Auto-weighted Multi-view Feature Selection with Graph Optimization

Qi Wang, *Senior Member, IEEE,* Xu Jiang, Mulin Chen and Xuelong Li, *Fellow, IEEE*

***Abstract*—In this paper, we focus on the unsupervised multi-view feature selection which tries to handle high dimensional data in the field of multi-view learning. Although some graph-based methods have achieved satisfactory performance, they ignore the underlying data structure across different views. Besides, their pre-defined laplacian graphs are sensitive to the noises in the original data space, and fail to get the optimal neighbor assignment. To address the above problems, we propose a novel unsupervised multi-view feature selection model based on graph learning, and the contributions are threefold: (1) during the feature selection procedure, the consensus similarity graph shared by different views is learned. Therefore, the proposed model can reveal the data relationship from the feature subset. (2) a reasonable rank constraint is added to optimize the similarity matrix to obtain more accurate information; (3) an auto-weighted framework is presented to assign view weights adaptively, and an effective alternative iterative algorithm is proposed to optimize the problem. Experiments on various datasets demonstrate the superiority of the proposed method compared with the state-of-the-art methods.**

***Index Terms*—Machine learning, multi-view learning, unsupervised feature selection, adaptive view weight, optimal similarity matrix.**

I. Introduction

THE data representation of sample largely determines the performance of learning tasks. With the development of the information acquisition and storage technology, massive data is emerging. Multi-view data which contains heterogeneous features has shown greater superiority than the traditional single-view data, and it has received increasing attention in many scientific fields, e.g., coherent groups detection [1], [2], event detection [3], image annotation [4], [5], recognition and retrieval tasks [6], [7], [8], [9] and image clustering/classification tasks [10], [11], [12], [13], [14]. In some practical problems, an object can be described from many different aspects, which can be considered as its multiple view features. For example, in many applications, various visual features are extracted for image representation. The features from each view capture specific information of the image. Meanwhile, the underlying data structure is always shared by different views. Benefit from the diversity and consistency of different views, more useful and comprehensive information can be obtained, and it will further contribute to a better performance.

However, each view of the multi-view dataset is always with high dimensionality, which could lead to high computing cost, sparse data samples and so on. Moreover, the redundant features and noises contained in the original data space would also influence the final performance. As an efficient technique to alleviate these troubles, feature selection has attracted great attention in multi-view learning [15]. It aims to learn a compact and representative subset of original features, where the valuable information is preserved. Since the labels are always difficult to obtain, we are committed to the research of unsupervised multi-view feature selection in this paper. Generally, there are two different solutions for this task.

The first way is to apply the traditional single-view feature selection approaches on multi-view data. Typically, the traditional single-view feature selection approaches include Structured Graph Optimization (SOGFS) [16], Joint Embedding Learning and Sparse Regression (JELSR) [17], Robust Spectral Feature Selection (RSFS) [18], Multi-Cluster Feature Selection [19], Laplacian Score (LapScor) [20], and so on. These approaches rank features using different strategies, and are able to find the significant features in many applications. The traditional single-view approaches are specialized for single-view data. Therefore, it is not appropriate to apply them to multi-view data due to the lack of attention for potential relationship between different views.

The second way is to use the methods designed specifically for multi-view data, where the underlying data structure across different views is valued. For example, Robust Multi-view Feature Selection (RMFS) [21] and Discriminatively Embedded K-Means (DEKM) [22] utilize the global structure to guide the multi-view subspace learning. Besides, owing to the powerful ability of graph learning [23], there are many graph-based approaches trying to preserve the local structure. Adaptive Unsupervised Multi-view Feature Selection (AUMFS) [8] utilizes several vital information in a unified framework. Adaptive Multi-view Feature Selection (AMFS) [7] establishes a general trace ratio optimization model for human motion retrieval. AMFS and AUMFS characterize the local data structure by means of simply connecting pre-defined laplacian graph of each view. Similarly, Cluster Structure Preserving Unsupervised Feature Selection (CSP-UFS) [24] construct the laplacian graph by adopting the same strategy as AMFS and AUMFS, but the difference is that it utilizes the discriminant analysis to preserve the cluster structure.

However, the original high-dimensional dataset always contains noise features and outliers. Importantly, the relationship

This work was supported by the National Natural Science Foundation of China under Grant U1864204, 61773316, U1801262, and 61871470.

Q. Wang, X. Jiang, M. Chen and X. Li are with the School of Computer Science and School of Artificial Intelligence, Optics and Electronics (iOPEN), Northwestern Polytechnical University, Xi'an 710072, P.R. China (e-mail: crabwq@gmail.com; jx19961023@mail.nwpu.edu.cn; chenmulin@mail.nwpu.edu.cn; xuelong_li@nwpu.edu.cn) (Corresponding author: Xuelong Li).

between samples learned from the high-dimensional space can not capture its intrinsic characteristic [25]. Therefore, the pre-defined similarity graph is probably to be unreliable. Moreover, in multi-view data, different views explore the original data from different perspectives, and the real cluster structure exists in every view actually. Therefore, benefit from the consistency of different views, a unified similarity matrix can give more valuable and accurate information, and further contribute to a better performance. If the number of connected components equals to the number of clusters exactly, the similarity graph will have the ideal neighbor assignment. In clustering task, without further processing, a similarity graph with the ideal neighbor assignment can get the final clustering result directly. In the feature selection task, the guidance of a similarity graph is crucial to the process of feature selection. Accordingly, if the similarity graph has the ideal neighbor assignment, more accurate information will be obtained, and more valuable features will be selected. To address these issues, an unsupervised multi-view feature selection method named as Multi-view Feature Selection with Graph Learning (MFSGL) is proposed. We highlight the main contributions of the paper as follows:

1) MFSGL learns an optimal similarity graph for all views, which indicates the cluster structure. A reasonable constraint is added to the similarity matrix, which make the number of connected components equals to the number of clusters exactly, and each connected component corresponds to one cluster.
2) MFSGL carrys out the multi-view feature selection and similarity graph learning simultaneously. Therefore, it is able to learn the local data structure adaptively and get more valuable information.
3) To balance the importance of different views, an efficient weight assignment strategy is proposed. The weight of each view can be determined more concisely and effectively.

The remainder of the paper is summarized here. Section II reviews the related works on multi-view feature selection task. Section III introduces the details of MFSGL. An effective optimization algorithm for this problem is also provided in Selection III. In Section IV, experiments on different datasets are conducted to validate the effectiveness of MFSGL, followed by the conclusion and future work in Section V.

II. RELATED WORK

Since MFSGL constructs the similarity graph to reveal the data cluster structure, in this section, we introduce several representative related works based on similarity graph constructing briefly

A. AMFS

Adaptive Multi-view Feature Selection (AMFS) is proposed for generating efficient motion data representation. Firstly, for preserving the local geometric structure, AMFS uses a local regression model and neighbor similarity weights to learn the laplacian graph of each view automatically. Secondly, these graphs are connected by non-negative view weights to explore the correlation of different views. Thirdly, to capture the global information, the final feature selection framework is formulated in a trace ratio form motivated by PCA. The final objective function is shown as:

$$\begin{aligned} \min_{W,\alpha} \; & \frac{tr(W^T X \sum_{v=1}^{V} \alpha_v^r L^{(v)} X^T W)}{tr(W^T X H_X X^T W)} \\ s.t. \; & \sum_{v=1}^{V} \alpha_v = 1, \; \alpha_v \geq 0, \; W \in \{0,1\}^{D \times d}, \end{aligned} \tag{1}$$

where $\alpha_1, \alpha_2, \alpha_3, \cdots, \alpha_V$ are the non-negative view weights to combine all views, and r is used to prevent that only the best view is selected with its α_v equals to 1. $L^{(v)}$ denotes the pre-defined laplacian graph of v-th view. W is the feature selection matrix that picks out most representative feature elements from original high-dimensional data. There is only one non-zero element in each row of W, and the desired features will be identified according to the non-zero elements, so $W^T X$ denotes the compact feature subset. D is the original feature number and is larger than d. H is the centralized matrix.

B. AUMFS

Another method is Adaptive Unsupervised Multi-view Feature Selection (AUMFS) which is applied to several recognition tasks. It tries to simultaneously use the following key information: the local structure in data space, the similarity of different samples and the correlation of all views. Specifically, the authors attempt to obtain the pseudo cluster labels by a robust loss function based on regression model. It is necessary to consider the geometric structure hidden in the original data space, so the similarity graphs of each view are constructed. What's more, to explore the underlying complemental information between different views, all views are connected using the non-negative view weights $\lambda = [\lambda_1, \lambda_2, \cdots, \lambda_V]^T$. AUMFS is finally formulated in the following form:

$$\begin{aligned} & \min_{F,\lambda,W} \; tr(F^T \sum_{v=1}^{V} \lambda_v^r L^{(v)} F) + \alpha \|X^T W - F\|_{2,1} + \beta \|W\|_{2,1} \\ & s.t. \; F^T F = I_c, F \geq 0, \sum_{v=1}^{V} \lambda_v = 1, \lambda_v \geq 0. \end{aligned} \tag{2}$$

The first term is used to obtain the data cluster labels and the last two terms make up the robust sparse regression model for feature selection. F denotes the predicted cluster label matrix. $F^T F = I_c$ is an orthogonal constraint on F. $\|W\|_{2,1}$ is the feature selection matrix with the $\ell_{2,1}$-norm regularization. $X^T W$ can be regarded as the low-dimensional representation based on the most valued feature information. λ_v is the view weight vector. α and β are used to balance the contributions of last two items. We can solve the problem by a simple efficient iterative method.

