0.55*	**48.94**±1.54
	20%	44.70±3.06	44.93±1.91	44.98±1.69	45.09±1.42	45.37±2.27*	**45.89**±1.99
	25%	41.47±2.26	42.36±2.00	42.75±1.33	42.81±2.03	43.55±0.95	**44.53**±1.29
	30%	37.89±3.01	38.27±1.46	38.64±2.24	38.98±1.60	**39.98**±1.25	39.69±2.06*
	35%	34.43±2.72	34.57±2.40	34.64±2.69	34.67±1.98	35.06±1.12*	**35.54**±1.61
	40%	31.45±3.18	31.71±1.82	31.78±2.25	31.97±1.93	32.55±1.45*	**32.96**±2.03
COIL100	10%	62.99±1.31	67.17±1.89	88.10±0.86	88.49±1.26*	88.45±1.09*	**88.85**±1.07
	15%	68.58±1.13	74.61±1.00	95.61±0.55	95.87±0.49	95.84±0.51*	**96.24**±0.49
	20%	70.24±0.93	76.74±1.57	94.88±0.51	95.14±0.50	95.28±0.67*	**95.76**±0.44
	25%	71.15±1.30	78.03±1.28	93.76±0.42	94.30±0.75	94.68±0.69	**95.24**±0.38
	30%	73.51±1.17	79.43±1.97	92.28±0.45	92.86±0.74	93.71±0.86*	**94.02**±0.51
	35%	74.51±1.02	80.76±2.38	90.79±0.53	91.52±0.91	**92.88**±1.08	92.68±0.86*
	40%	74.53±1.08	81.60±1.93	89.72±0.78	90.47±1.07	**92.28**±1.25	91.74±0.82*
ALOI	10%	49.10±2.32	66.16±4.45	65.50±2.29	69.04±2.16	69.95±1.93	**77.29**±2.41
	15%	66.47±6.12	80.91±1.48	79.43±2.90	80.07±2.12	86.29±1.65*	**87.16**±1.15
	20%	78.67±2.74	85.85±1.22	86.15±1.55	85.70±1.61	**89.99**±1.02	89.64±1.11*
	25%	80.66±2.42	89.03±0.90	90.88±0.80	90.81±0.80	**91.70**±1.02	91.42±1.44*
	30%	83.19±2.28	90.79±0.84	92.42±0.76	92.43±0.85*	**93.00**±0.91	92.55±1.01*
	35%	84.92±2.23	91.72±0.89	93.18±0.83*	93.24±0.84*	**93.39**±0.81	92.97±0.77*
	40%	86.00±1.95	91.97±0.90	93.42±0.78*	93.46±0.85*	**93.47**±0.77	93.25±0.92*
NUS-WIDE	10%	40.68±1.43	42.51±1.12	45.80±0.82	46.15±0.88*	45.45±0.89	**46.48**±0.97
	15%	44.58±1.34	44.88±2.70	47.13±0.91	47.95±0.80	47.29±0.79	**48.49**±0.81
	20%	46.03±0.78	46.26±1.88	47.32±0.87	48.40±1.02*	47.70±0.98	**48.85**±0.88
	25%	46.81±1.08	47.02±2.67	47.76±0.93	48.76±0.98*	48.10±0.86	**48.87**±0.80
	30%	47.56±0.81	47.63±1.93	47.80±0.89	48.85±1.07*	48.50±0.78*	**49.17**±0.82
	35%	47.80±0.78	48.12±1.47	47.83±0.83	48.63±0.89*	48.48±0.76*	**49.09**±0.91
	40%	47.82±1.10	47.82±1.11	47.91±0.64	48.28±0.77*	48.39±0.86*	**48.97**±0.80
MNIST	10%	69.81±1.63	70.08±1.23	OM	OM	OM	**78.89**±0.48
	15%	73.45±1.18	74.37±1.75	OM	OM	OM	**79.79**±0.44
	20%	74.63±1.19	75.89±1.39	OM	OM	OM	**79.87**±0.41
	25%	75.33±1.04	76.10±2.73	OM	OM	OM	**79.77**±0.48
	30%	76.30±1.07	77.14±1.80	OM	OM	OM	**79.53**±0.53
	35%	76.25±1.40	77.99±1.03	OM	OM	OM	**79.06**±0.62
	40%	76.01±1.44	78.05±1.11*	OM	OM	OM	**78.71**±0.76

feature space are lesser than CD, SMFS encounters the out-of-memory error on the large-scale dataset (i.e., MNIST). Generally, the statistical significance test shows that the proposed EMSFS achieves pretty sound improvements in multi-view semi-supervised feature selection.

Meanwhile, we present the running times of each method on eight datasets in Fig. 3 to assess the efficiency of EMSFS. As shown in Fig. 3, the running time of EMSFS exhibits a linear increase with respect to the training data size, validating that the learned

Table 5
Results (ACC± STD%) on the testing data with different numbers of labeled samples. The best results are in bold, "OM" denotes an out-of-memory error and * denotes the results are not significantly worse than the best using the paired t-test at the 5% significance level.

