Knowledge-Based Systems 194 (2020) 105482

Contents lists available at ScienceDirect

Knowledge-Based Systems

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

Graph-regularized least squares regression for multi-view subspace clustering☆

Yongyong Chen [a], Shuqin Wang [b,*], Fangying Zheng [c], Yigang Cen [b]

[a] Department of Computer and Information Science, University of Macau, Macau 999078, China
[b] Institute of Information Science, Beijing Jiaotong University, Beijing 100044, China
[c] Department of Mathematical Sciences, Zhejiang Sci-Tech University, Hangzhou 310018, China

ARTICLE INFO

Article history:
Received 3 October 2019
Received in revised form 30 November 2019
Accepted 6 January 2020
Available online 13 January 2020

Keywords:
Multi-view clustering
Subspace clustering
Least squares regression
Column-sparsity norm
Manifold constraint

ABSTRACT

Many works have proven that the consistency and differences in multi-view subspace clustering make the clustering results better than the single-view clustering. Therefore, this paper studies the multi-view clustering problem, which aims to divide data points into several groups using multiple features. However, existing multi-view clustering methods fail to capturing the grouping effect and local geometrical structure of the multiple features. In order to solve these problems, this paper proposes a novel multi-view subspace clustering model called graph-regularized least squares regression (GLSR), which uses not only the least squares regression instead of the nuclear norm to generate grouping effect, but also the manifold constraint to preserve the local geometrical structure of multiple features. Specifically, the proposed GLSR method adopts the least squares regression to learn the globally consensus information shared by multiple views and the column-sparsity norm to measure the residual information. Under the alternating direction method of multipliers framework, an effective method is developed by iteratively update all variables. Numerical studies on eight real databases demonstrate the effectiveness and superior performance of the proposed GLSR over eleven state-of-the-art methods.

1. Introduction

Clustering refers to the process of dividing a collection of unlabeled objects into multiple categories, which has received wide attention in image and motion segmentation [1,2], face clustering [3–5], image representation [6,7] and other fields. Considerable efforts have been expended to develop efficient and effective methods for clustering, such as K-means-based methods [8], collaborative methods [9], graph-based methods [10–14], and spectral-based methods [1,15–18].

Due to the complexity and diversity of data in practical applications such as trade transaction data, web documents, gene expression data, existing algorithms may yield unsatisfactory clustering results, especially for high-dimensional and large data. To solve this problem, the concept of subspace clustering [19] was proposed, which has become the most popular method for clustering. The whole process of subspace clustering methods include two steps: (1) learn the similarity matrix using different regularizers, such as sparsity [1], low-rankness [15], and their combinations [20]; (2) perform spectral clustering [17] on the learned similarity matrix (also called affinity matrix) to obtain the final clustering results. Generally speaking, the quality of the learned similarity matrix directly determines the clustering performance. For example, sparse subspace clustering (SSC) [1] and low-rank representation (LRR) [15] used the l_1-norm and nuclear norm to learn the affinity matrix, respectively. The work in [20] proposed to exploit both the low-rankness and sparsity to construct the similarity matrix. The works in [21,22] incorporated the Laplacian regularizer into the sparse coding and LRR model to develop the Laplacian sparse coding framework and Laplacian regularized LRR, respectively. The studies in [10,23] proposed to combine the ideas of the graph theory and subspace clustering into one unified model to discover the global mixture of subspace structure within data. The main differences among them are that the work in [10] used the non-negative matrix factorization to find a low-rank approximation while the study in [23] still used the nuclear norm to measure the low-rank property. Instead of the convex approximation, several non-convex low-rank regularizers, such as Logdet rank [24], gamma-norm [25], and weighted nuclear norm [26] have been developed. The affinity matrix obtained by LRR may not be a diagonal matrix, so it cannot guarantee accurate clustering results [16]. To address this limitation, the least squares regression (LSR) [16] method was

☆ No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.knosys.2020.105482.

* Corresponding author.
E-mail address: ShuqinWang.cn@hotmail.com (S. Wang).

https://doi.org/10.1016/j.knosys.2020.105482
0950-7051/

proposed to effectively preserve important structural information underlying the data and generate grouping effect to yield promising clustering results.

The above methods belong to single-view clustering, however, in practice, the same data object can be described from different perspectives leading to multi-view data (also called multi-view features) [12,27,28]. For example, in face recognition [27], each face image can be represented by different types of features, such as color, texture and edge. In outlier detection [29], multi-view data are selected to identify different types of data outliers. In [30], miRNA-disease associations, disease semantic information, experimentally verified miRNA–target gene interactions, and gene–gene interaction network are to estimate the miRNA similarity. Inspired by the huge success of the sparsity and low-rankness in single-view clustering setting, many works have extended single-view clustering methods into multi-view settings. For example, Brbić et al. [31] proposed the multi-view low-rank sparse subspace clustering by imposing both low-rank and sparsity constraints on the construction of the affinity matrix. The studies in [29,32] both imposed the low-rank representation to identify outliers from the multi-view perspective. Based on the connection between spectral clustering and Markov chains, Xia et al. [33] developed a novel multi-view subspace clustering method by the low-rank and sparse matrix decomposition. By integrating all representation matrices as a third-order tensor, Zhang et al. [27] extended the low-rank matrix representation into the tensor setting, in which a low-rank tensor constraint was introduced to explore the complementary information among multiple views. More discussions about related works are reported in the next section.

Unfortunately, there are several drawbacks in existing multi-view clustering methods: (1) most of multi-view clustering methods such as [27,31,33,34] used the nuclear norm or l_1-norm to construct the affinity matrix which may not be diagonal [16]. (2) they do not take into account the inherently local geometrical structure of multiple features. Therefore, it still has room for further boosting performance of multi-view clustering. To address these limitations, in this paper, we proposed the Graph-regularized Least Squares Regression (GLSR) method for multi-view subspace clustering. Fig. 1 illustrates the flowchart of the proposed GLSR model. Considering that the LSR method [16] tends to reduce the coefficients of the relevant data and shows the grouping effect, the proposed GLSR method utilizes the Frobenius norm instead of the nuclear norm to learn the consensus information shared among all views and the column-sparsity norm to measure the residual information of each view. The main contributions of this paper are as follows:

- The proposed GLSR method effectively utilizes the consensus and residual information between different views for multi-view subspace clustering. This is achieved by using the least squares regression instead of the nuclear norm to generate grouping effect and the column-sparsity norm to measure the residual information. At the same time, the manifold regularization is introduced to maintain the local geometrical structure.
- We adopt the augmented Lagrangian multiplier (ALM) method to solve the proposed GLSR method, where the closed-form solution of each subproblem is derived.
- The superiority of the proposed GLSR method is sufficiently validated by conducting extensive experiments on eight real databases over eleven state-of-the-art methods.

The remainder of this paper is organized as follows. Section 2 provides a brief review of multi-view clustering methods. Section 3 presents our GLSR model and designs an effective algorithm to solve the GLSR model. We evaluate the performance of our proposed method in Section 4 and summarize this paper in Section 5.

2. Related work

Before introducing the proposed GLSR model, we briefly review the multi-view subspace clustering methods proposed in recent years, which can be roughly divided into the following categories: matrix-based methods, graph-based methods and tensor-based methods.

Matrix-based methods: The key to matrix-based methods is to obtain a low-rank coefficient matrix. There are two common schemes to learn a low-rank coefficient matrix. The first scheme is based on the nuclear norm, while the last one is to use the non-negative matrix factorization (NMF). For the first scheme, the nuclear norm is adopted as the convex approximation of the non-convex rank function to depict the low-rank property of the coefficient matrix. For instance, the work in [33] exploited the nuclear norm to recover a shared low-rank transition probability matrix as a crucial input to the standard Markov chain method for multi-view clustering. Brbić et al. [31] performed sparse and low-rank subspace clustering on each view and then combined the correlation matrices to obtain a common clustering. For the second scheme, several NMF-based multi-view clustering algorithms have been proposed to effectively learn the basic clustering structure embedded in multiple views and make the different coefficient matrices comparable and meaningful. For example, Liu et al. [35] decomposed the original matrix into discriminative basis matrices and coefficient matrices and then found a consensus coefficient matrix to balance all coefficient matrices. However, the above methods do not maintain the local geometrical structure of multiple features.

Graph-based methods: For the graph-based methods, the key point is to construct data graph matrices effectively. In order to connect the features of two views, the work in [36] constructed a bipartite graph and then used the standard spectral clustering to obtain the clustering results of two views. However, it cannot handle more than two views. Nie et al. [37] proposed a novel multi-view clustering model, which performed both clustering and local structure learning. In addition, this method directly divides the obtained optimal graph into specific clusters and automatically assigns the ideal weight to each view without explicit weight definition and penalty parameters. In order to evaluate the impact of different graph matrices on multi-view clustering performance, Wang et al. [12] proposed a general graph-based system (GBS-KO) which can effectively construct a data graph matrix, automatically assign weight to each one and directly generate clustering results. Similar to [12], Wang et al. [38] proposed a graph-based multi-view clustering method (GMC). Compared with GBS-KO, GMC can help to learn each view graph matrix and unified matrix in a mutual reinforcement manner. In order to better capture the relationship between multiple undirected graphs, the method in [39] combined information from multiple graphs and link matrix factorization, where matrix factorization is performed for each graph. When handling large-scale data, most of the existing graph methods may cause the information loss. To address this problem, Luo et al. [40] developed a general framework for multi-view discrete graph clustering by directly learning a consensus partition across multiple views. The work in [41] proposed a novel large-scale multi-view spectral clustering approach to improve computational efficiency.

