Graph Regularized Subspace Clustering via Low-Rank Decomposition

Aimin Jiang, Weigao Cheng, Jing Shang, Xiaoyu Miao
College of Internet of Things Engineering
Hohai University
Changhzou, China
jiangam@hhuc.edu.cn, {chengwg, shangj, miaoxy}@hhu.edu.cn

Yanping Zhu
School of Information Science and Engineering
Changzhou University
Changzhou, China
zhuyanping@cczu.edu.cn

***Abstract*—Subspace clustering (SC) is able to identify low-dimensional subspace structures embedded in high-dimensional data. Recently, graph-regularized approaches aim to tackle this problem by learning a linear representation of data samples and also a graph structure in a unified framework. However, previous approaches exploit a graph embedding term based on representation matrix, which could over-smooth the graph structure and thus adversely affect the clustering performance. In this paper, we present a novel algorithm based on joint low-rank decomposition and graph learning from data samples. In graph learning, only a low-rank component of the representation matrix is employed to construct the graph embedding term. An alternating direction method of multipliers (ADMM) is further developed to tackle the resulting nonconvex problem. Experimental results on both synthetic data and real benchmark databases validate the effectiveness of the proposed SC algorithm.**

***Index Terms*—Affinity matrix, alternating direction method of multipliers (ADMM), graph Laplacian, low-rank decomposition, subspace clustering.**

I. Introduction

In many real-world applications, one needs to deal with a collection of high-dimensional data $\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_N] \in \mathbb{R}^{d\times N}$ generated from c low-dimensional subspaces $\{S_i\}_{i=1}^{c}$, where d and N represent the dimension and the number of data samples, respectively. Subspace clustering (SC) aims to cluster data samples in $\mathbf{X}$ into different classes in an unsupervised manner [1]. It has found various applications, such as image clustering [2], [11]-[14], and motion segmentation [4], [7], [10]. Existing SC approaches can be roughly classified into four groups: algebraic methods [3], statistical methods [4], iterative methods [5], and spectral-clustering-based methods [7]-[13].

Generally, spectral-clustering-based methods need first to construct a graph affinity matrix $\mathbf{B}$, whose element B_{ij} reflects the similarity between data samples i and j. A larger B_{ij} indicates a stronger connection and nodes i and j should be clustered into the same subspace. Then, spectral clustering is deployed to segment graph nodes [6]. Obviously, the step of graph learning plays an important role for the SC task.

Considering the graph learning problem, recent efforts are devoted to obtain the similarity graph through data representation [7]-[10]. Then, previous SC approaches aim to find a block-diagonal representation matrix $\mathbf{Z}$, that satisfies $\mathbf{X} = \mathbf{XZ} + \mathbf{E}$ where $\mathbf{E}$ denotes unobservable components or noise. But different assumptions of $\mathbf{Z}$ are adopted in these approaches. For example, supposing that each data sample can be reconstructed by a few data samples in the same cluster, sparse subspace clustering (SSC) [7] adopts the ℓ_1-norm regularizer in order to find a sparse representation $\mathbf{Z}$ for $\mathbf{X}$. Low-rank representation (LRR) [8] adopts the nuclear norm of $\mathbf{Z}$ to stabilize solutions, aiming to capture the underlying low-dimensional structure of data samples. Once obtaining $\mathbf{Z}$, graph affinity matrix $\mathbf{B}$ is further built by $\frac{1}{2}\left(|\mathbf{Z}| + \left|\mathbf{Z}^T\right|\right)$ or more sophisticated techniques. This category of SC approaches separate the learning of representation matrices and graph structures, thus yielding a depressed clustering performance. SC approaches developed in [11]-[13] adopt an optimization strategy of jointly learning $\mathbf{Z}$ and $\mathbf{B}$ through the graph embedding term $\mathrm{Tr}\left\{\mathbf{ZL}_B\mathbf{Z}^T\right\}$, where $\mathbf{L}_B = \mathbf{D} - \mathbf{W}$ is the graph Laplacian, $\mathbf{W} = \frac{1}{2}\left(\mathbf{B} + \mathbf{B}^T\right)$, and $\mathbf{D}$ is the degree matrix with diagonal elements $\mathbf{D}_{ii} = \sum_j \mathbf{W}_{ij}$. From the perspective of graph signal processing [15]-[16], minimizing $\mathrm{Tr}\left\{\mathbf{ZL}_B\mathbf{Z}^T\right\}$ inherently enhances the smoothness of a similarity graph. But this is different from the objective of data clustering. Thus, graphs obtained could be over-smoothed, leading to performance degradation.

As an attempt to address above issues, a novel SC algorithm is developed in this paper. To extract low-dimensional structures existing among data samples, low-rank decomposition of $\mathbf{Z}$ is incorporated in the proposed model. But only one of low-rank components of $\mathbf{Z}$ is directly exploited in graph learning. In this way, the learned graph cannot be over-smoothed. The proposed SC model can be further simplified and the resulting problem is tackled by an alternating direction method of multipliers (ADMM).