Datasets	Labeled ratio	MSFS	FSSLL	MLSFS	MASFS	SMFS	**EMSFS**
COIL20	10%	94.55±1.21	95.97±1.15	96.18±1.78*	96.28±1.27*	96.35±1.08*	**96.46**±1.26
	20%	97.12±0.70	97.85±0.94	98.26±0.54	98.61±1.27*	98.56±0.40*	**98.78**±0.47
	30%	98.47±0.87	98.65±1.08	99.27±0.39	99.34±0.61*	99.51±0.38*	**99.59**±0.34
	40%	98.99±0.68	99.13±0.41	99.31±0.57	99.58±0.34	99.76±0.22*	**99.83**±0.15
Leaves	10%	42.50±2.56	44.81±1.23	45.98±5.81	45.98±5.60	**47.14**±3.73	46.75±3.47*
	20%	65.02±3.25	68.12±4.93	70.05±3.81	72.47±3.84*	70.83±3.32	**72.94**±2.57
	30%	65.73±3.40	71.94±1.23	74.02±3.54	75.67±3.50	76.89±2.30	**79.09**±2.13
	40%	70.41±3.28	73.87±4.33	75.00±4.70	75.97±2.56	76.44±2.83	**79.72**±2.79
Handwritten	10%	88.75±1.92	92.89±0.90	93.35±1.38	94.47±1.45*	94.83±1.34*	**94.94**±1.39
	20%	95.50±1.38	96.52±1.24	97.26±0.80*	97.53±0.69*	97.58±1.34*	**97.71**±0.66
	30%	96.69±0.93	97.70±1.04	97.99±0.50*	98.06±0.47*	**98.08**±0.64	98.00±0.60*
	40%	97.03±0.74	97.68±0.86	98.28±0.60*	98.28±0.56*	98.42±0.59*	**98.44**±0.41
SCENE	10%	24.14±2.64	25.61±2.18	26.95±1.64	27.30±1.36	30.49±1.25	**32.48**±2.25
	20%	32.51±1.98	33.10±2.26	33.85±1.92*	34.05±2.45*	34.44±2.92*	**34.68**±1.64
	30%	41.47±2.26	42.36±2.00	42.75±1.33	42.81±2.03	43.55±0.95	**44.53**±1.29
	40%	44.50±2.59	44.50±2.30	46.48±0.80	46.00±1.63	46.67±1.72	**47.74**±1.57
COIL100	10%	63.42±2.90	70.65±2.80	88.63±2.51	88.50±1.78	88.45±2.60	**89.78**±1.18
	20%	68.53±1.84	75.59±1.60	92.15±1.50	93.05±1.51*	93.31±1.58*	**93.85**±0.86
	30%	71.15±1.30	78.03±1.28	93.76±0.42	94.30±0.75	94.68±0.69	**95.24**±0.38
	40%	72.54±1.34	79.61±1.65	94.50±0.66	94.97±0.70*	95.14±0.74*	**95.39**±0.56
ALOI	10%	73.29±1.86	87.75±1.38	88.37±1.22	89.40±1.20*	88.08±2.11	**89.92**±1.34
	20%	83.99±1.69	89.15±1.16	89.83±0.91	90.45±0.88	**91.17**±0.86	90.71±1.40*
	30%	80.66±2.42	89.03±0.90	90.88±0.80	90.81±0.80	**91.70**±1.02	91.42±1.44*
	40%	84.22±2.52	89.66±1.12	90.73±0.82	89.92±1.10	91.70±0.91*	**92.65**±0.89
NUS-WIDE	10%	40.38±1.18	41.53±2.67	**41.73**±1.11	41.59±0.88*	41.27±1.29*	41.70±0.84*
	20%	44.45±0.88	45.02±1.04	45.25±1.05	46.21±1.16*	46.35±0.95*	**46.68**±0.81
	30%	46.81±1.08	47.02±2.67	47.76±0.93	48.76±0.98*	48.10±0.86	**48.87**±0.80
	40%	48.18±0.90	48.43±0.70	48.72±0.67	49.63±0.70*	49.56±0.95*	**49.87**±0.86
MNIST	1%	63.22±4.10	64.46±2.71	OM	OM	OM	**68.73**±2.61
	2%	72.77±2.09	74.67±2.35	OM	OM	OM	**77.48**±0.71
	3%	75.33±1.04	76.10±2.73	OM	OM	OM	**79.77**±0.48
	4%	77.67±2.18	78.37±1.64	OM	OM	OM	**80.65**±0.42

bipartite graph can effectively improve the computational efficiency of EMSFS and make it scalable to relatively large-scale data. Specifically, other multi-view feature selection methods that involve the inverse operations of high-order matrices (i.e., MASFS, MLSFS, and SMFS) emerge exponentially increasing running times with the increase of training data sizes. Moreover, these methods encounter the out-of-memory error when applied to the MNIST dataset. Although the single-view feature selection method (i.e., FSSLL) can deal with the MNIST, it shows worse classification performance and costs more running time than EMSFS. These results fully demonstrate that EMSFS is not only effective in achieving superior classification accuracy but also efficient in exploiting the similarity relationships in the projected feature subspace to learn a bipartite graph that is compatible across views.

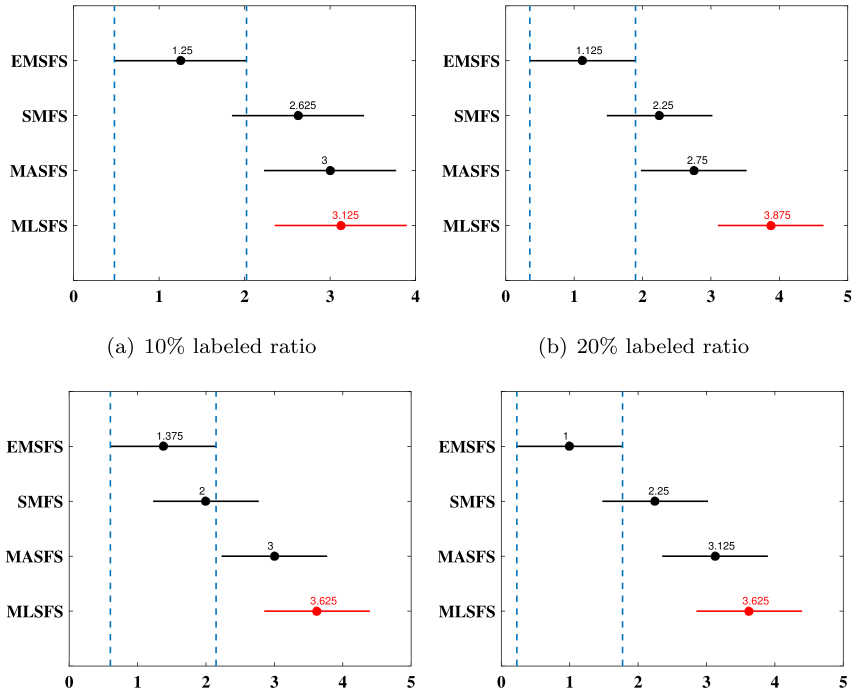
4.3. The analysis for EMSFS

4.3.1. Effect of anchor points' number

Section 3.2 provides the computational complexity analysis, revealing that the number of anchors m has a direct impact on EMSFS. To investigate the effects of anchors on classification accuracy and running time, we conduct experiments with different numbers of anchors, with the results recorded in Fig. 4. Specifically, we varied the numbers of anchors from $\{1\%, 2\%, \cdots, 9\%\} \times n$ on the NUS-WIDE and $\{0.2\%, 0.4\%, \cdots, 1.8\%\} \times n$ on the MNIST. The findings show that, as the number of anchors increases, the running time on each dataset grows steadily (shown in Figs. 4 (a) and (c)). However, there is only a slight change in the accuracy (shown in Figs. 4 (b) and (d)), demonstrating that it might not be effective for EMSFS to use more anchors. Therefore, to ensure effectiveness and efficiency, we should generate a proper number of anchors to maintain accuracy without significantly increased time.