C. MVFS

Unsupervised Feature Selection for Multi-view Data (MVFS) [26] tries to explore the structural information existing in different views by utilizing pseudo cluster labels and minimizing the regression loss, which is similar to AUMFS. But it has two differences between MVFS and AUMFS. MVFS pre-defines the view weights which are fixed in the following iterations, and the another one is that MVFS uses the Frobenius norm in the regression term instead of the $l_{2,1}$-norm.

$$\begin{aligned}&\min_{W,Z} \sum_{i=1}^{m} \lambda_i (Tr(Z^T L_i Z) + \alpha(\|X_i^T W_i - Z\|_F^2 + \beta\|W_i\|_{2,1})\\ &s.t.\ Z^T Z = I, Z \geq 0.\end{aligned} \tag{3}$$

Here, m is the view number. Z denotes the pseudo label matrix. L_i is the i-th laplacian matrix. $\|W_i\|_{2,1}$ controls the capacity and sparsity of the projection matrix W_i. The parameter λ_i denotes the i-th view weight. α and β are two balanced parameters.

D. CSP-UFS

Cluster Structure Preserving Unsupervised Feature Selection (CSP-UFS) is proposed by Shi et al.. CSP-UFS tries to improve the ability of distinguishing different categories exactly after feature selection. The authors stress that the labels contain valuable information, so CSP-UFS uses spectral clustering to get the labels firstly. And then, the discriminant analysis is adopted to conduct the feature selection process while preserving the data cluster structure. Because the information of different views is complementary and reinforce each other, CSP-UFS imposes a non-negative weight on each view to connect all views, and the better view would get a larger weight. Therefore, CSP-UFS employs the spectral clustering and the discriminant analysis simultaneously for unsupervised feature selection.

$$\begin{aligned}\min_{W,F,\alpha}\ & \frac{Tr(W^T X(I - FF^T)X^T W)}{Tr(W^T X X^T W)}\\ & + \lambda_1 \sum_{v=1}^{V} \alpha_v^r Tr(F^T L^{(v)} F) + \lambda_2 \|W\|_{2,1} \qquad (4)\\ & s.t.\ FF^T = I_c, F \geq 0, \sum_{v=1}^{V} \alpha_v = 1, \alpha_v \geq 0.\end{aligned}$$

The first term is the feature selection procedure using discriminative analysis, and the last term ensures the sparsity of projection matrix $W \in R^{d \times q}$. The pseudo label matrix F can be learned in the second term. Similar to the above methods, $L^{(v)}$ and α_v denote the v-th pre-defined laplacian matrix and view weight respectively. This problem can be solved by an alternating optimization algorithm.

E. ASVW

Obviously, all of the above methods pre-define the fixed laplacian graph of each view, and connect all views using a non-negative view weight vector. Different from them, Adaptive Similarity and View Weight (ASVW) [27] learns a similarity graph shared by different data views adaptively.

$$\begin{aligned}&\min \mathcal{L}(W_1, \cdots, W_V, \alpha, S) =\\ &\quad \sum_{v=1}^{V}\sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_v^{r_1} \|W_v^T x_i^{(v)} - W_v^T x_j^{(v)}\|^2 (s_{ij})^{r_2}\\ &\quad + \lambda \sum_{v=1}^{V} \|W_v\|_{2,p}^p \qquad (5)\\ &s.t.\ W_v^T W_v = I, \sum_{v=1}^{V} \alpha_v = 1, \alpha_v \geq 0, \sum_{j=1}^{n} s_{ij} = 1,\\ &\quad s_{ij} \geq 0, \|s_i\|_0 = k.\end{aligned}$$

Here, S and α_v denote the similarity matrix and v-th view weight respectively. r_1 and r_2 are two parameters to avoid the trivial solution. The $\ell_{2,p}$-norm is used to ensure the sparsity of projection matrix W_v. In the first term, the common similarity matrix is learned adaptively. However, the similarity matrix learned by ASVW fails to have an ideal neighbor assignment, and the clustering result can not be obtained directly from it.

III. Methodology

In this section, we first formulate the objective function of MFSGL, and then an efficient alternative iterative algorithm is introduced to solve it.

For better introducing the proposed method, all the notations are summarized in Table I.

TABLE I
Descriptions of all notations.

notations	descriptions
n	the number of samples
c	the number of clusters
V	the number of views
s	the number of selected features
d_v	the dimension in the v-th view
m_v	the projection dimension in the v-th view
$\alpha^{(v)}$	the v-th view weight
$x_i^{(v)} \in \mathbb{R}^{d_v}$	the i-th sample in the v-th view
$X^{(v)} \in \mathbb{R}^{d_v \times n}$	the v-th view of the multi-view data
$W_v \in \mathbb{R}^{d_v \times m_v}$	the projection matrix of the v-th view
S	the similarity matrix
F	the cluster indicator matrix
$Tr(\cdot)$	the trace of a matrix
$\|\cdot\|_F$	Frobenius norm
$\|\cdot\|_2$	ℓ_2-norm of a vector
$\|\cdot\|_{2,1}$	$\ell_{2,1}$-norm
γ, λ, μ	the regularization parameters

A. Multi-View Feature Selection with Graph Learning

The local structure is pretty conspicuous for its information discovery capability, and it is believed better than the global structure. Therefore, lots of unsupervised feature selection methods explore the local structure information and preserve it after projection. The traditional methods always learn the similarity graph of each view in advance. However, as mentioned above, the pre-defined similarity graph may be unreliable. In fact, to efficiently capture the structure of data space, we propose to learn the graph and the feature subset simultaneously. In the data space, the closer the two data points are, the larger similarity between them should be. In this paper, the square of Euclidean distance is used as the measurement of the similarity between samples. To evaluate the effectiveness of different views, a reasonable view weight assignment method is also required.

Based on above discussion, we firstly introduce the following unsupervised multi-view feature selection framework:

$$\begin{aligned}\min_{W_v,S}\ &\sum_{v=1}^{V}((\sum_{i,j}\|W_v^T x_i^{(v)} - W_v^T x_j^{(v)}\|_2^2 s_{ij})^{\frac{p}{2}} + \gamma\|W_v\|_{2,1})\\ &s.t.\ \forall i, s_i^T\mathbf{1} = 1, 0 \leq s_{ij} \leq 1, W_v^T W_v = I.\end{aligned} \tag{6}$$

Here, for v-th view, the projection matrix is denoted as $W_v \in \mathbb{R}^{d_v \times m_v}$, and it projects the original d_v-dimensional data space into the latent m_v-dimensional subspace, where m_v less than d_v definitely. Moreover, in experiments, m_v always needs to be tuned to get the best result [16]. γ is the regularization parameter. The $\ell_{2,1}$-norm regularization on W makes it row sparse to select more valuable features [28], [29]. The constraint $W_v^T W_v = I$ makes the feature space distinctive after reduction [30]. For similarity matrix $S \in \mathbb{R}^{n\times n}$, the element s_{ij} denotes the similarity between *i*-th sample and *j*-th sample, and the transpose of the *i*-th row is denoted as s_i. We denote the column vector whose all elements are 1 as $\mathbf{1}$. $\|W_v^T x_i^{(v)} - W_v^T x_j^{(v)}\|_2^2$ denotes the distance between two samples in *v*-th view after reduction. With different parameter p ($0 < p \leq 2$), different exponential functions are obtained in Eq. (6). Thanks to the exponential function, the weight of each view can be assigned automatically which will be presented in the following part.