Tensor-based methods: Although the above methods is effective, the advantages of multi-view features may not be fully explored. The reason is that most previous methods capture only the pairwise correlations between different views, while ignoring high-order correlations between multiple features. To overcome this limitation, a variety of tensor-based methods have been proposed [27,42–45]. For example, Zhang et al. [27] proposed

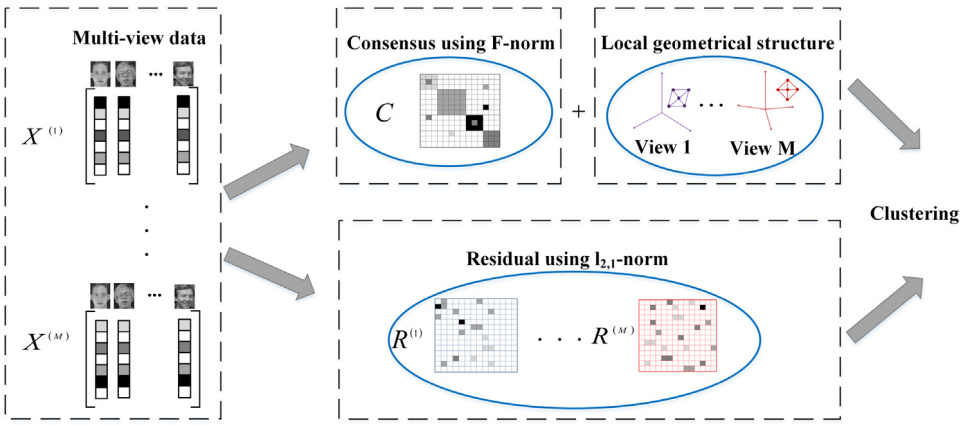

Fig. 1. The flowchart of the proposed GLSR model. Given a collection of data points with multi-view features $\{X^{(1)}, \ldots, X^{(M)}\}$, we use the proposed GLSR model to derive the consensus matrix C to learn the global consensus information shared by all views and the residual matrix $R^{(V)}$ to measure the residual information of the vth view. In addition, GLSR exploits the manifold constraint to describe the local geometrical structure of multiple features.

a multi-view subspace clustering method with low-rank tensor constraint, in which the subspace representation matrices of different views are considered to capture high-order correlation under multi-view data. In [43], Xie et al. applied the low-rank tensor constraint on the rotated subspace coefficient tensor by introducing a new tensor decomposition scheme to ensure consistency between multiple views. However, the tensor-based approaches are used to capture the global structure of all views and explore the correlation between them, while ignoring the local spatial geometrical structure of all view features. A detailed motivation of our paper can be found in the next section.

3. The proposed GLSR

In this section, we first present the proposed GLSR model for multi-view subspace clustering. Then, we use the augmented Lagrangian multiplier method to solve the GLSR model and derive the closed-form solution of each subproblem. The key to the proposed GLSR is to use the least squares regression instead of the nuclear norm to generate the grouping effect, which can achieve accurate clustering. The column-sparsity norm and manifold constraint are simultaneously taken into consideration to measure the residual information and preserve the local geometrical structure of multiple features, respectively.

3.1. Model formulation

Suppose $X = [X_1, X_2, \ldots, X_n] \in R^{d\times n}$ is the data matrix, where each column is a d dimensional sample vector and n is the total number of data points. Subspace clustering aims to obtain a self-representation matrix and then implement the spectral clustering on it. One common assumption is that each data point in a union of subspaces can be represented as a linear combination of the other data points, *i.e.*, $X = XZ + E$, where $Z \in R^{n\times n}$ and $E \in R^{d\times n}$ are the self-representation and error matrices, respectively. The most representative subspace clustering method LRR finds the low rank representation by solving the following optimization model:

$$\begin{aligned} &\min_{Z,E} \|Z\|_* + \beta\|E\|_{2,1} \\ &s.t.\ \ X = XZ + E, \end{aligned} \tag{1}$$

where $\|\cdot\|_*$ is the nuclear norm, *i.e.*, the sum of all the singular values of the matrix. $\|\cdot\|_{2,1}$ is the $l_{2,1}$-norm, which encourages the columns of E to be sparse.

Although LRR has achieved promising performance, it only focuses on single-view clustering task without exploring multi-view features. At the same time, the diagonality of the affinity matrix obtained by LRR cannot be guaranteed, so it may not obtain effective clustering results. To represent the data comprehensively and deeply, this paper proposes a GLSR method for multi-view subspace clustering by using the superior power of the LSR method. Let $X^{(v)} \in R^{d_{(v)}\times n}$ be the vth feature matrix. $d_{(v)}$ is the feature dimension of a sample vector in the vth view. The proposed GLSR method can be expressed as the following model:

$$\begin{aligned} &\min_{C,R^{(v)},E^{(v)}} \|C\|_F^2 + \sum_{v=1}^{M}\left(\alpha\|R^{(v)}\|_{2,1} + \beta\|E^{(v)}\|_{2,1} + \lambda tr(CL^{(v)}C^T)\right) \\ &s.t.\ \ X^{(v)} = X^{(v)}C + X^{(v)}R^{(v)} + E^{(v)}, \quad v = 1, \ldots, M, \end{aligned} \tag{2}$$

where $L^{(v)}$ is the normalized graph Laplacian matrix of the vth view and M is the number of views. In the proposed GLSR model, we divide the multi-view data matrix into three parts: C, $R^{(v)}$, $E^{(v)}$ and use the Frobenius norm and $l_{2,1}$-norm to draw the corresponding term. Matrix $C \in R^{n\times n}$ is the consensus matrix shared by all views. Matrix $R^{(v)} \in R^{n\times n}$ is the residual matrix to the vth view. The size of the error matrix $E^{(v)}$ is $d_{(v)} \times n$.

The proposed GLSR in Eq. (2) may offer several favorable merits:

- Two important characteristics in multi-view subspace clustering, *i.e.*, consensus shared among all views and residuals of different views are simultaneously exploited in the proposed GLSR.
- For the consensus matrix C, we use the least squares regression to generate the grouping effect. For the residual matrix $R^{(v)}$, it can be viewed as the error component for the consensus matrix of the vth view. Thus, we constrain $R^{(v)}$ with the $l_{2,1}$-norm to make its columns sparse. For the error matrix $E^{(v)}$, due to the outliers and sample-specific corruptions, the $l_{2,1}$-norm is also adopted to characterize the error term $E^{(v)}$ in our objective function.
- In order to maintain the local geometrical structure of multiple features, the manifold regularization is introduced in the GLSR model.

3.2. Optimization of GLSR

Borrowing the idea of ALM method [46], the constraint optimization model (2) is solved by minimizing the following unconstrained augmented Lagrangian function

$$\mathcal{L}_\rho(C, R^{(v)}, E^{(v)}; \Theta^{(v)}) = \|C\|_F^2 + \sum_{v=1}^{M}\Big(\alpha\|R^{(v)}\|_{2,1} + \beta\|E^{(v)}\|_{2,1} + \lambda tr(CL^{(v)}C^T)\Big) + \sum_{v=1}^{M}\Big(\langle\Theta^{(v)}, X^{(v)} - X^{(v)}C - X^{(v)}R^{(v)} - E^{(v)}\rangle + \frac{\rho}{2}\|X^{(v)} - X^{(v)}C - X^{(v)}R^{(v)} - E^{(v)}\|_F^2\Big), \tag{3}$$

where $\Theta^{(v)}$ is the Lagrange multiplier; $\rho > 0$ is the penalty parameter; $\langle\cdot,\cdot\rangle$ denotes the inner product. Then, each variable is updated iteratively by fixing the other variables as follows:

(1) Update consensus C: When other variables are fixed, C_{k+1} can be updated by

$$\min_C \|C\|_F^2 + \sum_{v=1}^{M}\Big(\lambda tr(CL^{(v)}C^T) + \frac{\rho_k}{2}\|X^{(v)}C - T_k^{(v)}\|_F^2\Big) \tag{4}$$

where $T_k^{(v)} = X^{(v)} - X^{(v)}R_k^{(v)} - E_k^{(v)} + \frac{\Theta_k^{(v)}}{\rho_k}$. By setting the derivative of Eq. (4) with respect to C to zero, we can obtain a Sylvester equation $A*C + C*B = Y$, in which $A = 2*I + \sum_{v=1}^{M}\rho_k X^{(v)^T}X^{(v)}$, $B = 2\lambda\sum_{v=1}^{M}L^{(v)}$, and $Y = \sum_{v=1}^{M}\rho_k X^{(v)^T}(T_k^{(v)} - X^{(v)}R_k^{(v)})$.