4.3.2. Ablation study

In this section, we conduct an ablation study from two perspectives. We first introduce a simplified version of the proposed EMSFS, named EMSFS$_1$, which sets $\{\alpha_v = 1/V\}_{v=1}^{V}$ and removes the procedures for adaptively updating $\{\alpha_v\}_{v=1}^{V}$. We then obtain another simplified version of EMSFS, named EMSFS$_2$, which uses a fixed graph (e.g., $\mathbf{S} = \sum_{v=1}^{V} \mathbf{S}^v / V$) during feature selection. The experimental results on testing data with varying proportions of labeled samples are presented in Fig. 5. Our findings reveal



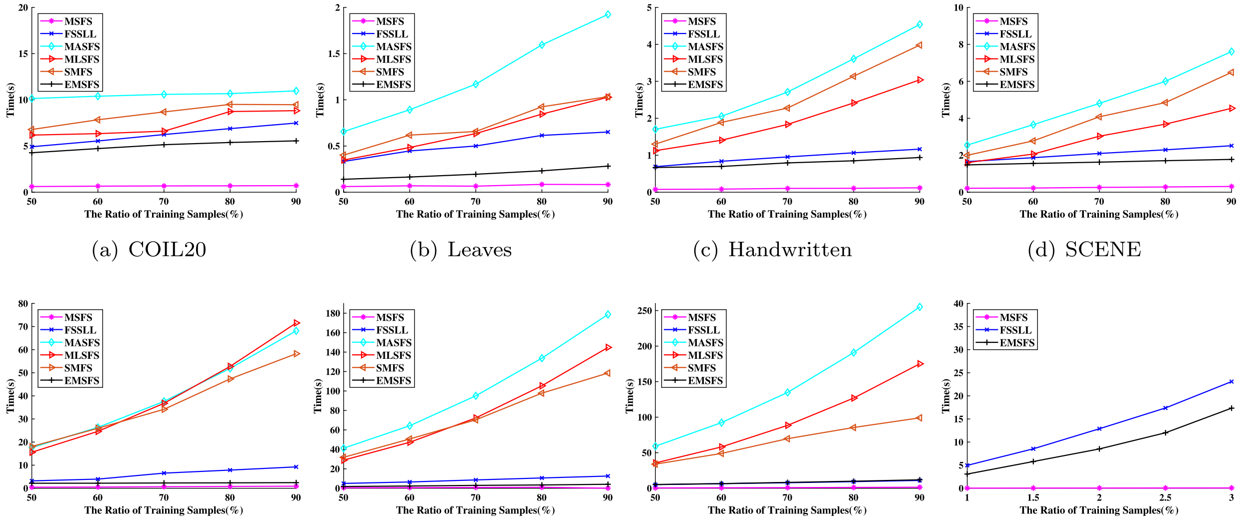
(a) 10% labeled ratio

(b) 20% labeled ratio

(c) 30% labeled ratio

(d) 40% labeled ratio

Fig. 2. The Friedman test for the performance of EMSFS and other multi-view semi-supervised feature selection methods. The dots denote the average ranks, the blue bars indicate the critical value with the post-hoc tests at a 5% significance level, and the methods having non-overlapped bars are significantly inferior to EMSFS.



(a) COIL20

(b) Leaves

(c) Handwritten

(d) SCENE

(e) COIL100

(f) ALOI

(g) NUS-WIDE

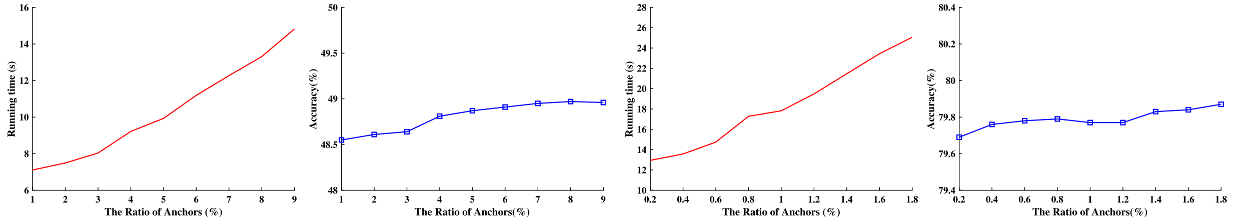
(h) MNIST

Fig. 3. Running time versus the number of training samples (i.e. $n = T \times \text{ratio}\%$).

that EMSFS outperforms its two simplified versions, suggesting that discriminating different feature projections and adaptive graph learning are crucial in enhancing the performance of multi-view feature selection.

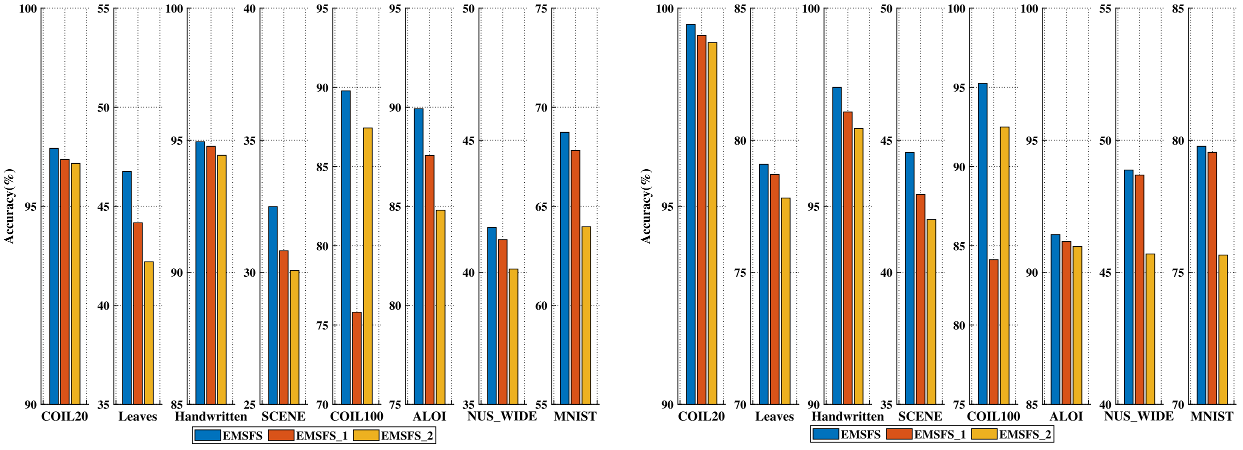
4.3.3. Robustness against noisy features (views)