Denote the laplacian matrix as $L_S = D - \frac{S^T+S}{2}$, and the elements of diagonal matrix D are set as $\sum_j \frac{(s_{ij}+s_{ji})}{2}$. It can be proved that Eq. (6) is equivalent to

$$\begin{aligned}\min_{W_v,S}\ &\sum_{v=1}^{V}(\alpha_v Tr(W_v^T (X^{(v)})^T L_S X^{(v)} W_v) + \gamma\|W_v\|_{2,1})\\ &s.t.\ \forall i, s_i^T\mathbf{1} = 1, 0 \leq s_{ij} \leq 1, W_v^T W_v = I,\end{aligned} \tag{7}$$

where

$$\alpha_v = \frac{p}{2Tr(W_v^T (X^{(v)})^T L_S X^{(v)} W_v)^{\frac{2-p}{2}}}. \tag{8}$$

Proof: The Eq. (6) can be rewritten as

$$\begin{aligned}\min_{W_v,S}\ &\sum_{v=1}^{V}(Tr(W_v^T (X^{(v)})^T L_S X^{(v)} W_v)^{\frac{p}{2}} + \gamma\|W_v\|_{2,1})\\ &s.t.\ \forall i, s_i^T\mathbf{1} = 1, 0 \leq s_{ij} \leq 1, W_v^T W_v = I.\end{aligned} \tag{9}$$

We denote the Lagrangian multiplier as Λ and the Lagrangian function of problem (9) should be

$$\begin{aligned}&\sum_{v=1}^{V}(Tr(W_v^T (X^{(v)})^T L_S X^{(v)} W_v)^{\frac{p}{2}} + \gamma\|W_v\|_{2,1}\\ &+ \mathcal{G}(\Lambda, W_v, S)).\end{aligned} \tag{10}$$

Then, let the derivative of Eq. (10) w.r.t W_v be zero, and we get

$$\begin{aligned}&\sum_{v=1}^{V}(\alpha_v \frac{\partial Tr(W_v^T (X^{(v)})^T L_S X^{(v)} W_v)}{\partial W_v} + \gamma\frac{\partial\|W_v\|_{2,1}}{\partial W_v} +\\ &\frac{\partial \mathcal{G}(\Lambda, W_v, S)}{\partial W_v}) = 0\end{aligned} \tag{11}$$

and Eq. (8).

If α_v is set to be stationary, solving Eq. (11) is equivalent to solving Eq. (7). Therefore, the solution of Eq. (6) is transformed into the alternative iterative solution of Eq. (7) and Eq. (8). If the feature set of v-th view is more compact and efficient, $\sum_{i,j}\|W_v^T x_i^{(v)} - W_v^T x_j^{(v)}\|_2^2 s_{ij}$ ought to be smaller, which brings a larger v-th view weight α_v based on Eq. (8). Accordingly, a worse view will be assigned a smaller weight. Therefore, the introduced framework is able to optimize the weights adaptively.

If the number of connected components in a similarity matrix equals to c, the cluster structure can be discovered clearly, and it is conducive to the follow-up treatment obviously in clustering tasks. However, the similarity matrix learned by problem (6) fails to have this property [31]. Fortunately, if the rank of L_S is equivalent to $n - c$, this problem will be solved [32], [33]. L_S is positive semi-definite, that is to say, $\sigma_i(L_S) \geq 0$ where $\sigma_i(L_S)$ denotes the i-th smallest eigenvalue of L_S. It is able to be proved that $rank(L_S) = n - c$ brings $\sum_{i=1}^{c}\sigma_i(L_S) = 0$. And then, based on Ky Fans Theorem [34], the following equation is obtained:

$$\sum_{i=1}^{c}\sigma_i(L_S) = \min_{F^TF=I, F\in\mathbb{R}^{n\times c}} Tr(F^T L_S F), \tag{12}$$

where F denotes the cluster indicator matrix. Moreover, to avoid the trivial solution, the constraint $\mu\sum_{i,j}s_{ij}^2$ should be added where μ is a regularization parameter [35]. Otherwise, the optimal solution should be that the similarity between the closest two samples is 1, and the others are 0.

By combining all of the above constraints, we finally summarize the Unsupervised Multi-view Feature Selection with Graph Learning (MFSGL) framework as

$$\begin{aligned}\min_{W_v,F,S}\ &\sum_{v=1}^{V}(Tr(W_v^T (X^{(v)})^T L_S X^{(v)} W_v)^{\frac{p}{2}} + \gamma\|W_v\|_{2,1})\\ &+ \mu\sum_{i,j}s_{ij}^2 + 2\lambda Tr(F^T L_S F)\\ &s.t.\ \forall i, s_i^T\mathbf{1} = 1, 0 \leq s_{ij} \leq 1, F^TF = I, F \in \mathbb{R}^{n\times c}\\ &\quad W_v^T W_v = I.\end{aligned} \tag{13}$$

Here, the regularization parameter λ is used to guarantee the existence of ideal neighbor assignment. The value of λ can be determined adaptively during iteration. If there are more than c connected components in S, λ should be decreased. Otherwise, λ should be increased. In summary, we integrate the feature selection and similarity graph learning, and the similarity matrix is constrained to have exactly c connected components by the last term. Therefore, the proposed method

MFSGL can learn an optimal similarity graph and a reliable projection matrix.

After deriving the optimal solution, $\|w_{vi}\|_2$ is adopted to evaluate the importance of each feature of all views, where w_{vi} denotes the i-th row of W_v. And the s features we want to select are the top s features based on $\|w_{vi}\|_2$.

B. Optimization Algorithm

From section III-A, it is easy to know that the solution of problem (13) can be transformed into the alternative iterative solution of the following problem and Eq. (8).

$$\begin{aligned}\min_{W_v,F,S} & \sum_{v=1}^{V}(\alpha_v Tr(W_v^T(X^{(v)})^T L_S X^{(v)} W_v) + \gamma\|W_v\|_{2,1}) \\ & + \mu\sum_{i,j} s_{ij}^2 + 2\lambda Tr(F^T L_S F) \\ s.t.\ & \forall i, s_i^T\mathbf{1} = 1, 0 \le s_{ij} \le 1, F^T F = I, F \in \mathbb{R}^{n\times c} \\ & W_v^T W_v = I.\end{aligned} \tag{14}$$

With fixed α_v, the solution of problem (14) is the same as the solution of problem (13), and we can get F, W_v and S by solving problem (14). Then, according to the newly obtained W_v and S, α_v is able to be updated by Eq. (8).

Here, an efficient alternative iterative algorithm is given to get the solution of problem (14).

1) Update W_v with Fixed α_v , F and S: If α_v, F and S are fixed, we can rewrite the problem (14) as the following problem for each view:

$$\min_{W_v^T W_v=1} \alpha_v Tr(W_v^T(X^{(v)})^T L_S X^{(v)} W_v) + \gamma\|W_v\|_{2,1}, \tag{15}$$

where $\|W_v\|_{2,1} = \sum_i \|w_{vi}\|_2$. Apparently, w_{vi} is theoretically possible to be zero, which will cause Eq. (15) non-differentiable. So we set ε as a very small constant and replace $\|w_{vi}\|_2$ with $\sqrt{{w_{vi}}^T w_{vi} + \varepsilon}$. Then, problem (15) is converted into

$$\begin{aligned}\min_{W_v^T W_v=1} & \alpha_v Tr(W_v^T(X^{(v)})^T L_S X^{(v)} W_v) \\ & + \gamma\sum_i \sqrt{w_{vi}^T w_{vi} + \varepsilon}.\end{aligned} \tag{16}$$

If ε approaches zero infinitely, the problem (15) is the same as problem (16).

We denote the Lagrangian multiplier as Λ and write the Lagrangian function of problem (16) as

$$\begin{aligned}\mathcal{L}(W_v,\Lambda) &= \alpha_v Tr(W_v^T(X^{(v)})^T L_S X^{(v)} W_v) \\ &+ \gamma\sum_i \sqrt{w_{vi}^T w_{vi} + \varepsilon} + Tr(\Lambda(W_v^T W_v - 1)),\end{aligned} \tag{17}$$

Then, take the derivative of Eq. (17) on W_v and let it be zero:

$$\frac{\partial\mathcal{L}(W_v,\Lambda)}{\partial W_v} = \alpha_v (X^{(v)})^T L_S X^{(v)} W_v + \gamma G W_v + W_v\Lambda = 0, \tag{18}$$

where $G \in \mathbb{R}^{d_v\times d_v}$ is a diagonal matrix whose diagonal entries are defined as

$$G_{ii} = \frac{1}{2\sqrt{w_{vi}^T w_{vi} + \varepsilon}}. \tag{19}$$

With fixed W_v, G is obtained by Eq. (19). And with fixed G, the solution of Eq. (18) can be obtained by solving

$$\min_{W_v^T W_v=1} Tr(W_v^T(X^{(v)})^T L_S X^{(v)} W_v) + \frac{\gamma}{\alpha_v} Tr(W_v^T G W_v). \tag{20}$$

It is easy to know that the m_v column vectors of the optimal W_v are the m_v eigenvectors of $((X^{(v)})^T L_S X^{(v)} + \frac{\gamma}{\alpha_v} G)$, which correspond to the m_v smallest eigenvalues. The details of deriving the solution of W_v is summarized in Algorithm 1, and the KKT condition is satisfied. In section III-B4, we will give its convergence proof.