(2) Update residual $R^{(v)}$: With other variables fixed, updating $R_{k+1}^{(v)}$ is equal to solving the following problem

$$\min_{R^{(v)}} \alpha\|R^{(v)}\|_{2,1} + \frac{\rho_k}{2}\|X^{(v)}R^{(v)} - N_k^{(v)}\|_F^2, \tag{5}$$

where $N_k^{(v)} = X^{(v)} - X^{(v)}C_{k+1} - E_k^{(v)} + \frac{\Theta_k^{(v)}}{\rho_k}$. The closed-form solution of Eq. (5) is

$$R_{k+1}^{(v)} = \big(2\alpha\phi + \rho_k X^{(v)^T}X^{(V)}\big)^{-1}\rho_k X^{(v)^T}N_k^{(v)}, \tag{6}$$

where ϕ is a diagonal matrix, whose diagonal entry is $1/(2\sqrt{\|R^{(v)}(i,:)\|_2^2 + \epsilon})$. A small constant $\epsilon > 0$ is used to avoid the trivial solution.

(3) Update error E: Similar to the subproblems C and $R^{(v)}$, the subproblem E is determined by keeping the other variables unchanged and omitting the unrelated variables. The optimization problem in Eq. (3) with respect to E is expressed as:

$$\begin{aligned}&\min_E \sum_{v=1}^{M}\beta\|E^{(v)}\|_{2,1} + \frac{\rho_k}{2}\|E^{(v)} - F_k^{(v)}\|_F^2,\\ &\min_E \frac{\beta}{\rho_k}\|E\|_{2,1} + \frac{1}{2}\|E - F_k\|_F^2,\end{aligned} \tag{7}$$

where $F_k^{(v)} = X^{(v)} - X^{(v)}C_{k+1} - X^{(v)}R_{k+1}^{(v)} + \frac{\Theta_K^{(v)}}{\rho_k}$, $F_k = [F_k^{(1)}; F_k^{(2)}; \cdots; F_k^{(M)}]$. The jth column of the optimal solution E_{k+1} is

$$E_{k+1}(:,j) = \begin{cases}\frac{\|F_k(:,j)\|_2 - \frac{\beta}{\rho_k}}{\|F_k(:,j)\|_2}F_k(:,j), & \text{if } \frac{\beta}{\rho_k} < \|F_k(:,j)\|_2;\\ 0, & \text{otherwise.}\end{cases} \tag{8}$$

(4) Update the Lagrange multiplier $\Theta^{(v)}$ **and penalty parameter** ρ: Finally, $\Theta_{k+1}^{(v)}$ and ρ_{k+1} can be updated by

$$\begin{aligned}&\Theta_{k+1}^{(v)} = \Theta_k^{(v)} + \rho_k(X^{(v)} - X^{(v)}C_{k+1} - X^{(v)}R_{k+1}^{(v)} - E_{k+1}^{(v)});\\ &\rho_{k+1} = \min\{\gamma * \rho_k, \rho_{max}\},\end{aligned} \tag{9}$$

where $\gamma > 1$ is to facilitate the convergence speed. ρ_{max} is the maximum value of the penalty parameter ρ.

Algorithm 1: GLSR for multi-view subspace clustering

Input: multi-view features: $\{X^{(v)}\}$; parameters: α, β, λ; nearest neighbors number 5; graph Laplacian matrices $\{L^{(v)}\}$;
Initialize: C_1, $R_1^{(v)}$, E_1, $\Theta_1^{(v)}$ initialized to $\mathbf{0}$; $\rho_1 = \epsilon = 10^{-3}$, $\gamma = 1.3$, $tol = 10^{-6}$, $k = 1$;
while not converged **do**
  Update C_{k+1} by Eq. (4);
  for $v = 1$ to M **do**
    Update $R_{k+1}^{(v)}$ by Eq. (6);
  end for
  Update E_{k+1} by Eq. (8);
  Update $\Theta_{k+1}^{(v)}$ and ρ_{k+1} by Eq. (9);
  Check the convergence condition:
    $\|X^{(v)} - X^{(v)}C - X^{(v)}R^{(v)} - E^{(v)}\|_\infty \le tol$;
end while
Output: C_k and $R_k^{(v)}$.

Remarks:

- Each subproblem $R^{(v)}$ is independent and thus can be updated in parallel;
- When we obtain the consensus matrix C shared among all views and the residual matrix $R^{(v)}$ of the vth view, we construct the affinity matrix by

$$S = \mathcal{P}(C) + \frac{1}{M}\sum_{v=1}^{M}\mathcal{P}(R^{(v)}), \tag{10}$$

  where $\mathcal{P}(x)$ is defined as $\mathcal{P}(x) = 0.5*(|x| + |x^T|)$;
- We use the k-NN to construct a symmetric weight matrix $W^{(v)}$. The vth graph Laplacian matrix $L^{(v)}$ is defined as $L = D^{(v)} - W^{(v)}$, where $D^{(v)}$ is a diagonal matrix and $D_{ii}^{(v)} = \sum_j W_{ij}^{(v)}$;
- The clustering results are yielded by performing the spectral clustering algorithm [17] on the affinity matrix S defined in Eq. (10).

3.3. Complexity analysis

In this subsection, we analyze the complexity of our proposed GLSR method. There are four subproblems shown in Algorithm 1. The main computational complexity of our proposed algorithm is dominated by updating variables: C, $R^{(v)}$ and $E^{(v)}$. The running times of updating C and $R^{(v)}$ are $\mathcal{O}(dn^2 + d^2n + d^3)$ and $\mathcal{O}(d_{(v)}n^2 + d_{(v)}^2n)$, respectively, where $d = d_{(1)} + d_{(2)} + \cdots + d_{(M)}$. For each iteration, updating $E^{(v)}$ needs $\mathcal{O}(d_{(v)}n^2)$ cost. The time complexities of updating the Lagrange multiplier $\Theta^{(v)}$ and penalty parameter ρ can be omitted compared with the other subproblems. Therefore, the total complexity of our proposed GLSR algorithm is $\mathcal{O}(dn^2 + d^2n + d^3)$.

4. Numerical experiments

In this section, we conduct experiments on eight real-world databases with five different categories: News stories, face images, generic object, leaves, and handwritten digits. We first describe the experimental settings and implementation details including the databases, competitors, and evaluation metrics. Then, the experimental results are reported. Finally, we perform a detailed analysis of the proposed GLSR method with respect to

Table 1
Summary of eight real multi-view databases.

Category	Database	Instance	cluster	View 1	View 2	View 3	View 4
News stories	BBC4view	685	5	4659*d*	4633*d*	4655*d*	4684*d*
	BBCSport	544	5	3183*d*	3203*d*	–	–
	3Sources	169	6	3560*d*	3631*d*	3068*d*	–
	Wikipedia	693	10	128*d*	10*d*	–	–
Face images	ORL	400	40	4096*d*	3304*d*	6750*d*	–
Generic object	COIL_20	1440	20	1024*d*	3304*d*	6750*d*	–
Leaves	100leaves	1600	100	64*d*	64*d*	64*d*	–
Handwritten Digits	UCI-3views	2000	10	240*d*	76*d*	6*d*	–

parameter selection and numerical convergence. All the experiments were performed on a Lenovo laptop with an Intel Core i3-3240T 2.30 GHz CPU that has 4 cores and 4GB of memory, running with Windows 8 and MATLAB R2013a.

4.1. Experimental settings and implementation details

Databases: Eight widely used real-world databases were selected to evaluate the clustering performance of the proposed GLSR method. Brief description of all databases are reported in Table 1. Detailed introductions of them are as follows: **BBC4view**:[1] It consists of 685 images of 5 object categories, each with 137 images and associated with four views. **BBCSport**:[2] It contains files from the BBC Sports website, which corresponds to sports news in 5 subject areas and is associated with 2 views. **3Sources**:[3] It consists of 169 news, which is reported by three news organizations, *i.e.*, BBC, Reuters, and Guardian. Each news was manually annotated with one of six topical labels. **Wikipedia**:[4] It consists of 693 images of 10 object categories associated with two views. **ORL**:[5] The ORL database contains 40 different subjects, each of which has 10 different images. Three types of features are used: intensity (view 1), LBP (view 2), and Gabor (view 3). **100leaves**:[6] One-hundred plant species leaves (100leaves) dataset contains 1600 samples from one hundred plant species. Three view features, including shape descriptor, fine scale margin and texture histogram are exploited. **COIL_20**:[7] It consists of 1440 images of 20 object categories, each with 72 images and is associated with three views. **UCI-3views**: It contains 2000 handwritten digits images with 10 classes. Three features including Fourier coefficients, pixel averages and morphological features are explored.

Compared methods: The proposed GLSR method is compared with the following state-of-the-art single-view and multi-view clustering methods: **SPC** [17]: single-view clustering method using standard spectral clustering; **SSC** [1]: single-view clustering method using the l_1-norm to learn a representation matrix; **LRR** [15]: single-view clustering method using the nuclear norm to learn a representation matrix; **LSR** [16]: single-view clustering method using the least squares regression to learn a representation matrix; **RMSC** [33]: multi-view clustering method using the low-rank and sparse matrix decomposition to learn a shared transition probability matrix; **LT-MSC** [27]: multi-view clustering method using the low-rank tensor constraint to learn a representation tensor; **ECMSC** [34]: multi-view clustering method to simultaneously exploit the representation exclusivity and indicator consistency; **MLAN** [37]: multi-view clustering method with adaptive neighbors; **GBS-KO** [12]: multi-view clustering by graph-based system; **GMC** [38]: graph-based multi-view clustering; **AWP** [14]: multi-view clustering via adaptively weighted procrustes. The first four methods are single-view clustering methods, while the other seven methods belong to multi-view clustering ones. For a fair comparison, the open source code of each competitor was used and parameter settings were followed the original papers.