To investigate the robustness of EMSFS against noise, we conduct experiments on the Leaves$_{noise}$ dataset, which includes three normal views from the Leaves dataset and one view with 64 Gaussian noise features. Figs. 6(a) and 6(b) illustrate the comparison results of EMSFS on the Leaves and Leaves$_{noise}$ with varying numbers of selected features. It can be observed that the performance



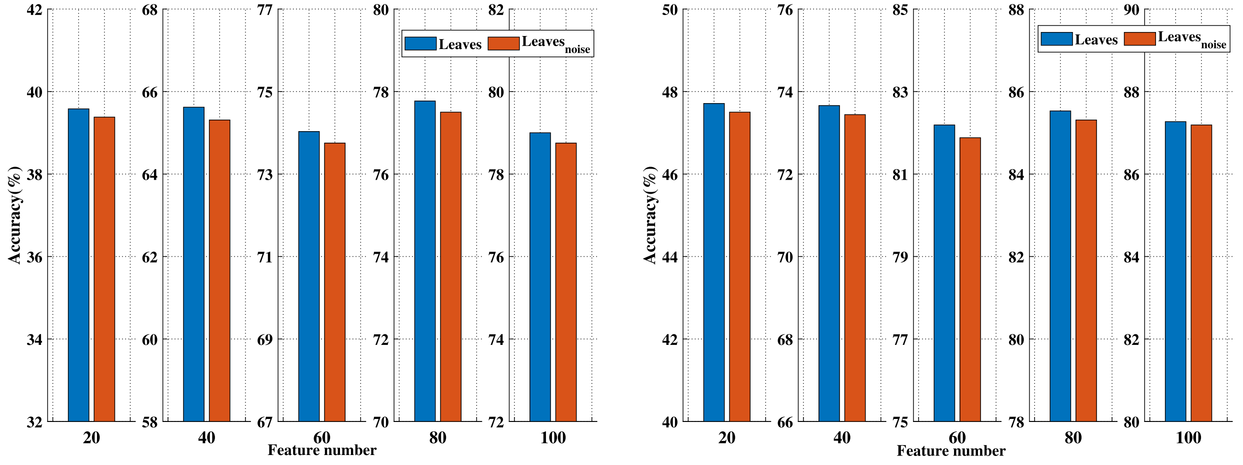
(a) Time (b) Accuracy (c) Time (d) Accuracy

Fig. 4. Accuracy and running time versus the number of anchors (i.e. $m = n\times\text{ratio}\%$), in which (a) and (b) show the results on the NUS-WIDE, and (c) and (d) on the MNIST.



(a) 10% or 1% labeled samples (b) 30% or 3% labeled samples

Fig. 5. Results of EMSFS, EMSFS$_1$ and EMSFS$_2$ on testing samples.



(a) 20% labeled samples (b) 30% labeled samples

Fig. 6. The results of robustness analysis on Leaves and Leaves$_{noise}$ datasets, in which (a) and (b) represent the performance with 20% and 30% labeled samples, respectively.

of EMSFS on the Leaves$_{noise}$ has a slight decrease when comparing the performance on the Leaves, indicating that EMSFS has better robustness against noisy features. Accordingly, Fig. 7 further shows the view weights after each iteration on the Leaves$_{noise}$. We find that the weight of view$_4$ decreases significantly in the initial iterations and gradually tends to be zero after several iterations. These results demonstrate that EMSFS can effectively identify the noisy views and assigns fewer view weights to them, thereby reducing

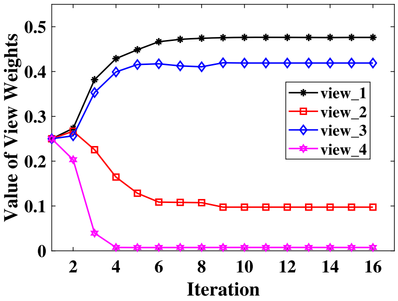
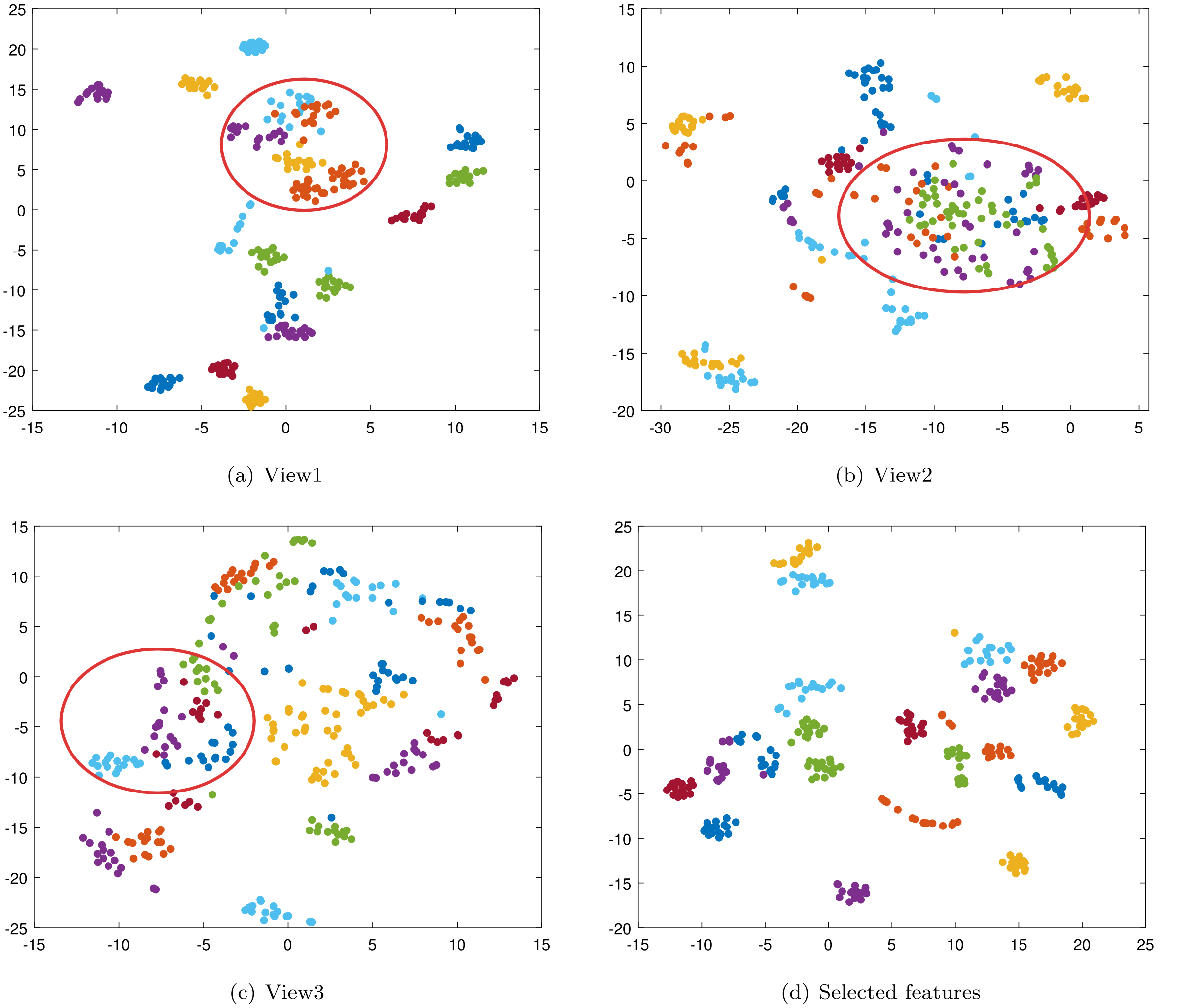