2) Update S with Fixed α_v, F and W_v: If α_v, F and W_v are fixed, we can rewrite the problem (14) as the following problem:

$$\begin{aligned}\min_S & \sum_{v=1}^{V}(\alpha_v Tr(W_v^T(X^{(v)})^T L_S X^{(v)} W_v) + \mu\sum_{i,j} s_{ij}^2 + \\ & 2\lambda Tr(F^T L_S F) \\ & s.t.\ \forall i, s_i^T\mathbf{1} = 1, 0 \le s_{ij} \le 1.\end{aligned} \tag{21}$$

In spectral analysis [36],

$$\sum_{i,j} \|f_i - f_j\|_2^2 s_{ij} = 2Tr(F^T L_S F). \tag{22}$$

So problem (21) can be rewritten as

$$\begin{aligned}\min_S & \sum_{i,j}(\sum_{v=1}^{V} \alpha_v \|W_v^T x_i^{(v)} - W_v^T x_j^{(v)}\|_2^2 s_{ij} + \mu s_{ij}^2) \\ & + \lambda\sum_{i,j}\|f_i - f_j\|_2^2 s_{ij} \\ & s.t.\ \forall i, s_i^T\mathbf{1} = 1, 0 \le s_{ij} \le 1.\end{aligned} \tag{23}$$

Because the row vectors of similarity matrix are independent with each other, problem (23) can be divided into n subproblems, and each subproblem aims to acquire the similarity vector of the corresponding sample. Here, we take the i-th sample as an example.

$$\begin{aligned}\min_{s_i^T\mathbf{1}=1, 0\le s_{ij}\le 1} & \sum_j(\sum_{v=1}^{V} \alpha_v \|W_v^T x_i^{(v)} - W_v^T x_j^{(v)}\|_2^2 s_{ij} \\ & + \mu s_{ij}^2) + \lambda\sum_j \|f_i - f_j\|_2^2 s_{ij}.\end{aligned} \tag{24}$$

Then, set $p_{ij} = \sum_v \alpha_v \|W_v^T X_i^{(v)} - W_v^T X_j^{(v)}\|_2^2$, $q_{ij} = \|f_i - f_j\|_2^2$ and $t_i \in \mathbb{R}^{n\times 1}$ whose element is $t_{ij} = p_{ij} + \lambda q_{ij}$. Therefore, problem (24) can be transformed into

$$\min_{s_i^T\mathbf{1}=1, 0\le s_{ij}\le 1} \|s_i + \frac{t_i}{2\mu}\|_2^2. \tag{25}$$

The solution of this problem and the method to optimize parameter μ will be shown later.

3) Update F with Fixed α_v, W_v and S: If S, α_v and W_v are fixed, problem (14) becomes:

$$\min_{F^TF=I, F\in\mathbb{R}^{n\times c}} \lambda Tr(F^T L_S F). \tag{26}$$

After acquiring the c smallest eigenvalues of L_S, we can obtain the optimal solution of F which is made up of c corresponding eigenvectors.

4) Convergence Proof of Algorithm 1: To prove the convergence of Algorithm 1, the lemma proposed by [37] is needed, which is described as follows.

Lemma 1: *The following inequality holds for any positive real number u and v.*

$$\sqrt{u} - \frac{u}{2\sqrt{v}} \leq \sqrt{v} - \frac{v}{2\sqrt{v}}. \tag{27}$$

The converged solution of problem (16) can be derived by Algorithm 1, and the convergence is proven by the following theorem.

Theorem 1: *In Algorithm 1, updated W_v will decrease the objective value of problem (16) until converge.*

Proof: Let $\widetilde{W}_v$ denotes the updated W_v. Therefore,

$$\begin{aligned} &Tr(\widetilde{W}_v^T (X^{(v)})^T L_S X^{(v)} \widetilde{W}_v) + \frac{\gamma}{\alpha_v} Tr(\widetilde{W}_v^T G \widetilde{W}_v) \\ \leq &Tr({W_v}^T (X^{(v)})^T L_S X^{(v)} W_v) + \frac{\gamma}{\alpha_v} Tr({W_v}^T G W_v). \end{aligned} \tag{28}$$

Then, by adding $\frac{\gamma}{\alpha_v}\sum_i \frac{\varepsilon}{2\sqrt{w_{vi}^T w_{vi}+\varepsilon}}$ to the both sides and substituting the definition of G shown in Eq. (19), the inequality (28) changes to

$$\begin{aligned} &Tr(\widetilde{W}_v^T (X^{(v)})^T L_S X^{(v)} \widetilde{W}_v) + \frac{\gamma}{\alpha_v} \sum_i \frac{\widetilde{w}_{vi}^T \widetilde{w}_{vi} + \varepsilon}{2\sqrt{w_{vi}^T w_{vi} + \varepsilon}} \\ \leq &Tr({W_v}^T (X^{(v)})^T L_S X^{(v)} W_v) + \frac{\gamma}{\alpha_v} \sum_i \frac{w_{vi}^T w_{vi} + \varepsilon}{2\sqrt{w_{vi}^T w_{vi} + \varepsilon}}. \end{aligned} \tag{29}$$

According to Lemma 1, it is easy to know

$$\begin{aligned} &\frac{\gamma}{\alpha_v}\sqrt{\widetilde{w}_{vi}^T \widetilde{w}_{vi} + \varepsilon} - \frac{\gamma}{\alpha_v} \sum_i \frac{\widetilde{w}_{vi}^T \widetilde{w}_{vi} + \varepsilon}{2\sqrt{w_{vi}^T w_{vi} + \varepsilon}} \\ \leq &\frac{\gamma}{\alpha_v}\sqrt{w_{vi}^T w_{vi} + \varepsilon} - \frac{\gamma}{\alpha_v} \sum_i \frac{w_{vi}^T w_{vi} + \varepsilon}{2\sqrt{w_{vi}^T w_{vi} + \varepsilon}}. \end{aligned} \tag{30}$$

Finally, by summing the inequality (29) and inequality (30), we get the following inequality and complete the proof.

$$\begin{aligned} &Tr(\widetilde{W}_v^T (X^{(v)})^T L_S X^{(v)} \widetilde{W}_v) + \frac{\gamma}{\alpha_v}\sqrt{\widetilde{w}_{vi}^T \widetilde{w}_{vi} + \varepsilon} \\ \leq &Tr({W_v}^T (X^{(v)})^T L_S X^{(v)} W_v) + \frac{\gamma}{\alpha_v}\sqrt{w_{vi}^T w_{vi} + \varepsilon}. \end{aligned} \tag{31}$$

5) Determination of μ: It has been noted that the parameter μ is used to avoid the trivial solution. Let us consider two extreme conditions of μ. When $\mu = 0$, the similarity between the closet two samples is 1, and the others are 0. When $\mu = \infty$, all similarities should equal to $\frac{1}{n}$. Therefore, μ is crucial to the number of sample neighbors.

```
**Algorithm 1** Algorithm to get projection matrices $W_v$
**Input:** The data matrix $\{X^{(1)}, X^{(2)}, \cdots, X^{(V)}\}$, $X^{(v)} \in \mathbb{R}^{d_v\times n}$, regularization parameter $\gamma$, the view weights $\alpha_v$, projection dimension $m_v$, laplacian matrix $L_S \in \mathbb{R}^{n\times n}$.
Initialize $G \in \mathbb{R}^{d_v\times d_v}$ as $G = I$.

For each view:
**Repeat**
1. Get the solution $W_v$ of problem (20).
2. Update $G$ via Eq. (19).
**Until** converge

**Output:**
Projection matrices $\{W_v \in \mathbb{R}^{d_v\times m_v}\}_{v=1}^V$
```

```
**Algorithm 2** Algorithm to solve MFSGL
**Input:** The multi-view dataset $\{X^{(1)}, X^{(2)}, \cdots, X^{(V)}\}$, $X^{(v)} \in \mathbb{R}^{d_v\times n}$, regularization parameter $\gamma$, number of selected features $s$, number of clusters $c$, projection dimension $m_v$, a large enough $\lambda$.
Initialize $\{\alpha_v\}_{v=1}^V = \frac{1}{V}$.
Initialize $S$ by solving
    $\min_S \sum_{i,j}(\sum_{v=1}^V \alpha_v \|x_i^{(v)} - x_j^{(v)}\|_2^2 s_{ij} + \mu s_{ij}^2)$
    $s.t. \forall i, s_i^T\mathbf{1} = 1, 0 \leq s_{ij} \leq 1.$

**Repeat**
1. Obtain the diagonal matrix $D$ and $L_S$ according to $D_{ii} = \sum_j \frac{s_{ij}+s_{ji}}{2}$ and $L_S = D - \frac{S^T+S}{2}$ respectively.
2. Update $W_v$ for each view by using Algorithm 1.
3. Obtain $F$ by solving problem (26).
4. Obtain each $s_i$ via solving problem (25).
5. Update the parameter $\mu$ according to Eq. (35).
6. Update the view weights $\alpha_v$ for each view according to Eq. (8).
**Until** converge

**Output:** Features sorted by $\|w_{vi}\|_2$ in descending order. The final feature subset which is made up of the top $s$ features.
```

If the neighbor number of a sample is set to be k, here we derive the optimal μ to ensure that there are k non-zero elements in most s_i.

θ and φ_i are denoted as the Lagrangian multipliers, and the Lagrangian function of problem (25) can be constructed as

$$\mathcal{L}(s_i, \theta, \varphi_i) = \frac{1}{2}\|s_i + \frac{t_i}{2\mu_i}\|_2^2 - \theta(s_i^T\mathbf{1} - 1) - \varphi_i^T s_i. \tag{32}$$

According to the KKT condition, the solution of this problem is obtained.