Evaluation metrics: In order to evaluate the performance of the above different clustering methods, we use NMI (normalized mutual information), ACC (accuracy), Precision, Recall, F-score, AR (adjusted rand index) and Specificity metrics for quantitative comparison. To validate the statistic significance of comparison results, we perform the *t-test* to calculate the *p*-value on the clustering results of all methods. Since different metrics have different evaluation criteria for clustering, we present multiple metrics for a more comprehensive analysis. For seven metrics, a higher value indicates a better clustering quality.

4.2. Experimental results

This subsection conducts experiments on five categories of databases. The clustering results of all databases are shown in Tables 2 and 3, in which the best clustering results in each database are highlighted in bold font and the second best clustering results are underlined.

Table 2 provides a quantitative comparison between all methods on four News stories databases. For the BBC4view database, it can be observed that most of the compared methods have relatively unsatisfactory performance. However, our GLSR method achieved improvements of around 1.3%, 1.9%, 1.4%, 1.1%, −0.3%, 1.8% and 1.4% compared to the most competitive method AWP in terms of ACC, NMI, AR, F-score, Precision, Recall and Specificity , respectively. In addition, we can see that since the affinity matrix is constructed by the l_1-norm and the local geometrical structure is not considered, the ECMSC method usually has poor performance. For the BBCSport database, the two most competitive multi-view clustering methods MLAN, GMC and AWP have achieved considerable results and the single-view methods LRR and LSR also have competitive results. However, the proposed GLSR method has achieved optimal improvements.

Comparative experiments on the 3Sources database show that the methods SSC, LT-MSC and GBS-KO produced more impressive results. For all seven metrics, the proposed GLSR method still achieved significant improvements of around 10.7%, 5.3%, 12.9%, 9.6%, 3.8%, 6.8% and 6.1% compared to the competitive LT-MSC method. Thus, our method achieved higher performance on 3Sources database. For the Wikipedia database, we can see that the RMSC method shows the competitive performance due to the connection between spectral clustering and Markov chains. However, the proposed GLSR method displays the absolute advantage in all metrics. In addition, the *p*-value is very small on all databases, which means that our method outperforms all

1 http://mlg.ucd.ie/datasets/segment.html.
2 http://mlg.ucd.ie/datasets/segment.html.
3 http://mlg.ucd.ie/datasets/3sources.html.
4 http://lig-membres.imag.fr/grimal/data.html.
5 http://www.uk.research.att.com/facedatabase.html.
6 https://archive.ics.uci.edu/ml/datasets/One-hundred+plant+species+leaves+data+set
7 http://www.cs.columbia.edu/CAVE/software/softlib/.

Table 2
Comparison results on four databases.

Dataset	Method	ACC	NMI	AR	F-score	Precision	Recall	Specificity
BBC4view	SPC [17]	0.675 ± 0.000	0.447 ± 0.000	0.371 ± 0.000	0.522 ± 0.000	0.513 ± 0.000	0.531 ± 0.000	0.619 ± 0.000
	SSC [1]	0.660 ± 0.002	0.494 ± 0.005	0.470 ± 0.001	0.599 ± 0.001	0.578 ± 0.001	0.622 ± 0.001	0.621 ± 0.001
	LRR [15]	0.802 ± 0.000	0.568 ± 0.000	0.621 ± 0.000	0.712 ± 0.000	0.697 ± 0.000	0.727 ± 0.000	0.665 ± 0.000
	LSR [16]	0.815 ± 0.001	0.589 ± 0.001	0.608 ± 0.002	0.699 ± 0.001	0.701 ± 0.001	0.697 ± 0.001	0.702 ± 0.001
	RMSC [33]	0.775 ± 0.003	0.616 ± 0.004	0.560 ± 0.002	0.656 ± 0.002	0.703 ± 0.003	0.616 ± 0.001	0.614 ± 0.003
	LT-MSC [27]	0.591 ± 0.000	0.442 ± 0.005	0.400 ± 0.001	0.546 ± 0.000	0.525 ± 0.000	0.570 ± 0.001	0.568 ± 0.002
	ECMSC [34]	0.308 ± 0.028	0.047 ± 0.009	0.008 ± 0.018	0.322 ± 0.017	0.239 ± 0.009	0.497 ± 0.064	0.439 ± 0.049
	MLAN [37]	0.853 ± 0.007	0.698 ± 0.010	0.716 ± 0.005	0.783 ± 0.004	0.776 ± 0.003	0.790 ± 0.004	0.793 ± 0.006
	GBS-KO [12]	0.692 ± 0.000	0.561 ± 0.000	0.477 ± 0.000	0.632 ± 0.000	0.500 ± 0.000	0.859 ± 0.000	0.862 ± 0.000
	GMC [38]	0.693 ± 0.000	0.563 ± 0.000	0.479 ± 0.000	0.633 ± 0.000	0.501 ± 0.000	0.860 ± 0.000	0.864 ± 0.000
	AWP [14]	0.904 ± 0.000	0.761 ± 0.000	0.797 ± 0.000	0.845 ± 0.000	**0.838 ± 0.000**	0.851 ± 0.000	0.850 ± 0.000
	GLSR	**0.917 ± 0.000**	**0.780 ± 0.000**	**0.811 ± 0.000**	**0.856 ± 0.000**	0.835 ± 0.000	**0.878 ± 0.000**	**0.878 ± 0.000**
	p-value	1.509e−03	1.082e−03	5.929e−04	3.432e−04	8.454e−04	1.045e−03	1.135e−03
BBCSport	SPC [17]	0.719 ± 0.000	0.482 ± 0.002	0.460 ± 0.001	0.581 ± 0.001	0.616 ± 0.002	0.550 ± 0.001	0.553 ± 0.001
	SSC [1]	0.627 ± 0.003	0.534 ± 0.008	0.364 ± 0.007	0.565 ± 0.005	0.427 ± 0.004	0.834 ± 0.004	0.838 ± 0.004
	LRR [15]	0.836 ± 0.001	0.698 ± 0.002	0.705 ± 0.001	0.776 ± 0.001	0.768 ± 0.001	0.784 ± 0.001	0.783 ± 0.002
	LSR [16]	0.846 ± 0.002	0.629 ± 0.002	0.625 ± 0.003	0.719 ± 0.001	0.685 ± 0.002	0.756 ± 0.001	0.755 ± 0.001
	RMSC [33]	0.826 ± 0.001	0.666 ± 0.001	0.637 ± 0.001	0.719 ± 0.001	0.766 ± 0.001	0.677 ± 0.001	0.661 ± 0.035
	LT-MSC [27]	0.460 ± 0.046	0.222 ± 0.028	0.167 ± 0.043	0.428 ± 0.014	0.328 ± 0.028	0.629 ± 0.053	0.708 ± 0.058
	ECMSC [34]	0.285 ± 0.014	0.027 ± 0.013	0.009 ± 0.011	0.267 ± 0.020	0.244 ± 0.007	0.297 ± 0.045	0.252 ± 0.021
	MLAN [37]	0.721 ± 0.000	0.779 ± 0.000	0.591 ± 0.000	0.714 ± 0.000	0.567 ± 0.000	**0.962 ± 0.000**	**0.966 ± 0.000**
	GBS-KO [12]	0.806 ± 0.000	0.759 ± 0.000	0.721 ± 0.000	0.793 ± 0.000	0.725 ± 0.000	0.874 ± 0.000	0.876 ± 0.000
	GMC [38]	0.807 ± 0.000	0.760 ± 0.000	0.722 ± 0.000	0.794 ± 0.000	0.727 ± 0.000	0.875 ± 0.000	0.877 ± 0.000
	AWP [14]	0.809 ± 0.000	0.723 ± 0.000	0.726 ± 0.000	0.796 ± 0.000	0.743 ± 0.000	0.857 ± 0.000	0.857 ± 0.000
	GLSR	**0.873 ± 0.000**	**0.781 ± 0.000**	**0.803 ± 0.000**	**0.851 ± 0.000**	**0.837 ± 0.000**	0.865 ± 0.000	0.865 ± 0.000
	p-value	1.111e−02	1.688e−02	3.378e−03	3.235e−03	1.758e−03	4.730e−02	6.107e−02
3Sources	SPC [17]	0.564 ± 0.003	0.507 ± 0.002	0.361 ± 0.001	0.503 ± 0.001	0.526 ± 0.003	0.482 ± 0.004	0.479 ± 0.003
	SSC [1]	0.762 ± 0.003	0.694 ± 0.003	0.658 ± 0.004	0.743 ± 0.003	0.769 ± 0.001	0.719 ± 0.005	0.714 ± 0.006
	LRR [15]	0.647 ± 0.033	0.542 ± 0.018	0.486 ± 0.028	0.608 ± 0.033	0.594 ± 0.031	0.636 ± 0.096	0.634 ± 0.099
	LSR [16]	0.755 ± 0.011	0.687 ± 0.010	0.625 ± 0.023	0.705 ± 0.019	0.767 ± 0.016	0.653 ± 0.020	0.647 ± 0.018
	RMSC [33]	0.583 ± 0.022	0.630 ± 0.011	0.455 ± 0.031	0.557 ± 0.025	0.635 ± 0.029	0.497 ± 0.028	0.494 ± 0.025
	LT-MSC [27]	0.781 ± 0.000	0.698 ± 0.003	0.651 ± 0.003	0.734 ± 0.002	0.716 ± 0.008	0.754 ± 0.005	0.751 ± 0.008
	ECMSC [34]	0.346 ± 0.025	0.132 ± 0.029	0.011 ± 0.031	0.295 ± 0.013	0.240 ± 0.019	0.391 ± 0.043	0.426 ± 0.022
	MLAN [37]	0.775 ± 0.015	0.676 ± 0.005	0.580 ± 0.008	0.666 ± 0.007	0.756 ± 0.003	0.594 ± 0.009	0.585 ± 0.009
	GBS-KO [12]	0.692 ± 0.000	0.622 ± 0.000	0.443 ± 0.000	0.605 ± 0.000	0.484 ± 0.000	0.805 ± 0.000	0.810 ± 0.000
	GMC [38]	0.693 ± 0.000	0.622 ± 0.000	0.443 ± 0.000	0.605 ± 0.000	0.484 ± 0.000	0.804 ± 0.000	0.811 ± 0.000
	AWP [14]	0.757 ± 0.000	0.757 ± 0.000	0.621 ± 0.000	0.707 ± 0.000	0.721 ± 0.000	0.694 ± 0.000	0.690 ± 0.000
	GLSR	**0.888 ± 0.001**	**0.810 ± 0.001**	**0.787 ± 0.001**	**0.839 ± 0.001**	**0.807 ± 0.001**	**0.873 ± 0.001**	**0.872 ± 0.001**
	p-value	2.444e−04	1.968e−03	3.179e−04	1.694e−04	2.521e−03	2.008e−04	1.680e−04
Wikipedia	SPC [17]	0.567 ± 0.000	0.533 ± 0.000	0.429 ± 0.000	0.490 ± 0.000	0.499 ± 0.000	0.481 ± 0.000	0.478 ± 0.000
	SSC [1]	0.561 ± 0.001	0.527 ± 0.002	0.418 ± 0.001	0.481 ± 0.001	0.491 ± 0.001	0.471 ± 0.001	0.468 ± 0.001
	LRR [15]	0.554 ± 0.001	0.523 ± 0.001	0.417 ± 0.000	0.479 ± 0.000	0.490 ± 0.000	0.468 ± 0.001	0.466 ± 0.001
	LSR [16]	0.554 ± 0.001	0.523 ± 0.001	0.419 ± 0.001	0.479 ± 0.001	0.490 ± 0.001	0.468 ± 0.001	0.466 ± 0.001
	RMSC [33]	0.579 ± 0.018	0.534 ± 0.009	0.441 ± 0.016	0.501 ± 0.012	0.506 ± 0.027	0.498 ± 0.009	0.494 ± 0.016
	LT-MSC [27]	0.532 ± 0.003	0.496 ± 0.005	0.407 ± 0.005	0.471 ± 0.005	0.480 ± 0.004	0.461 ± 0.006	0.462 ± 0.004
	ECMSC [34]	0.563 ± 0.000	0.522 ± 0.000	0.413 ± 0.000	0.475 ± 0.000	0.494 ± 0.000	0.457 ± 0.000	0.457 ± 0.000
	MLAN [37]	0.203 ± 0.001	0.066 ± 0.000	0.020 ± 0.000	0.127 ± 0.000	0.127 ± 0.000	0.127 ± 0.000	0.124 ± 0.000
	GBS-KO [12]	0.447 ± 0.000	0.415 ± 0.000	0.143 ± 0.000	0.282 ± 0.000	0.189 ± 0.000	0.549 ± 0.000	**0.551 ± 0.000**
	GMC [38]	0.449 ± 0.000	0.417 ± 0.000	0.145 ± 0.000	0.284 ± 0.000	0.191 ± 0.000	**0.550 ± 0.000**	0.545 ± 0.000
	AWP [14]	0.573 ± 0.000	0.543 ± 0.000	0.434 ± 0.000	0.497 ± 0.000	0.493 ± 0.000	0.501 ± 0.000	0.499 ± 0.000
	GLSR	**0.587 ± 0.012**	**0.543 ± 0.009**	**0.448 ± 0.006**	**0.507 ± 0.005**	**0.516 ± 0.008**	0.497 ± 0.005	0.500 ± 0.005
	p-value	3.908e−02	8.794e−02	3.437e−02	3.520e−02	3.561e−02	2.769e−01	2.263e−01