Fig. 7. The view weight curves versus the number of iterations on the Leaves$_{noise}$ dataset.



(a) View1

(b) View2

(c) View3

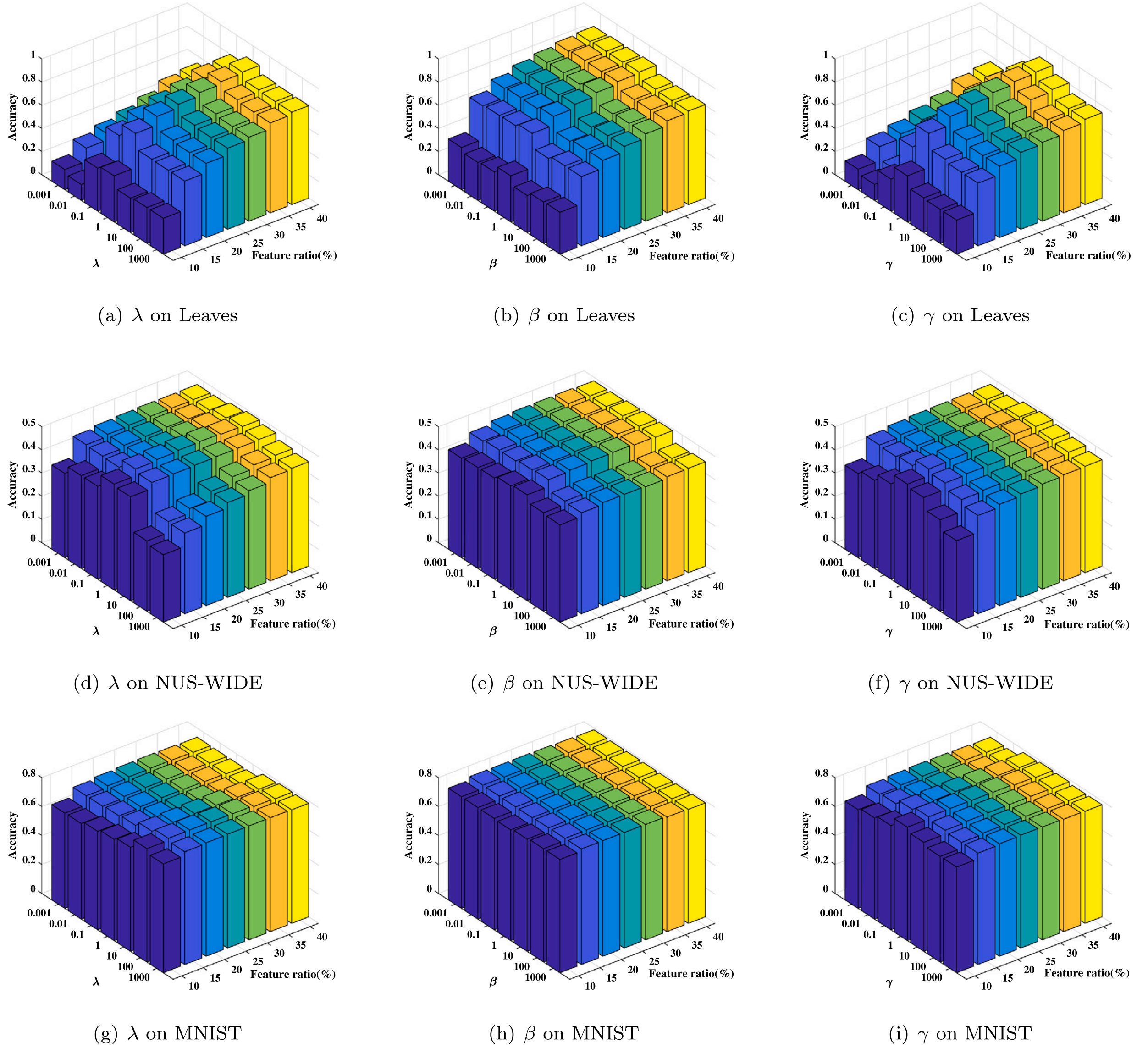
(d) Selected features

Fig. 8. Visualization on the Leaves dataset, in which (a), (b) and (c) show the t-SNE visualization results of three views, and (d) shows the visualization of the selected features.

the interference of the features in the noisy views. Therefore, EMSFS is effective and robust for multi-view feature selection in the presence of noise.