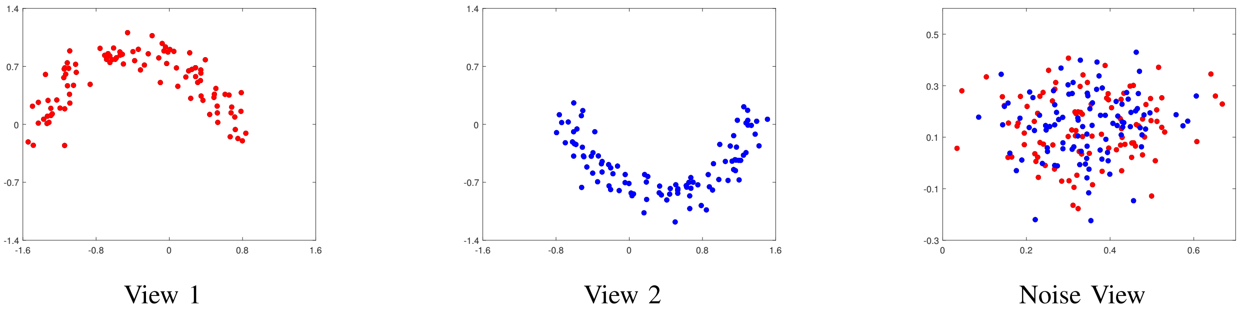
$$s_{ij} = (-\frac{t_i}{2\mu_i} + \theta)_+. \tag{33}$$

Here, $\theta = \frac{1}{k} + \frac{1}{2k\mu_i}\sum_{j=1}^k t_{ij}$ [31]. s_i should have k non-zero elements exactly, that is to say, $s_{i,k+1} \leq 0 < s_{i,k}$. Therefore, the μ_i should satisfy the following property:

$$\frac{k}{2}t_{i,k} - \frac{1}{2}\sum_{j=1}^k t_{ij} < \mu_i < \frac{k}{2}t_{i,k+1} - \frac{1}{2}\sum_{j=1}^k t_{ij}, \tag{34}$$

TABLE II
DETAILS OF THE MULTI-VIEW DATASETS USED IN OUR EXPERIMENTS.

view	*Outdoor-Scene*	*Caltech101-7*	*NUS-WIDE-OBJ*	*Handwritten*
1	GIST (512)	GABOR (48)	CH (64)	FAC (216)
2	CM (432)	WM (40)	CMT (225)	PIX (240)
3	HOG (256)	CENTRIST (254)	CORR (144)	ZER (47)
4	LBP (48)	HOG (1984)	EDH (73)	MOR (6)
5	-	GIST (512)	WT (128)	KAR (64)
6	-	LBP (928)	-	FOU (76)
Number of features	1622	3766	634	649
Number of samples	210	8677	3000	2000
Classes	7	7	25	10

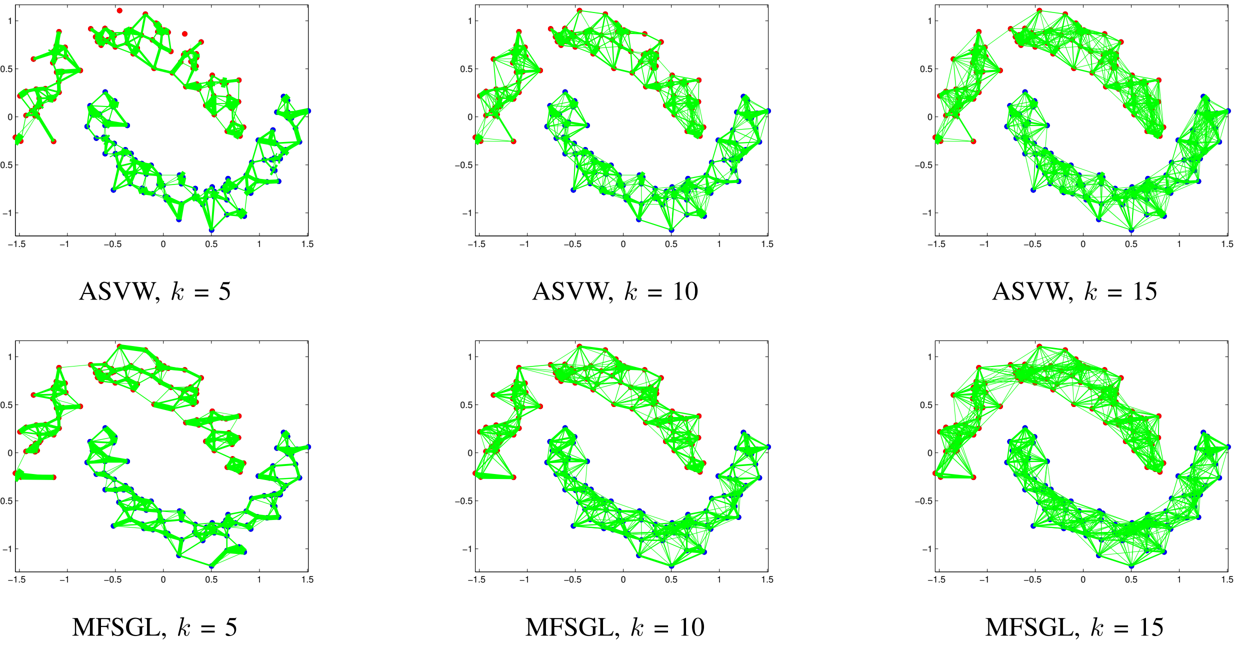


View 1

View 2

Noise View

Fig. 1. The synthetic multi-view two-moon dataset.



ASVW, k = 5

ASVW, k = 10

ASVW, k = 15

MFSGL, k = 5

MFSGL, k = 10

MFSGL, k = 15

Fig. 2. The graphs learned by MFSGL and ASVW on pure two-moon dataset.

and we can set a good enough μ as

$$\mu = \frac{1}{n}\sum_{i=1}^{n}\mu_i = \frac{1}{n}\sum_{i=1}^{n}(\frac{k}{2}t_{i,k+1} - \frac{1}{2}\sum_{j=1}^{k}t_{ij}), \qquad (35)$$

where $t_{i1}, t_{i2}, \cdots, t_{in}$ are sorted in ascending order.

In summary, we give an alternative iteration method to optimize the objective function of MFSGL. α_v can be modified according to Eq. (8). S, W_v and F can be modified by tackling problem (14). The details are summarized in Algorithm 2.

IV. EXPERIMENTS

Here, the effectiveness of MFSGL is demonstrated on both synthetic datasets and real-world datasets.

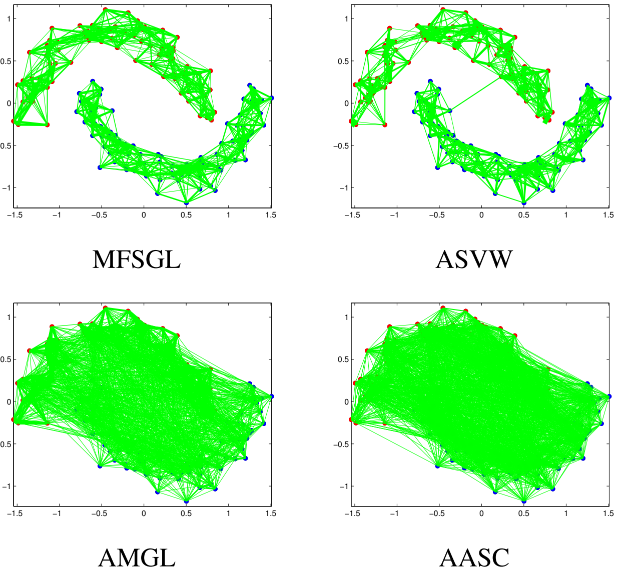
A. Experiments on the Synthetic Datasets

In this part, we first randomly generate the two-moon dataset to demonstrate the superiority of the graph learning strategy of MFSGL. In this dataset, two clusters of data samples are scattered in the two-moon space and there are 100 data points in each cluster. As shown in Fig. 1, the two clusters are labeled in red and blue respectively, and they are separated into two independent views. In addition, a noise view is also generated. An ideal multi-view similarity graph learning method should integrate the different views, and distinguish the two clusters exactly. View 1 and View 2 make up the pure two-moon dataset. View 1, View 2 and Noise View make up the noisy two-moon dataset. The experiments are conducted in the two datasets respectively. The parameter p of MFSGL is set as 1.

Firstly, on the pure two-moon dataset, we fix the neighbor number k as 5, 10 and 15, and show the learned similarity graphs of MFSGL and ASVW in Fig. 2. When k = 5, 10, some pairs of data samples contained in the same cluster are not linked in the learned graph of ASVW. That is to say, ASVW fails to separate data samples into two clusters. Whereas, in the learned graph of MFSGL, the data samples are divided into two clusters successfully. When k = 15, although ASVW gives the correct result, MFSGL shows more strength. Both ASVW and MFSGL learn a common similarity graph of all views during feature selection. In MFSGL, a rank constraint is added to the similarity matrix, which brings that the number of connected components in the learned graph equals to the number of clusters, and each connected component corresponds to one cluster. Therefore, MFSGL is able to capture the relationship between data samples accurately from multiple views.