competing methods with statistically significant level, and similar results are also shown in Table 3.

Table 3 shows the performance comparison of the different methods on the ORL, 100leaves, COIL_20, and UCI-3views databases. We obtain the following observations:

- Generally speaking, in most cases, the proposed GLSR achieves better or comparable performance than all competitors. Specifically, on all News stories databases and Generic object database COIL_20, the performance of GLSR is the best among all competing methods. On BBC4view and 3Source databases, the improvements over the best competing algorithms is at least 1.3% and 1.9% with respect to ACC and NMI values. On the other databases, the proposed GLSR is the second best among all methods. The superiority of GLSR over all competitors may come from the fact that the consensus among all views, residuals of different views, noise removal, and manifold regularization are integrated into a joint optimization model to yield a reliable affinity matrix.
- On all databases except 100leaves and UCI-3views, LSR outperforms LRR, in which LSR used the least squares regression to learn the representation matrix while LRR exploited the nuclear norm. This indicates that the least square regression is a better way to yield the grouping effect than the nuclear norm. This is also our original inspiration. In addition, RMSC and LT-MSC are two representative multi-view subspace clustering methods, both of which used the nuclear norm to handle the multi-view clustering task. However, their performance is unsatisfied on all databases over the proposed GLSR method.

Table 3
Comparison results on four databases.