4.3.4. Visualization

To intuitively confirm the effectiveness of selected features, the t-SNE method is employed to visualize the high-dimensional feature space in a two-dimensional space [45]. For convenience, we only select the 320 samples from 20 categories of Leaves for visualization, in which each sample has three views (denoted as View1, View2, and View3) and each view has 64 relevant



(a) λ on Leaves (b) β on Leaves (c) γ on Leaves

(d) λ on NUS-WIDE (e) β on NUS-WIDE (f) γ on NUS-WIDE

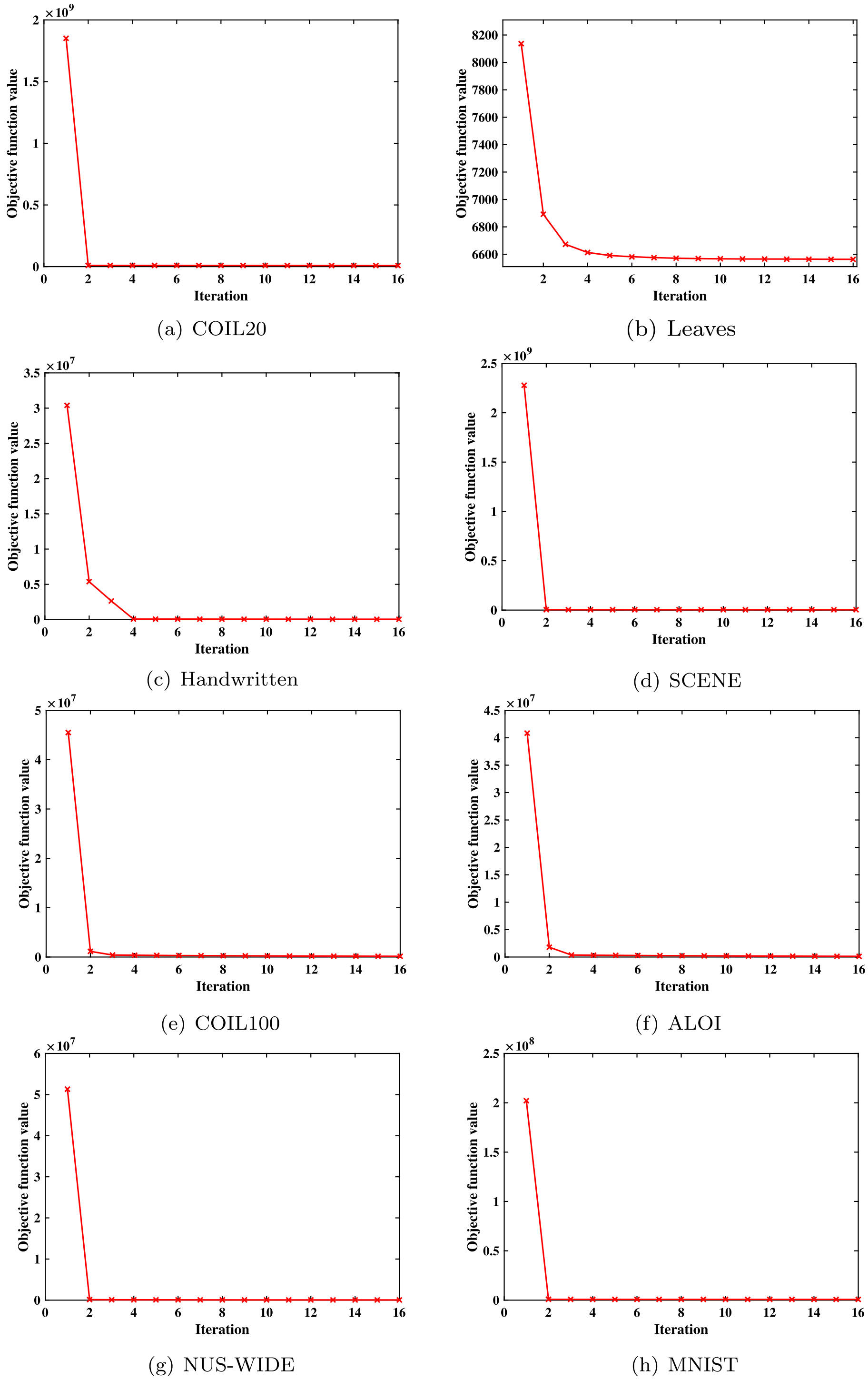
(g) λ on MNIST (h) β on MNIST (i) γ on MNIST

Fig. 9. Accuracy with different parameters on the Leaves, NUS-WIDE and MNIST.

features. For a fair comparison, the number of selected features is set as 64. The original features of different views are respectively visualized in Figs. 8 (a)-(c), and the selected features are visualized in Fig. 8 (d). As depicted in Fig. 8, there are different degrees of overlaps between different categories of samples in the original feature space, while the visualization results of the selected features outperform those of the original features. Specifically, we can find that samples from different classes are effectively separated in Fig. 8 (d), and the inter-class distances can be enlarged by the feature subset selected by EMSFS. This result fully validates that EMSFS can select informative and discriminative features that facilitate the subsequent classification task.

4.3.5. Parameter sensitivity and convergence analysis

In the proposed EMSFS method, there are three parameters λ, γ and β that need to be determined. Specifically, the parameter λ balances the regression loss and the sparsity of the feature projection matrices $\{\boldsymbol{W}_v\}_{v=1}^{V}$. In the Eq. (15), we can observe that using larger λ to minimize the problem can make the rows of learned feature projections sparser. The parameter γ controls the smoothness of the predicted label matrix, making neighbor samples share similar labels. The parameter β balances the importance of the graph learning that facilitates the label propagation on the learned graph. To analyze the influence of these parameters on the performance, we vary one parameter and the ratio of selected features by fixing other regularization parameters. Due to the space limitation, Fig. 9 illustrates the parameter sensitivity of EMSFS on the Leaves, NUS-WIDE and MNIST datasets. The results indicate that parameter determination takes an impact on the performance of the proposed EMSFS. Specifically, EMSFS is somewhat sensitive to λ, γ and β when the number of selected features is small. As the number of selected features increases, EMSFS can produce



(a) COIL20

(b) Leaves

(c) Handwritten

(d) SCENE

(e) COIL100

(f) ALOI

(g) NUS-WIDE

(h) MNIST

Fig. 10. Variation curves of objective function values.

satisfactory results with $\gamma = 1$, demonstrating that the discriminative features are selected. Moreover, EMSFS exhibits low sensitivity to the parameters (i.e., λ and β) and achieves better performance with varying λ and β from $\{10^{-1}, 10^{0}, 10^{1}\}$. This indicates that the terms of sparse projection learning and graph learning play significant roles in identifying informative features. Secondly, to validate the convergence of EMSFS, Fig. 10 displays the variation curves of the objective function over the number of iterations. From the results in Fig. 10, we find that the objective function decreases rapidly in the first iteration and converges within a few iterations, experimentally validating that the optimization strategy is effective and efficient.