Secondly, to further verify the efficiency of proposed multi-view similarity graph learning strategy, another experiment is conducted on the noisy two-moon dataset. The neighbor number k of all methods are set as 10. Affinity Aggregation Spectral Clustering (AASC) [38] and Auto-weighted Multiple Graph Learning (AMGL) [39] both pre-define the similarity graph of each view, and connect them by the learned view weights. The result in Fig. 3 shows that AASC, AMGL and ASVW all fails to divide the data samples into correct clusters. Obviously, MFSGL can learn the data structural information from different views precisely, and is robust to noise as well.

Moreover, the iterative curve of the view weights learned by MFSGL are demonstrated in Fig. 4. The presented results have been normalized. On the pure two-moon dataset, the two views contain complementary information, and the learned view weights are approximately equal to 0.5. On the noisy two-moon dataset, the weights of first two views increase while the third view weight decreases, which complies the assumption that noisy view should be with small weight. Therefore, MFSGL is able to assign an appropriate weight for each view according to their significance.



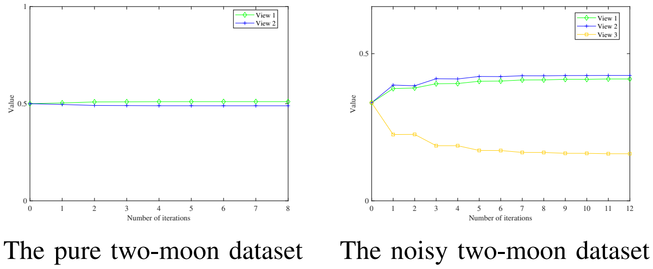
MFSGL

ASVW

AMGL

AASC

Fig. 3. The graphs learned on noisy two-moon dataset.



The pure two-moon dataset

The noisy two-moon dataset

Fig. 4. The iterative curve of the view weights on two datasets.

B. Experiments on the Real-world Datasets

In this part, MFSGL is compared with several state-of-the-art unsupervised feature selection approaches on four real-world datasets to demonstrate its effectiveness.

1) Real-world Datasets Description: In our experiments, four benchmark multi-view datasets are used, including Outdoor-Scene [40], Caltech101-7 [41], NUS-WIDE-OBJ [42] and Handwritten numerals [43]. In each dataset, heterogeneous features are extracted as different views. The details of them are summarized in Table II.

- *Outdoor-Scene*. This dataset includes 2688 images which belong to the following categories: mountain, open country, coast, forest, inside city, highways, street and tall buildings. For each image, we extract four kinds of visual features as different data views.

- *Caltech101-7*. This dataset is commonly used for object recognition. Caltech101 includes 8677 images from 101 classes. Following [27], we choose 1474 pictures totally of the widely used 7 categories, i.e., Faces, Motorbikes, Dolla-Bill, Garfield, Snoopy, Stop-Sign and Windsor-Chair. Different visual features are also extracted as different data views.
- *NUS-WIDE-OBJ*. This dataset consists of 30,000 images of 31 categories totally. In our experiment, we select 25 categories with a total of 3,000 images and extract five kinds of features to represent each image.
- *Handwritten numerals*. This dataset contains 2000 digital images from 0 to 9, and each category contains 200 images. Six different features are extracted in this dataset.

2) Comparison Scheme: Several representative single-view feature selection methods including LabScor, MCFS , RSFS and SOGFS are employed to show the effectiveness of MFSGL, and they take the samples represented by the connected features of all views as input. ASVW, RMFS and DEKM [22] are the multi-view feature selection methods. In addition, the connected features of all views is used to perform K-means as the baseline. For the selected feature subset of different size, we perform K-means from the same starting points to make the experiments fair enough. The parameter k in MFSGL is searched from 5 to 15 with the step size 5. The parameter γ in MFSGL is searched in logarithm form, i.e., $log_{10}\gamma$ is searched from -2 to 4 with the step size 1. Fixing parameter $p = 1$, we select the best result of MFSGL and show it in the following part. For compared methods, the value range of parameters are set as they reported and only the best result is given.

3) Evaluation Metrics: For each feature selection method, the samples represent by the selected features are fed into K-means, and the clustering result is used to measure the performance of feature selection. The clustering accuracy(ACC) and the normalized mutual information (NMI) are adopted to evaluate clustering result in this paper [44].

ACC reveals the matching degree of clustering result and the ground-truth by discovering the one-to-one correspondence. The definition of ACC is shown as follows:

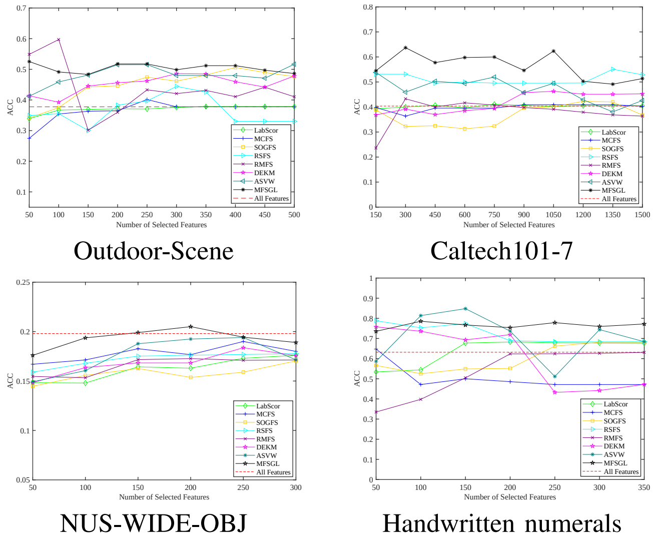
$$ACC(h, l) = \frac{\sum_{i=1}^{n} \delta(h_i, l_i)}{n}, \tag{36}$$

where h_i and l_i are the ground-truth label and clustering result label after best mapping of i-th sample respectively. If $h_i = l_i$, $\delta(h_i, l_i)$ equals 1. Otherwise, it equals 0.

Apart from ACC, NMI is another evaluation metric:

$$NMI = \frac{\sum_{i=1}^{c} \sum_{j=1}^{c} n_{ij} log \frac{n_{ij}}{n_i * \hat{n}_j}}{\sqrt{\sum_{i=1}^{c} n_i log \frac{n_i}{n} * \sum_{j=1}^{c} \hat{n}_j log \frac{\hat{n}_j}{n}}}. \tag{37}$$

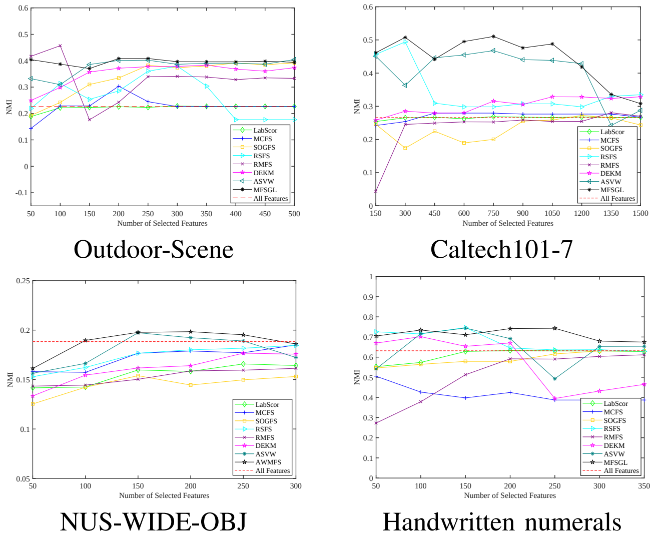
Here, n is the total number of samples, and c denotes the number of classes. n_i is the number of samples belonging to the i-th cluster based on the experimental result, and $\hat{n}_j$ denotes the real number of samples belonging to the j-th class. n_{ij} denotes the number of samples which are exist in i-th cluster and j-th class simultaneously.



Outdoor-Scene Caltech101-7

NUS-WIDE-OBJ Handwritten numerals

Fig. 5. ACC of different methods on four datasets.