Dataset	Method	ACC	NMI	AR	F-score	Precision	Recall	Specificity
ORL	SPC [17]	0.725 ± 0.025	0.884 ± 0.002	0.664 ± 0.005	0.664 ± 0.005	0.610 ± 0.006	0.728 ± 0.005	0.784 ± 0.005
	SSC [1]	0.765 ± 0.008	0.893 ± 0.007	0.694 ± 0.013	0.682 ± 0.012	0.673 ± 0.007	0.764 ± 0.005	0.823 ± 0.021
	LRR [15]	0.773 ± 0.003	0.895 ± 0.006	0.724 ± 0.020	0.731 ± 0.004	0.701 ± 0.001	0.754 ± 0.002	0.674 ± 0.003
	LSR [16]	0.787 ± 0.029	0.904 ± 0.010	0.719 ± 0.026	0.726 ± 0.025	0.684 ± 0.029	0.774 ± 0.024	0.778 ± 0.027
	RMSC [33]	0.723 ± 0.007	0.872 ± 0.012	0.645 ± 0.003	0.654 ± 0.007	0.607 ± 0.009	0.709 ± 0.004	0.663 ± 0.003
	LT-MSC [27]	0.795 ± 0.007	0.930 ± 0.003	0.750 ± 0.003	0.768 ± 0.004	0.766 ± 0.009	0.837 ± 0.005	0.810 ± 0.027
	ECMSC [34]	**0.854 ± 0.011**	**0.947 ± 0.009**	**0.810 ± 0.012**	**0.821 ± 0.015**	**0.783 ± 0.008**	**0.859 ± 0.012**	**0.925 ± 0.012**
	MLAN [37]	0.705 ± 0.022	0.854 ± 0.018	0.384 ± 0.010	0.376 ± 0.015	0.254 ± 0.021	0.721 ± 0.020	0.660 ± 0.010
	GBS-KO [12]	0.632 ± 0.000	0.856 ± 0.000	0.336 ± 0.000	0.361 ± 0.000	0.230 ± 0.000	0.801 ± 0.000	0.849 ± 0.000
	GMC [38]	0.633 ± 0.000	0.857 ± 0.000	0.337 ± 0.000	0.360 ± 0.000	0.232 ± 0.000	0.801 ± 0.000	0.850 ± 0.001
	AWP [14]	0.753 ± 0.000	0.908 ± 0.000	0.697 ± 0.000	0.705 ± 0.000	0.615 ± 0.000	0.824 ± 0.000	0.845 ± 0.000
	GLSR	0.827 ± 0.021	0.916 ± 0.012	0.758 ± 0.030	0.764 ± 0.030	0.729 ± 0.031	0.802 ± 0.030	0.819 ± 0.021
	p-value	1.610e−03	2.121e−02	2.119e−02	2.100e−02	2.534e−02	1.587e−01	2.577e−01
100leaves	SPC [17]	0.535 ± 0.011	0.748 ± 0.004	0.388 ± 0.011	0.394 ± 0.011	0.379 ± 0.012	0.411 ± 0.011	0.404 ± 0.013
	SSC [1]	0.627 ± 0.012	0.814 ± 0.005	0.508 ± 0.008	0.513 ± 0.008	0.479 ± 0.010	0.552 ± 0.009	0.542 ± 0.010
	LRR [15]	0.528 ± 0.011	0.752 ± 0.004	0.398 ± 0.009	0.404 ± 0.009	0.387 ± 0.008	0.423 ± 0.012	0.416 ± 0.008
	LSR [16]	0.496 ± 0.010	0.725 ± 0.005	0.346 ± 0.010	0.352 ± 0.010	0.335 ± 0.010	0.371 ± 0.010	0.346 ± 0.006
	RMSC [33]	0.711 ± 0.026	0.875 ± 0.008	0.630 ± 0.025	0.634 ± 0.025	0.595 ± 0.027	0.679 ± 0.022	0.673 ± 0.017
	LT-MSC [27]	0.736 ± 0.007	0.870 ± 0.006	0.641 ± 0.012	0.644 ± 0.012	0.615 ± 0.012	0.678 ± 0.013	0.659 ± 0.012
	ECMSC [34]	0.733 ± 0.005	0.863 ± 0.004	0.628 ± 0.009	0.632 ± 0.009	0.602 ± 0.009	0 0.665 ± 0.011	0.629 ± 0.016
	MLAN [37]	**0.883 ± 0.001**	0.950 ± 0.001	0.830 ± 0.001	0.823 ± 0.012	0.791 ± 0.018	0.858 ± 0.007	0.861 ± 0.006
	GBS-KO [12]	0.824 ± 0.000	0.934 ± 0.000	0.571 ± 0.000	0.577 ± 0.000	0.427 ± 0.000	**0.889 ± 0.000**	**0.898 ± 0.000**
	GMC [38]	0.824 ± 0.000	0.929 ± 0.000	0.497 ± 0.000	0.504 ± 0.000	0.352 ± 0.000	0.887 ± 0.000	0.891 ± 0.001
	AWP [14]	0.814 ± 0.000	0.920 ± 0.000	0.754 ± 0.000	0.757 ± 0.000	0.710 ± 0.000	0.810 ± 0.000	0.809 ± 0.000
	GLSR	0.871 ± 0.017	**0.953 ± 0.005**	**0.836 ± 0.017**	**0.838 ± 0.017**	**0.804 ± 0.022**	0.875 ± 0.014	0.875 ± 0.012
	p-value	1.941e−03	2.192e−03	1.488e−04	1.348e−04	1.069e−04	3.994e−03	3.980e−03
COIL_20	SPC [17]	0.672 ± 0.063	0.806 ± 0.008	0.619 ± 0.018	0.640 ± 0.017	0.596 ± 0.021	0.692 ± 0.013	0.668 ± 0.015
	SSC [1]	0.803 ± 0.022	0.935 ± 0.009	0.798 ± 0.022	0.809 ± 0.013	0.734 ± 0.027	0.804 ± 0.028	0.945 ± 0.025
	LRR [15]	0.761 ± 0.003	0.829 ± 0.006	0.720 ± 0.020	0.734 ± 0.006	0.717 ± 0.003	0.751 ± 0.002	0.719 ± 0.008
	LSR [16]	0.779 ± 0.012	0.862 ± 0.008	0.716 ± 0.017	0.731 ± 0.016	0.684 ± 0.028	0.786 ± 0.013	0.783 ± 0.013
	RMSC [33]	0.685 ± 0.045	0.800 ± 0.017	0.637 ± 0.044	0.656 ± 0.042	0.620 ± 0.057	0.698 ± 0.026	0.696 ± 0.037
	LT-MSC [27]	0.804 ± 0.011	0.860 ± 0.002	0.748 ± 0.004	0.760 ± 0.007	0.741 ± 0.009	0.776 ± 0.006	0.797 ± 0.008
	ECMSC [34]	0.782 ± 0.001	0.942 ± 0.001	0.781 ± 0.001	0.794 ± 0.001	0.695 ± 0.002	0.925 ± 0.001	0.928 ± 0.026
	MLAN [37]	0.862 ± 0.011	**0.961 ± 0.004**	0.835 ± 0.006	0.844 ± 0.013	0.758 ± 0.008	**0.953 ± 0.007**	0.931 ± 0.003
	GBS-KO [12]	0.790 ± 0.001	0.940 ± 0.000	0.783 ± 0.000	0.794 ± 0.000	0.693 ± 0.000	0.928 ± 0.000	0.933 ± 0.000
	GMC [38]	0.791 ± 0.001	0.941 ± 0.000	0.782 ± 0.000	0.794 ± 0.000	0.694 ± 0.000	0.929 ± 0.000	0.934 ± 0.001
	AWP [14]	0.650 ± 0.000	0.909 ± 0.000	0.695 ± 0.000	0.714 ± 0.000	0.573 ± 0.000	0.949 ± 0.000	**0.946 ± 0.000**
	GLSR	**0.872 ± 0.006**	0.940 ± 0.003	**0.836 ± 0.005**	**0.844 ± 0.005**	**0.789 ± 0.007**	0.907 ± 0.004	0.906 ± 0.004
	p-value	2.211e−04	1.889e−02	7.092e−04	7.541e−04	1.639e−04	4.381e−02	9.483e−02
UCI-3views	SPC [17]	0.732 ± 0.011	0.642 ± 0.005	0.545 ± 0.012	0.591 ± 0.011	0.582 ± 0.013	0.601 ± 0.011	0.614 ± 0.011
	SSC [1]	0.815 ± 0.011	0.840 ± 0.001	0.770 ± 0.005	0.794 ± 0.004	0.747 ± 0.010	0.848 ± 0.004	0.847 ± 0.000
	LRR [15]	0.871 ± 0.001	0.768 ± 0.002	0.736 ± 0.002	0.763 ± 0.002	0.759 ± 0.002	0.767 ± 0.002	0.784 ± 0.002
	LSR [16]	0.819 ± 0.000	0.863 ± 0.000	0.787 ± 0.000	0.810 ± 0.000	0.756 ± 0.000	0.872 ± 0.000	0.708 ± 0.000
	RMSC [33]	0.915 ± 0.024	0.822 ± 0.008	0.789 ± 0.014	0.811 ± 0.012	0.797 ± 0.017	.826 ± 0.006	0.812 ± 0.014
	LT-MSC [27]	0.803 ± 0.001	0.775 ± 0.001	0.725 ± 0.001	0.753 ± 0.001	0.739 ± 0.001	0.767 ± 0.001	0.763 ± 0.001
	ECMSC [34]	0.718 ± 0.001	0.780 ± 0.001	0.672 ± 0.001	0.707 ± 0.001	0.660 ± 0.001	0.760 ± 0.001	0.761 ± 0.001
	MLAN [37]	0.874 ± 0.000	**0.910 ± 0.000**	**0.847 ± 0.000**	**0.864 ± 0.000**	0.797 ± 0.000	**0.943 ± 0.000**	**0.875 ± 0.000**
	GBS-KO [12]	0.735 ± 0.000	0.814 ± 0.000	0.677 ± 0.000	0.711 ± 0.000	0.642 ± 0.000	0.798 ± 0.000	0.795 ± 0.000
	GMC [38]	0.736 ± 0.000	0.815 ± 0.000	0.678 ± 0.000	0.713 ± 0.000	0.644 ± 0.000	0.799 ± 0.000	0.797 ± 0.000
	AWP [14]	0.806 ± 0.000	0.842 ± 0.000	0.759 ± 0.000	0.785 ± 0.000	0.734 ± 0.000	0.842 ± 0.000	0.842 ± 0.000
	GLSR	**0.924 ± 0.001**	0.859 ± 0.001	0.841 ± 0.001	0.857 ± 0.001	**0.854 ± 0.001**	0.860 ± 0.001	0.860 ± 0.001
	p-value	1.169e−04	2.912e−02	8.361e−04	9.213e−04	6.498e−05	4.925e−02	4.890e−03

- Compared with the LT-MSC and ECMSC methods, the GLSR method significantly improves the subspace clustering performance, because it uses the manifold regularization to preserve the local geometrical structure of multiple features, while LT-MSC and ECMSC consider only the global structure.
- The advantage of our GLSR method over the recently proposed GBS-KO and GMC methods is very obvious on all databases. The GLSR has achieved at least 4.7% and 2.4% improvement with respect to ACC and NMI values over both of them. The reason is that both of the GBS-KO and GMC methods used the raw multi-view features to learn the similarity matrix, however, the raw data are usually contaminated by different noise and outliers. The proposed GLSR used the column-sparsity norm to achieve noise removal.

4.3. Model analysis

In this subsection, we analyze the parameter selection and numerical convergence on the 3Sources, BBC4view and BBCSport databases to fully understand the proposed GLSR method.

4.3.1. Parameter selection

We use several experiments to show the importance of three parameters including α, β, λ for the proposed method. α, β, λ are used to balance residual term, error term and manifold constraint and they are empirically selected from [0.001, 0.005, 0.01, 0.05, 0.1, 0.2, 0.4, 0.5], [0.01, 0.05, 0.1, 0.5, 1, 10] and [0.01, 0.1, 1, 10], respectively. Generally, the larger the parameter value, the greater the importance or impact on the corresponding metrics is.