5. Conclusion and future work

In this paper, we propose an efficient multi-view semi-supervised feature selection method (EMSFS). Unlike traditional methods that construct a graph to mine the similarity structure among all samples, the proposed EMSFS adaptively learns a bipartite graph between training samples and generated anchors, effectively reducing the computation cost of graph construction. Moreover, we employ matrix transformation skillfully to avoid the inverse operation of high-order matrices. Therefore, the computational complexity of EMSFS is linear to the number of training samples n, enhancing its scalability on large-scale data. Furthermore, EMSFS alternates between bipartite graph learning and feature selection, improving the effectiveness of the learned graph and selected features. Extensive experiments demonstrate the effectiveness and the superiority of our proposed EMSFS.

Although achieving superior performance to the state-of-the-art methods, the proposed EMSFS can be further improved and generalized. First, although the proposed EMSFS can assign appropriate weights to different views, it cannot completely remove the adverse impacts of poor views. A straightforward solution is to use the learned weights $\{\alpha_v\}_{v=1}^{V}$ to quantify the contribution of each view and thus eliminate the views with small weights in the training process. Specifically, we can set a threshold α_{min} for the view weights, then the v-th view will be removed from the current model if $\alpha_v \leqslant \alpha_{min}$. However, this manner excessively emphasizes the distinctions among views but neglects the correlations between different views, resulting in performance degradation. Therefore, effectively selecting an optimal view subset remains an important direction for improvement. Moreover, EMSFS performs feature selection by imposing an $l_{2,1}$-norm regularization on feature projection matrices. Considering the advantage of $l_{2,0}$-norm regularization that can directly select the relevant features without the extra sorting feature procedure as well as the regularization parameter, we plan to extend EMSFS by replacing the $l_{2,1}$-norm regularization with $l_{2,0}$-norm regularization in the feature. Additionally, it can be considered to automatically generate anchor points for each dataset in the training phase to further enhance performance.

CRediT authorship contribution statement

Chenglong Zhang: Data curation, Formal analysis, Investigation, Methodology, Software, Visualization, Writing – original draft. **Bingbing Jiang:** Conceptualization, Investigation, Methodology, Supervision, Writing – review & editing. **Zidong Wang:** Writing – review & editing. **Jie Yang:** Writing – review & editing. **Yangfeng Lu:** Validation. **Xingyu Wu:** Writing – review & editing. **Weiguo Sheng:** Funding acquisition, Project administration.

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.

Acknowledgements

This work is supported in part by the “Pioneer” and “Leading Goose” R&D Program of Zhejiang Province, China (Grant No. 2023C01022), the National Natural Science Foundation of China (Grant No. 62006065), and the Zhejiang Provincial Natural Science Foundation of China (Grant No. LY22F030004).

References

[1] Jing Zhao, Xijiong Xie, Xin Xu, Shiliang Sun, Multi-view learning overview: recent progress and new challenges, Inf. Fusion 38 (2017) 43–54.
[2] Xiaodong Jia, Xiao-Yuan Jing, Xiaoke Zhu, Songcan Chen, Bo Du, Ziyun Cai, Zheyu He, Dong Yue, Semi-supervised multi-view deep discriminant representation learning, IEEE Trans. Pattern Anal. Mach. Intell. 43 (7) (2021) 2496–2509.
[3] Sally El Hajjar, Fadi Dornaika, Fahed Abdallah, Multi-view spectral clustering via constrained nonnegative embedding, Inf. Fusion 78 (2022) 209–217.
[4] Fei Wu, Xiao-Yuan Jing, Pengfei Wei, Chao Lan, Yimu Ji, Guo-Ping Jiang, Qinghua Huang, Semi-supervised multi-view graph convolutional networks with application to webpage classification, Inf. Sci. 591 (2022) 142–154.
[5] Yang Xiangfei, Li Chun-Na, Shao Yuanhai, Robust multi-view discriminant analysis with view-consistency, Inf. Sci. 596 (2022) 153–168.
[6] En Yu, Jiande Sun, Jing Li, Xiaojun Chang, Xian-Hua Han, Alexander G. Hauptmann, Adaptive semi-supervised feature selection for cross-modal retrieval, IEEE Trans. Multimed. 21 (5) (2018) 1276–1288.
[7] Peican Zhu, Xin Hou, Keke Tang, Yang Liu, Yin-Ping Zhao, Zhen Wang, Unsupervised feature selection through combining graph learning and $l_{2,0}$-norm constraint, Inf. Sci. 622 (2023) 68–82.
[8] Bingbing Jiang, Junhao Xiang, Xingyu Wu, Yadi Wang, Huanhuan Chen, Weiwei Cao, Weiguo Sheng, Robust multi-view learning via adaptive regression, Inf. Sci. 610 (2022) 916–937.
[9] Rui Zhang, Feiping Nie, Xuelong Li, Xian Wei, Feature selection with multi-view data: a survey, Inf. Fusion 50 (2019) 158–167.
[10] Najmeh Ziraki, Fadi Dornaika, Alireza Bosaghzadeh, Multiple-view flexible semi-supervised classification through consistent graph construction and label propagation, Neural Netw. 146 (2022) 174–180.
[11] Yadi Wang, Jun Wang, Neurodynamics-driven holistic approaches to semi-supervised feature selection, Neural Netw. 157 (2023) 377–386.
[12] Weichan Zhong, Xiaojun Chen, Feiping Nie, Joshua Zhexue Huang, Adaptive discriminant analysis for semi-supervised feature selection, Inf. Sci. 566 (2021) 178–194.

[13] Qiang Lin, Min Men, Liran Yang, Ping Zhong, A supervised multi-view feature selection method based on locally sparse regularization and block computing, Inf. Sci. 582 (2022) 146–166.

[14] Chenping Hou, Feiping Nie, Hong Tao, Dongyun Yi, Multi-view unsupervised feature selection with adaptive similarity and view weight, IEEE Trans. Knowl. Data Eng. 29 (9) (2017) 1998–2011.

[15] Saeedeh Bahrami, Fadi Dornaika, Alireza Bosaghzadeh, Joint auto-weighted graph fusion and scalable semi-supervised learning, Inf. Fusion 66 (2021) 213–228.

[16] Jingliu Lai, Hongmei Chen, Tianrui Li, Xiaoling Yang, Adaptive graph learning for semi-supervised feature selection with redundancy minimization, Inf. Sci. 609 (2022) 465–488.