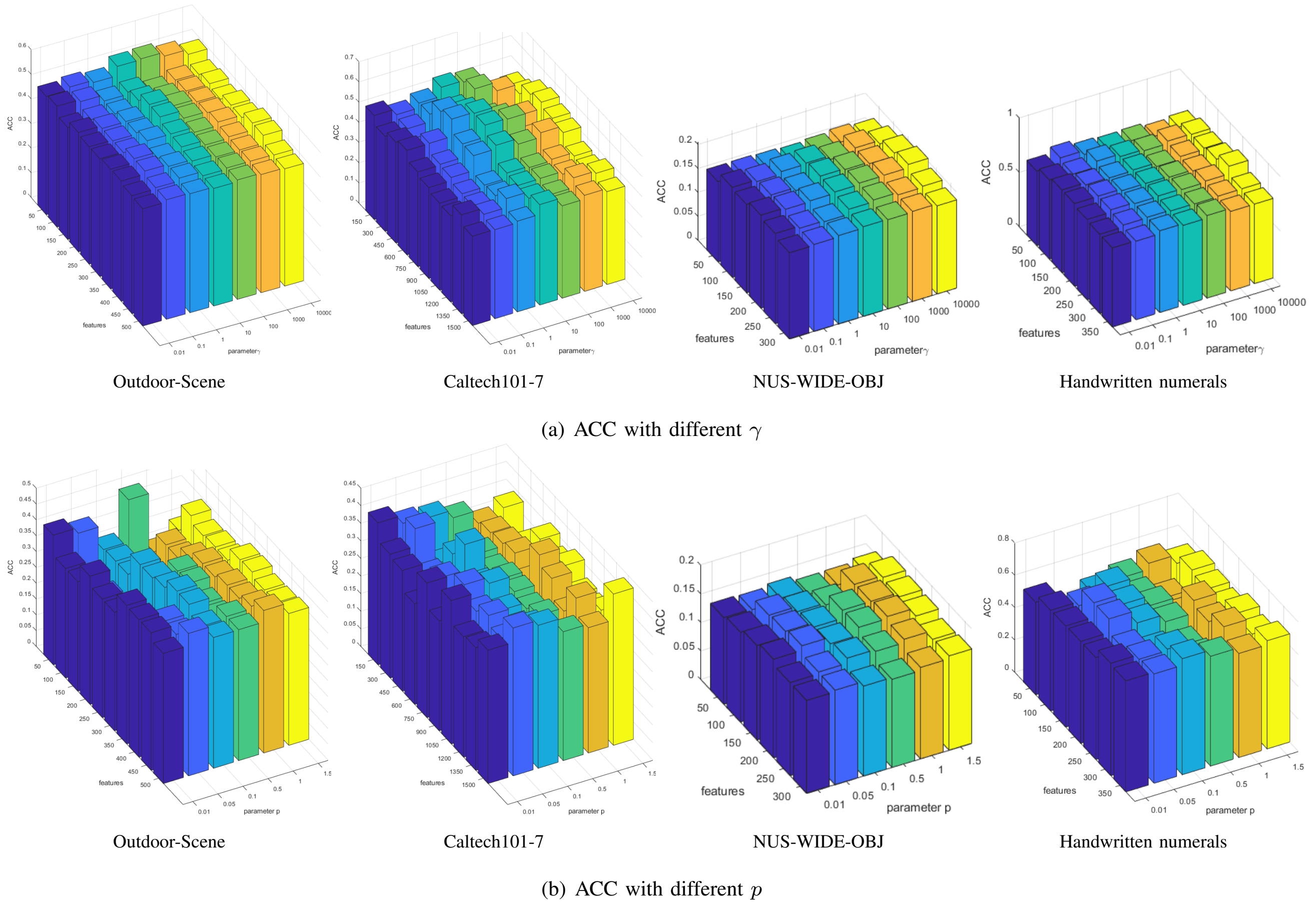
Outdoor-Scene Caltech101-7

NUS-WIDE-OBJ Handwritten numerals

Fig. 6. NMI of different methods on four datasets.

4) Clustering Results with Selected Features: Fig. 5 and Fig. 6 show the ACC performance and NMI performance of each approach respectively. The detailed analyses of experimental results are conducted and some valuable points can be obtained.

- As is seen from the experimental results, on *Outdoor-Scene*, *Handwritten numerals* and *Caltech101-7* datasets, the result of MFSGL is superior to other methods in most cases. On *NUS-WIDE-OBJ* dataset, the clustering results of all compared methods are worse than the baseline most of the time. We infer that this is because the feature number of this dataset is too small, and there is not as much noise as others. When the selected subset is too small, it can not contain all the valuable features, and perform worse than the baseline. As the feature number increases, the performance firstly becomes better for the newly added valuable features, and then the performance



(a) ACC with different γ

(b) ACC with different p

Fig. 7. Parameter sensitivity study.

becomes worse because the feature subset contains too much noise.

- In most cases, the performance of multi-view feature selection methods, i.e., MFSGL, ASVW, DEKM and RMFS show more strength than the other single-view feature selection methods. The fact indicates that considering the complementary information across different views makes a big significance in multi-view feature selection. Moreover, we believe that it is the same in other multi-view learning tasks.
- Obviously, MFSGL and ASVW also outperform the other two multi-view feature selection methods, i.e., RMFS and DEKM. These two methods focus on the global structure. The graph-based methods MFSGL and ASVW try to preserve the underlying local manifold structure shared by all views. It reveals that the local structure information is superior to the global structure information once again, which has been widely recognized by researchers.
- Furthermore, the proposed MFSGL also outperforms ASVW. Both MFSGL and ASVW try to learn the common similarity matrix of all views. Different from ASVW, the similarity matrix learned by MFSGL has an ideal neighbor assignment owing to the added rank constraint, which contributes a lot to the superior performance.

5) Parameter Sensitivity Study: The parameter m_v gives slight influence on the performance, and in our experiments we set it around $\frac{d_v}{3}$ to $\frac{2d_v}{3}$. This is because if m_v is set too small, some valuable features may be lost, and if m_v is set too large, the learned feature subset may be not compact and discriminative enough. On account of the parameter γ controls the row sparsity of W_v, we concentrate on the performance

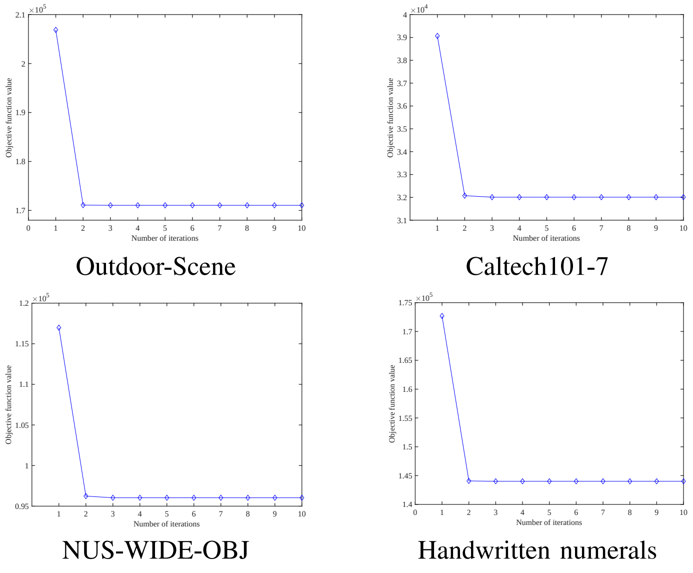


Fig. 8. Convergence curves of Algorithm 1 on four datasets.

of MFSGL with various γ. With fixed $p = 1$, the best ACC results under various γ on four real-world datasets are shown in Fig. 7a. It is easy to know that the proposed MFSGL is robust to γ to a certain extent. In spite of this, it is suggested to search the optimal γ for the best performance in practical application.

Moreover, the influence of different exponential functions should also be taken into account. Therefore, we change the value of parameter p and fix other parameters, and the variance on performance of MFSGL is demonstrated in Fig. 7b. It is clear that there is no significant difference with different p, and we can easily conclude that the performances of MFSGL with different exponential functions are all at a high level.

6) Convergence Study: The Algorithm 1 is proposed to solve problem (16) and its convergence has been proved in Section III-B4. Here, the speed of its convergence is further studied by experiment. For brevity, we only show the result of one view on each dataset. In Fig. 8, it is obvious that Algorithm 1 converges within about 5 iterations. And it also keeps the same convergence speed in other views. The fast convergence saves much computational time of the proposed feature selection framework.

V. Conclusion

In this paper, we present a novel method called MFSGL for unsupervised multi-view feature selection. MFSGL learns an optimal similarity graph across different views by adding a reasonable constraint, and an efficient view weight assignment strategy is adopted to balance the contribution of each view. An algorithm is also given to optimize this problem. The experiments on the synthetic datasets and four benchmark real-world datasets validate the superiority of MFSGL. In future, we will be committed to find some new techniques to further improve the quality of similarity graph.

References

[1] Q. Wang, M. Chen, F. Nie, and X. Li, “Detecting coherent groups in crowd scenes by multiview clustering,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 1, pp. 46–58, 2020.

[2] X. Li, M. Chen, F. Nie, and Q. Wang, “A multiview-based parameter free framework for group detection,” in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 2017, pp. 4147–4153.

[3] Z. Yang, Q. Li, W. Liu, and J. Lv, “Shared multi-view data representation for multi-domain event detection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 5, pp. 1243–1256, 2020.

[4] Y. Li, X. Shi, C. Du, Y. Liu, and Y. Wen, “Manifold regularized multi-view feature selection for social image annotation,” *Neurocomputing*, vol. 204, pp. 135–141, 2016.