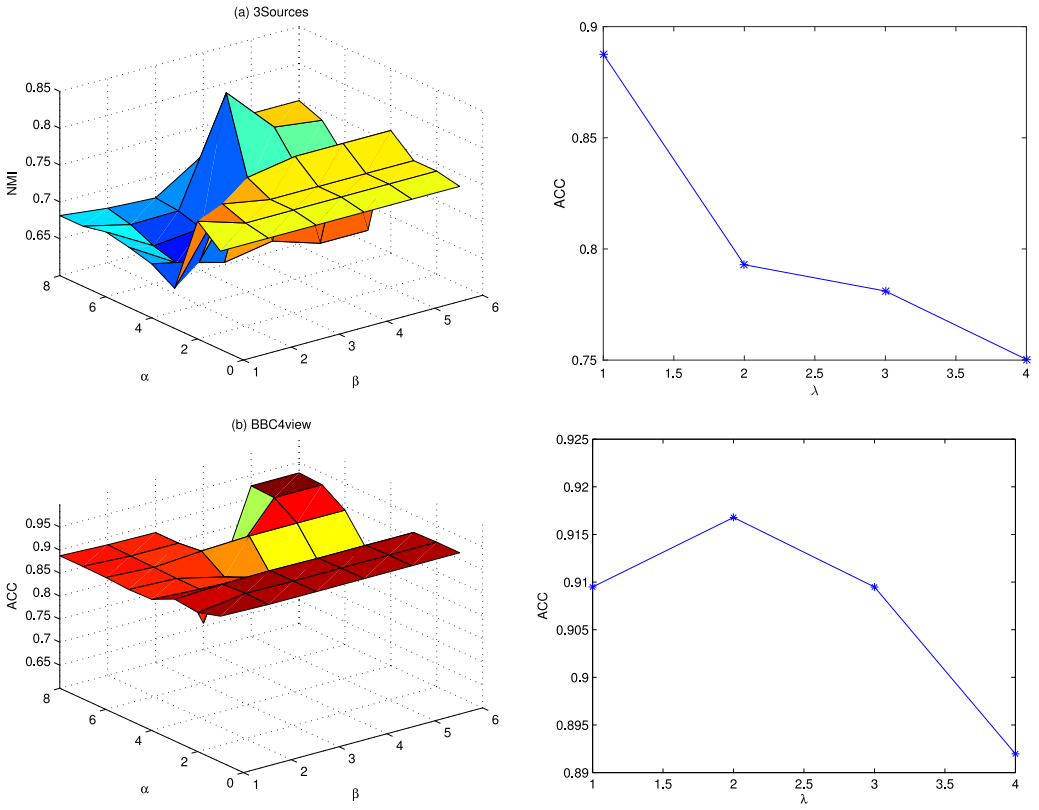

Fig. 2. ACC versus different values of parameters α and β (left), and parameter λ (right) on (a) 3Sources and (b) BBC4view.

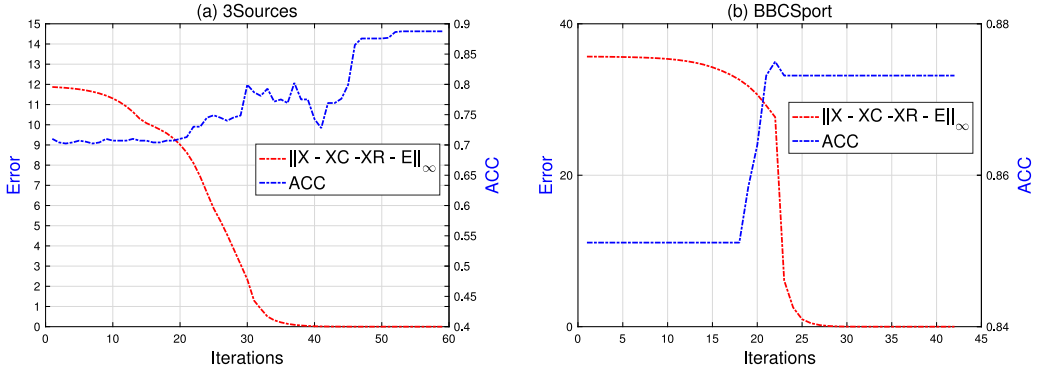

Fig. 3. Numerical convergence and ACC versus iterations on (a) 3Sources and (b) BBCSport.

The first column of Fig. 2 shows the relationship between ACC and the parameters α, β when the parameter λ is fixed. It can be seen that the parameter β has a greater impact on ACC to a certain extent, and when the parameters α, β are within the feasible range, the best clustering results can be obtained.

In the second column of Fig. 2, we first fix the parameters α, β and then assign different values to the parameter λ to show the effect of λ. We can see from the 3Sources database, the value of ACC is the largest when $\lambda = 0.01$, and the value of ACC reduces when λ becomes larger. For the BBC4view database, the ACC values have a peak at $\lambda = 0.1$. This is mainly because when the parameter λ is too large, the corresponding manifold constraint will play a leading role in the clustering, ignoring the global structure of the data.

4.3.2. Numerical convergence

This subsection investigates the numerical convergence of the proposed GLSR method. Fig. 3 shows the iterative error (red line) and ACC (blue line) curves on the 3Sources and BBCSport databases. The iterative error is calculated by $\|X^{(v)} - X^{(v)}C - X^{(v)}R^{(v)} - E^{(v)}\|_{\infty}$. One can be seen that our method converges stably after 40 iterations. In addition, during the optimization process, the ACC curve increases gradually as the number of iterations increases and stabilizes after several fluctuations. The above two figures have indicated that our algorithm has the strong numerical convergence.

5. Conclusions and future work

In this paper, we proposed a novel multi-view subspace clustering model named Graph-regularized Least Squares Regression (GLSR), which takes advantage of the tremendous information of the multiple features. The key point of the proposed GLSR method is that in order to obtain better grouping effect, the least squares regression is introduced instead of the nuclear norm to learn a

reliable affinity matrix. At the same time, the column-sparsity norm was used to obtain the residual information. In addition, the manifold regularization was adopted to preserve the local geometric structure of multiple features. Finally, we integrated all above terms into a unified framework and designed an effective algorithm based on the augmented Lagrangian multiplier method. The proposed GLSR method was compared with other state-of-the-art methods on eight real-world databases and the experimental results showed that GLSR method is superior to those eleven single-view and multi-view clustering methods.

In addition, our work also has certain limitations. Since the proposed method was originally designed to process the data located in multiple linear subspaces, the performance of the proposed GLSR method may be unsatisfactory when the multimedia data come from nonlinear subspace. Therefore, designing a method that can handle the data from multiple nonlinear and linear subspaces is the first focus of our future research work. The second research direction is that we will attempt to investigate more reliable priors such as the block diagonal property of the similarity matrix to further explore the underlying structures of multiple features. The last future research is to develop robust scalable algorithms to fit large-scale databases.

CRediT authorship contribution statement

Yongyong Chen: Writing - original draft, Data curation. **Shuqin Wang:** Formal analysis, Funding acquisition. **Fangying Zheng:** Formal analysis, Investigation. **Yigang Cen:** Writing - review and editing.

Acknowledgments

This work was supported by National Natural Science Foundation of China (Grant No. 11626143), and Natural Science Foundation of Zhejiang Province, China (Grant No. LY19A010025).

References

[1] E. Elhamifar, R. Vidal, Sparse subspace clustering: algorithm, theory, and applications, IEEE Trans. Pattern Anal. Mach. Intell. 35 (11) (2013) 2765–2781, http://dx.doi.org/10.1109/TPAMI.2013.57.

[2] C. Lu, J. Feng, Z. Lin, Subspace clustering by block diagonal representation, IEEE Trans. Pattern Anal. Mach. Intell. 41 (2) (2019) 487–501, http://dx.doi.org/10.1109/TPAMI.2018.2794348.

[3] Y. Chen, Z. Yi, Locality-constrained least squares regression for subspace clustering, Knowl.-Based Syst. 163 (2019) 51–56, http://dx.doi.org/10.1016/j.knosys.2018.08.014.

[4] X. Wang, Z. Lei, X. Guo, C. Zhang, H. Shi, S.Z. Li, Multi-view subspace clustering with intactness-aware similarity, Pattern Recognit. 88 (2019) 50–63, http://dx.doi.org/10.1016/j.patcog.2018.09.009.

[5] W. Zhu, J. Lu, J. Zhou, Nonlinear subspace clustering for image clustering, Pattern Recognit. Lett. 107 (2018) 131–136, http://dx.doi.org/10.1016/j.patrec.2017.08.023.

[6] Z. Ding, Y. Fu, Robust multi-view data analysis through collective low-rank subspace, Trans. Neural Netw. Learn. Syst. 29 (2017) 1986–1997, http://dx.doi.org/10.1109/TNNLS.2017.2690970.

[7] J. Gu, L. Jiao, F. Liu, Random subspace based ensemble sparse representation, Pattern Recognit. 74 (2018) 544–555, http://dx.doi.org/10.1016/j.patcog.2017.09.016.

[8] X. Wu, V. Kumar, J.R. Quinlan, J. Ghosh, Q. Yang, H. Motoda, G.J. McLachlan, A. Ng, B. Liu, S.Y. Philip, et al., Top 10 algorithms in data mining, Know. Inf. Syst. 14 (1) (2008) 1–37, http://dx.doi.org/10.1007/s10115-007-0114-2.