[17] Zhigang Ma, Feiping Nie, Yi Yang, Jasper R.R. Uijlings, Nicu Sebe, Alexander G. Hauptmann, Discriminating joint feature analysis for multimedia data understanding, IEEE Trans. Multimed. 14 (6) (2012) 1662–1672.

[18] Dan Shi, Lei Zhu, Jingjing Li, Zhiyong Cheng, Zhenguang Liu, Binary label learning for semi-supervised feature selection, IEEE Trans. Knowl. Data Eng. 35 (3) (2023) 2299–2312.

[19] Chengrui Zhang, Lei Zhu, Dan Shi, Jiecai Zheng, Haibao Chen, Bo Yu, Semi-supervised feature selection with soft label learning, IEEE/CAA J. Autom. Sin. (2022).

[20] Xiaojun Chen, Renjie Chen, Qingyao Wu, Feiping Nie, Min Yang, Rui Mao, Semisupervised feature selection via structured manifold learning, IEEE Trans. Cybern. 52 (7) (2021) 5756–5766.

[21] Han Zhang, Maoguo Gong, Feiping Nie, Xuelong Li, Unified dual-label semi-supervised learning with top-k feature selection, Neurocomputing 501 (2022) 875–888.

[22] Caijuan Shi, Qiuqi Ruan, Gaoyun An, Chao Ge, Semi-supervised sparse feature selection based on multi-view Laplacian regularization, Image Vis. Comput. 41 (2015) 1–10.

[23] Yangxi Li, Xin Shi, Cuilan Du, Yang Liu, Yonggang Wen, Manifold regularized multi-view feature selection for social image annotation, Neurocomputing 204 (2016) 135–141.

[24] Caijuan Shi, Gaoyun An, Ruizhen Zhao, Qiuiqi Ruan, Qi Tian, Multiview Hessian semisupervised sparse feature selection for multimedia analysis, IEEE Trans. Circuits Syst. Video Technol. 27 (9) (2016) 1947–1961.

[25] Caijuan Shi, Zhibin Gu, Changyu Duan, Qi Tian, Multi-view adaptive semi-supervised feature selection with the self-paced learning, Signal Process. 168 (2020) 107332.

[26] Bingbing Jiang, Xingyu Wu, Xiren Zhou, Yi Liu, Anthony G. Cohn, Weiguo Sheng, Huanhuan Chen, Semi-supervised multiview feature selection with adaptive graph learning, IEEE Trans. Neural Netw. Learn. Syst. (2022).

[27] Feiping Nie, Heng Huang, Xiao Cai, Chris H. Ding, Efficient and robust feature selection via joint $l_{2,1}$-norms minimization, in: Proceedings of Advances in Neural Information Processing Systems, 2010, pp. 1813–1821.

[28] Xiaojun Chen, Guowen Yuan, Feiping Nie, Zhong Ming, Semi-supervised feature selection via sparse rescaled linear square regression, IEEE Trans. Knowl. Data Eng. 32 (1) (2020) 165–176.

[29] Razieh Sheikhpour, Mehdi Agha Sarram, Sajjad Gharaghani, Mohammad Ali Zare Chahooki, A robust graph-based semi-supervised sparse feature selection method, Inf. Sci. 531 (2020) 13–30.

[30] Dan Shi, Lei Zhu, Jingjing Li, Zheng Zhang, Xiaojun Chang, Unsupervised adaptive feature selection with binary hashing, IEEE Trans. Image Process. 32 (2023) 838–853.

[31] Razieh Sheikhpour, Mehdi Agha Sarram, Sajjad Gharaghani, Mohammad Ali Zare Chahooki, A survey on semi-supervised feature selection methods, Pattern Recognit. 64 (2017) 141–158.

[32] M. Kumar, Benjamin Packer, Daphne Koller, Self-paced learning for latent variable models, in: Proceedings of Advances in Neural Information Processing Systems, 2010, pp. 1189–1197.

[33] Hebing Nie, Qun Wu, Haifeng Zhao, Weiping Ding, Muhammet Deveci, Flexible adaptive graph embedding for semi-supervised dimension reduction, Inf. Fusion (2023) 101872.

[34] Feiping Nie, Xiaoqian Wang, Heng Huang, Clustering and projected clustering with adaptive neighbors, in: Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining, 2014, pp. 977–986.

[35] Fang He, Feiping Nie, Rong Wang, Xuelong Li, Weimin Jia, Fast semisupervised learning with bipartite graph for large-scale data, IEEE Trans. Neural Netw. Learn. Syst. 31 (2) (2019) 626–638.

[36] Fang He, Feiping Nie, Rong Wang, Haojie Hu, Weimin Jia, Xuelong Li, Fast semi-supervised learning with optimal bipartite graph, IEEE Trans. Knowl. Data Eng. 33 (9) (2020) 3245–3257.

[37] Bin Zhang, Qianyao Qiang, Fei Wang, Feiping Nie, Fast multi-view semi-supervised learning with learned graph, IEEE Trans. Knowl. Data Eng. 34 (1) (2022) 286–299.

[38] Hong Chen, Feiping Nie, Rong Wang, Xuelong Li, Fast unsupervised feature selection with bipartite graph and $l_{2,0}$-norm constraint, IEEE Trans. Knowl. Data Eng. 35 (5) (2023) 4781–4793.

[39] Han Zhang, Danyang Wu, Feiping Nie, Rong Wang, Xuelong Li, Multilevel projections with adaptive neighbor graph for unsupervised multi-view feature selection, Inf. Fusion 70 (2021) 129–140.

[40] Dimitri P. Bertsekas, Constrained Optimization and Lagrange Multiplier Methods, Academic Press, 2014.

[41] Guangcan Liu, Zhouchen Lin, Shuicheng Yan, Ju Sun, Yong Yu, Yi Ma, Robust recovery of subspace structures by low-rank representation, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2012) 171–184.

[42] Fan R.K. Chung, Spectral Graph Theory, vol. 92, American Mathematical Soc., 1997.

[43] Yongshan Zhang, Jia Wu, Zhihua Cai, Philip S. Yu, Multi-view multi-label learning with sparse feature selection for image annotation, IEEE Trans. Multimed. 22 (11) (2020) 2844–2857.

[44] Janez Demšar, Statistical comparisons of classifiers over multiple data sets, J. Mach. Learn. Res. 7 (1) (2006) 1–30.

[45] Laurens Van der Maaten, Geoffrey Hinton, Visualizing data using t-sne, J. Mach. Learn. Res. 9 (11) (2008) 2579–2605.