[5] W. Liu and D. Tao, “Multiview hessian regularization for image annotation,” *IEEE Transcations on Image Processing*, vol. 22, no. 7, pp. 2676–2687, 2013.

[6] Y. Han, Y. Yang, Y. Yan, Z. Ma, and X. Zhou, “Semisupervised feature selection via spline regression for video semantic recognition,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 2, pp. 252–264, 2015.

[7] Z. Wang, Y. Feng, T. Qi, Y. X, and J. Zhang, “Adaptive multi-view feature selection for human motion retrieval,” *Signal Processing*, vol. 120, pp. 691–701, 2016.

[8] Y. Feng, J. Xiao, and Y. Z. andX. Liu, “Adaptive unsupervised multi-view feature selection for visual concept recognition,” in *Asian Conference on Computer Vision*, 2012.

[9] J. Y. X. C. C. Hong, J. Yu and D. Tao, “Multi-view ensemble manifold regularization for 3d object recognition,” vol. 320, 2015, p. 395405.

[10] F. Nie, G. Cai, J. Li, and X. Li, “Auto-weighted multi-view learning for image clustering and semi-supervised classification,” *IEEE Transactions on Image Processing*, vol. 27, no. 3, pp. 1501–1511, 2018.

[11] H. Tao, C. Hou, D. Yi, and J. Zhu, “Multiview classification with cohesion and diversity,” *IEEE Transactions on Cybernetics*, vol. 50, no. 5, pp. 2124–2137, 2020.

[12] X. Zhu, X. Li, and S. Zhang, “Block-row sparse multiview multilabel learning for image classification,” *IEEE Transactions on Cybernetics*, vol. 46, no. 2, pp. 450–461, 2016.

[13] J. Yu, Y. Rui, Y. Y. Tang, and D. Tao, “High-order distance-based multiview stochastic learning in image classification,” *IEEE Transactions on Cybernetics*, vol. 44, no. 12, pp. 2431–2442, 2014.

[14] X. Wang, T. Zhang, and X. Gao, “Multiview clustering based on non-negative matrix factorization and pairwise measurements,” *IEEE Transactions on Cybernetics*, vol. 49, no. 9, pp. 3333–3346, 2019.

[15] I. Guyon, *An introduction to variable and feature selection*, 2003.

[16] F. Nie, W. Zhu, and X. Li, “Unsupervised feature selection with structured graph optimization,” in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, 2016, pp. 1302–1308.

[17] C. Hou, F. Nie, X. Li, D. Yi, and Y. Wu, “Joint embedding learning and sparse regression: A framework for unsupervised feature selection,” *IEEE Transactions on Cybernetics*, vol. 44, no. 6, pp. 793–804, 2014.

[18] L. Shi, L. Du, and Y. Shen, “Robust spectral learning for unsupervised feature selection,” in *IEEE International Conference on Data Mining*, 2014, pp. 977–982.

[19] C. Deng, C. Zhang, and 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] X. He, C. Deng, and P. Niyogi, “Laplacian score for feature selection,” in *Advances in Neural Information Processing Systems*, 2005, pp. 507–514.

[21] H. Liu, H. Mao, and Y. Fu, “Robust multi-view feature selection,” in *IEEE 16th International Conference on Data Mining*, 2016, pp. 281–290.

[22] J. Xu, J. Han, and F. Nie, “Discriminatively embedded k-means for multi-view clustering,” in *IEEE Conference on Computer Vision and Pattern Recognition,*, 2016, pp. 5356–5364.

[23] W. Liu, X. Ma, Y. Zhou, D. Tao, and J. Cheng, “p-laplacian regularization for scene recognition,” *IEEE Transactions on Cybernetics*, vol. 49, no. 8, pp. 2927–2940, 2019.

[24] H. Shi, Y. Li, Y. Han, and Q. Hu, “Cluster structure preserving unsupervised feature selection for multi-view tasks,” *Neurocomputing*, vol. 175, pp. 686–697, 2016.

[25] Y. Pang, B. Zhou, and F. Nie, “Simultaneously learning neighborship and projection matrix for supervised dimensionality reduction,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 9, pp. 2779–2793, 2019.

[26] J. Tang, X. Hu, H. Gao, and H. Liu, “Unsupervised feature selection for multi-view data in social media,” in *Proceedings of the 13th International Conference on Data Mining*, 2013, pp. 270–278.

[27] C. Hou, F. Nie, H. Tao, and D. Yi, “Multi-view unsupervised feature selection with adaptive similarity and view weight,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 9, pp. 1998–2011, 2017.

[28] Y. Yang, H. T. Shen, Z. Ma, Z. Huang, and X. Zhou, “$l_{2,1}$-norm regularized discriminative feature selection for unsupervised learning,” in *Proceedings of the 22nd International Joint Conference on Artificial Intelligence*, 2011, pp. 1589–1594.

[29] F. Nie, H. Huang, X. Cai, and C. H. Q. Ding, “Efficient and robust feature selection via joint 2, 1-norms minimization,” in *24th Annual Conference on Neural Information Processing Systems*, 2010, pp. 1813–1821.

[30] D. Wang, F. Nie, and H. Huang, “Unsupervised feature selection via unified trace ratio formulation and k-means clustering (TRACK),” in *Machine Learning and Knowledge Discovery in Databases - European Conference,*, vol. 8726, 2014, pp. 306–321.

[31] F. Nie, X. Wang, and H. Huang, “Clustering and projected clustering with adaptive neighbors,” in *The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD*, 2014, pp. 977–986.

[32] B. Mohar, Y. Alavi, G. Chartrand, O. R. Oellermann, and A. J. Schwenk, “The laplacian spectrum of graphs,” *Graph Theory Combinations and Applications*, vol. 18, no. 7, pp. 871–898, 1991.

[33] F. Nie, X. Wang, M. Jordan, and H. Huang, "The constrained laplacian rank algorithm for graph-based clustering," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, 2016, pp. 1969–1976.

[34] K. Fan, "On a theorem of weyl concerning eigenvalues of linear transformations: Ii," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 36, no. 1, pp. 31–35, 1950.

[35] Q. Wang, Z. Qin, F. Nie, and X. Li, "Spectral embedded adaptive neighbors clustering," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 4, pp. 1265–1271, 2019.

[36] U. Luxburg, "A tutorial on spectral clustering," *Statistics and Computing*, vol. 17, no. 4, pp. 395–416, 2007.

[37] F. Nie, H. Huang, C. Xiao, and C. H. Q. Ding, "Efficient and robust feature selection via joint l2,1-norms minimization," in *International Conference on Neural Information Processing Systems*, 2010, pp. 1813–1821),.

[38] H. Huang, Y. Chuang, and C. Chen, "Affinity aggregation for spectral clustering," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 2012, pp. 773–780.

[39] F. Nie, J. Li, and X. Li, "Parameter-free auto-weighted multiple graph learning: A framework for multiview clustering and semi-supervised classification," in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, 2016, pp. 1881–1887.

[40] A. Monadjemi, B. Thomas, and M. Mirmehdi, "Experiments on high resolution images towards outdoor scene classification," *Technical Report, University of Bristol, Department of Computer Science*, 2002.

[41] F. Li, R. Fergus, and P. Perona, "Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories," in *Conference on Computer Vision and Pattern Recognition Workshop*, 2004, p. 1787.

[42] T. Chua, J. Tang, R. Hong, H. Li, Z. Luo, and Y. Zheng, "Nus-wide: a real-world web image database from national university of singapore." in *CIVR*, 2009.

[43] A. Asuncion and D. Newman., "Uci machine learning repository," 2007.

[44] J. Huang, F. Nie, H. Huang, and C. H. Q. Ding, "Robust manifold nonnegative matrix factorization," *ACM Transactions on Knowledge Discovery from Data*, vol. 8, no. 3, pp. 11:1–11:21, 2013.

Qi Wang (M'15-SM'15) received the B.E. degree in automation and the Ph.D. degree in pattern recognition and intelligent systems from the University of Science and Technology of China, Hefei, China, in 2005 and 2010, respectively. He is currently a Professor with the School of Artificial Intelligence, Optics and Electronics (iOPEN), Northwestern Polytechnical University, Xi'an, China. His research interests include computer vision, pattern recognition and remote sensing.

Xu Jiang received the B.E. degree in electronics and information engineering from Northwestern Polytechnical University, Xi'an, China, in 2019. He is currently working toward the M.S. degree in computer science in the School of Artificial Intelligence, Optics and Electronics (iOPEN), Northwestern Polytechnical University, Xi'an, China. His research interests include machine learning and data mining.

Mulin Chen received the B.E. degree in software engineering and the Ph.D. degree in computer application technology from Northwestern Polytechnical University, Xian, China, in 2014 and 2019 respectively. He is currently a researcher with the School of Artificial Intelligence, Optics and Electronics (iOPEN), Northwestern Polytechnical University, Xi'an, China. His current research interests include computer vision and machine learning.

Xuelong Li (M'02-SM'07-F'12) is a full professor with the School of Artificial Intelligence, Optics and Electronics (iOPEN), Northwestern Polytechnical University, Xi'an, China.