[9] P.G.A. Cornuéjols, C. Wemmert, Collaborative clustering: why, when, what and how, Inform. Fusion 39 (2018) 81–95, http://dx.doi.org/10.1016/j.inffus.2017.04.008.

[10] C. Peng, Z. Kang, Y. Hu, J. Cheng, Q. Cheng, Robust graph regularized nonnegative matrix factorization for clustering, ACM Trans. Knowl. Discov. Data 11 (3) (2017) 33, http://dx.doi.org/10.1145/3003730.

[11] F. Nie, J. Li, X. Li, Self-weighted multiview clustering with multiple graphs, in: Proc. Joint Conf. Artif. Intell., 2017, pp. 2564–2570, http://dx.doi.org/10.24963/ijcai.2017/357.

[12] H. Wang, Y. Yang, B. Liu, H. Fujita, A study of graph-based system for multi-view clustering, Knowl.-Based Syst. 163 (2019) 1009–1019, http://dx.doi.org/10.1016/j.knosys.2018.10.022.

[13] X. Peng, Z. Yu, Z. Yi, H. Tang, Constructing the l2-graph for robust subspace learning and subspace clustering, IEEE Trans. Cybern. 47 (4) (2016) 1053–1066, http://dx.doi.org/10.1109/TCYB.2016.2536752.

[14] F. Nie, L. Tian, X. Li, Multiview clustering via adaptively weighted procrustes, in: Proc. ACM SIGKDD Int. Conf. Knowl. Disc. Data Min., ACM, 2018, pp. 2022–2030.

[15] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, Y. Ma, Robust recovery of subspace structures by low-rank representation, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2013) 171–184, http://dx.doi.org/10.1109/TPAMI.2012.88.

[16] C. Lu, H. Min, Z. Zhao, L. Zhu, D. Huang, S. Yan, Robust and efficient subspace segmentation via least squares regression, in: Proc. Eur. Conf. Comput. Vis., 2012, pp. 347–360, http://dx.doi.org/10.1007/978-3-642-33786-4_26.

[17] A.Y. Ng, M.I. Jordan, Y. Weiss, On spectral clustering: Analysis and an algorithm, in: Proc. Neural Inf. Process. Syst., 2002, pp. 849–856.

[18] Y. Chen, X. Xiao, Y. Zhou, Jointly learning kernel representation tensor and affinity matrix for multi-view clustering, IEEE Trans. Multimed. 1 (2019) 1–13, http://dx.doi.org/10.1109/TMM.2019.2952984.

[19] R. Vidal, Subspace clustering, IEEE Signal Process. Mag. 28 (2) (2011) 52–68.

[20] R. Vidal, P. Favaro, Low rank subspace clustering (lrsc), Pattern Recognit. Lett. 43 (2014) 47–61, http://dx.doi.org/10.1016/j.patrec.2013.08.006.

[21] S. Gao, I.W.-H. Tsang, L.-T. Chia, Laplacian sparse coding, hypergraph laplacian sparse coding, and applications, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2012) 92–104, http://dx.doi.org/10.1109/TPAMI.2012.63.

[22] M. Yin, J. Gao, Z. Lin, Laplacian regularized low-rank representation and its applications, IEEE Trans. Pattern Anal. Mach. Intell. 38 (3) (2016) 504–517, http://dx.doi.org/10.1109/TPAMI.2015.2462360.

[23] Z. Kang, H. Pan, S.C. Hoi, Z. Xu, Robust graph learning from noisy data, IEEE Trans. Cybern. (2019) 1–11.

[24] Z. Kang, C. Peng, J. Cheng, Q. Cheng, Logdet rank minimization with application to subspace clustering, Comput. Intell. Neurosci. 2015 (2015) http://dx.doi.org/10.1155/2015/824289.

[25] Y. Chen, Y. Wang, M. Li, G. He, Augmented lagrangian alternating direction method for low-rank minimization via non-convex approximation, Signal Image Video Process. 11 (7) (2017) 1271–1278, http://dx.doi.org/10.1007/s11760-017-1084-9.

[26] S. Gu, Q. Xie, D. Meng, W. Zuo, X. Feng, L. Zhang, Weighted nuclear norm minimization and its applications to low level vision, Int. J. Comput. Vis. 121 (2) (2017) 183–208, http://dx.doi.org/10.1007/s11263-016-0930-5.

[27] C. Zhang, H. Fu, S. Liu, G. Liu, X. Cao, Low-rank tensor constrained multiview subspace clustering, in: Proc. IEEE Int. Conf. Comput. Vis., 2015, pp. 1582–1590, http://dx.doi.org/10.1109/ICCV.2015.185.

[28] Y. Zhang, Y. Yang, T. Li, H. Fujita, A multitask multiview clustering algorithm in heterogeneous situations based on LLE and LE, Knowl.-Based Syst. 163 (2019) 776–786.

[29] S. Li, M. Shao, Y. Fu, Multi-view low-rank analysis with applications to outlier detection, ACM Trans. Knowl. Discov. Data 12 (3) (2018) 32, http://dx.doi.org/10.1145/3168363.

[30] Q. Xiao, J. Dai, J. Luo, H. Fujita, Multi-view manifold regularized learning-based method for prioritizing candidate disease miRNAs, Knowl.-Based Syst. 175 (2019) 118–129.

[31] M. Brbić, I. Kopriva, Multi-view low-rank sparse subspace clustering, Pattern Recognit. 73 (2018) 247–258, http://dx.doi.org/10.1016/j.patcog.2017.08.024.

[32] K. Li, S. Li, Z. Ding, W. Zhang, Y. Fu, Latent discriminant subspace representations for multi-view outlier detection, in: Proc. AAAI, 2018, pp. 3522–3529.

[33] R. Xia, Y. Pan, L. Du, J. Yin, Robust multi-view spectral clustering via low-rank and sparse decomposition, in: Proc. AAAI, 2014, pp. 2149–2155.

[34] X. Wang, X. Guo, Z. Lei, C. Zhang, S.Z. Li, Exclusivity-consistency regularized multi-view subspace clustering, in: Proc. Conf. Comput. Vis. Pattern Recognit., 2017, pp. 923–931, http://dx.doi.org/10.1109/CVPR.2017.8.

[35] J. Liu, C. Wang, J. Gao, J. Han, Multi-view clustering via joint nonnegative matrix factorization, in: Proc. SIAM Int. Conf. Data Min., 2013, pp. 252–260.

[36] V.R. De Sa, Spectral clustering with two views, in: Proc. ICML workshop on learning with multiple views, 2005, pp. 20–27.

[37] F. Nie, G. Cai, J. Li, X. Li, Auto-weighted multi-view learning for image clustering and semi-supervised classification, IEEE Trans. Image Process. 27 (3) (2018) 1501–1511, http://dx.doi.org/10.1109/TIP.2017.2754939.

[38] H. Wang, Y. Yang, B. Liu, Gmc: graph-based multi-view clustering, IEEE Trans. Knowl. Data Eng. (2019) 1–14.

[39] W. Tang, Z. Lu, I.S. Dhillon, Clustering with multiple graphs, in: Proc. Int. Conf. Data Min., 2009, pp. 1016–1021, http://dx.doi.org/10.1109/ICDM.2009.125.

[40] M. Luo, C. Yan, Q. Zheng, X. Chang, L. Chen, F. Nie, Discrete multi-graph clustering, IEEE Trans. Image Process. 28 (9) (2019) 4701–4712.

[41] Y. Li, F. Nie, H. Huang, J. Huang, Large-scale multi-view spectral clustering via bipartite graph, in: Twenty-Ninth AAAI Conference on Artificial Intelligence, 2015.
[42] Y. Chen, X. Xiao, Y. Zhou, Multi-view clustering via simultaneously learning graph regularized low-rank tensor representation and affinity matrix, in: Proc. IEEE Int. Conf. Multimedia Expo, 2019, pp. 1348–1353, http://dx.doi.org/10.1109/ICME.2019.00234.
[43] Y. Xie, D. Tao, W. Zhang, Y. Liu, L. Zhang, Y. Qu, On unifying multi-view self-representations for clustering by tensor multi-rank minimization, Int. J. Comput. Vis. 126 (11) (2018) 1157–1179, http://dx.doi.org/10.1007/s11263-018-1086-2.
[44] Y. Luo, D. Tao, R. Kotagiri, C. Xu, Y. Wen, Tensor canonical correlation analysis for multi-view dimension reduction, IEEE Trans. Knowl. Data Eng. 27 (11) (2015) 3111–3124, http://dx.doi.org/10.1109/TKDE.2015.2445757.
[45] Y. Chen, S. Wang, Y. Zhou, Tensor nuclear norm-based low-rank approximation with total variation regularization, IEEE J. Sel. Topics Signal Process. 12 (6) (2018) 1364–1377, http://dx.doi.org/10.1109/JSTSP.2018.2873148.
[46] Y. Chen, X. Xiao, Y. Zhou, Low-rank quaternion approximation for color image processing, IEEE Trans. Image Process. 29 (1) (2019) 1426–1439, http://dx.doi.org/10.1109/TIP.2019.2941319.