The remainder of this paper is organized as follows. The SC problem considered in this paper is first formulated in Section II. Then the ADMM is further developed. The performance of the proposed algorithm is evaluated in Section III. Conclusions

This work was supported in part by the National Key Research and Development Program under grant 2018AAA0100800, the National Nature Science Foundation of China under grants 61772090, 61701471, and 61801055, the Fundamental Research Funds for the Central Universities of China under grants 2018B23014 and 2018B47114, and the Key Development Program of Jiangsu Province of China under grants BE2017071, BE2017647, and BE2018004-04.

are summarized in section IV.

II. Proposed Algorithm

A. Problem Formulation

To enhance clustering performance, a number of SC approaches aim to learn a graph structure from data samples. To this end, the graph embedding term $\mathrm{Tr}\left\{\mathbf{Z}\mathbf{L}_B\mathbf{Z}^T\right\} = \frac{1}{2}\sum_{i,j}\|\mathbf{z}_i-\mathbf{z}_j\|_2^2 B_{ij}$ is widely adopted, where $\mathbf{z}_i$ is the ith column of $\mathbf{Z}$. If $\mathbf{Z}$ is considered as graph signals generated over different nodes of the estimated graph, minimizing the above term leads to a smoothed graph. However, the smoothness assumption cannot guarantee the success of data clustering. To address this issue, the proposed algorithm is developed based on a new graph learning strategy. It is known that a graph, whose Laplacian matrix is given by $\mathbf{L}_B$, can be clustered into c connected components, if the multiplicity of eigenvalue zero of its graph Laplacian is equal to c [14]. In the other words, the following problem should have zero objective function value

$$\sum_{i=1}^{c}\sigma_i(\mathbf{L}_B) = \min_{\mathbf{P}\in\mathbb{R}^{N\times c}} \mathrm{Tr}\left\{\mathbf{P}^T\mathbf{L}_B\mathbf{P}\right\},\ \text{s.t. } \mathbf{P}^T\mathbf{P}=\mathbf{I}, \quad (1)$$

where $\sigma_i(\mathbf{L}_B)$ denotes the ith smallest eigenvalue of $\mathbf{L}_B$.

As mentioned before, low-rank assumption is useful to to extract low-dimensional structures of samples, which facilitates data clustering. In this paper, we employ a direct decomposition $\mathbf{Z}=\mathbf{U}\mathbf{V}^T$ where $\mathbf{U},\mathbf{V}\in\mathbb{R}^{N\times K}$ ($K<N$). Since the low-rank decomposition is not unique, we further assume $\mathbf{V}^T\mathbf{V}=\mathbf{I}$ to reduce the ambiguity. Combining (1) with the low-rank decomposition, we finally have

$$\min_{\mathbf{Z},\mathbf{B}} \|\mathbf{X}-\mathbf{X}\mathbf{Z}\|_F^2 + \frac{\lambda_1}{K}\mathrm{Tr}\left\{\mathbf{V}^T\mathbf{L}_B\mathbf{V}\right\} + \lambda_2\|\mathbf{B}\|_F^2 \quad (2a)$$

$$\text{s.t. } \mathbf{Z}=\mathbf{U}\mathbf{V}^T,\ \mathbf{U},\mathbf{V}\in\mathbb{R}^{N\times K}, \quad (2b)$$

$$\mathbf{V}^T\mathbf{V}=\mathbf{I}, \quad (2c)$$

$$\mathbf{B}^T\mathbf{1}=\mathbf{1},\ B_{ij}\geq 0, \quad (2d)$$

where (2d) guarantees the probability property of each column of $\mathbf{B}$, $\mathbf{L}_B$ is the graph Laplacian constructed by affinity matrix $\mathbf{B}$, λ_1 and λ_2 are non-negative weights. As in [11], the third term in (2a) is introduced to avoid trivial solutions, otherwise only the nearest node of $\mathbf{x}_i$ is assigned as the neighbor with probability 1 and all the others with probabilities 0. Note that, in (2), K is much lower than N, but generally not equal to the number c of clusters.

In the above problem, low-rank components $\mathbf{U}$ and $\mathbf{V}$ play different roles. It is noticed that $\mathbf{U}$ is devoted to minimizing the representation error, and only $\mathbf{V}$ is involved in graph learning. Making use of (2c), one can further obtain

$$\|\mathbf{X}-\mathbf{X}\mathbf{Z}\|_F^2 = \mathrm{Tr}\left\{\left(\mathbf{I}-\mathbf{V}\mathbf{V}^T\right)\mathbf{X}^T\mathbf{X}\right\}. \quad (3)$$

Substituting (3) into (2) yields a simplified SC problem

$$\min_{\mathbf{V},\mathbf{B}} \mathrm{Tr}\left\{\mathbf{V}^T\tilde{\mathbf{L}}_B\mathbf{V}\right\} + \lambda_2\|\mathbf{B}\|_F^2 \quad (4a)$$

$$\text{s.t. } \mathbf{V}^T\mathbf{V}=\mathbf{I},\ \mathbf{V}\in\mathbb{R}^{N\times K}, \quad (4b)$$

$$\mathbf{B}^T\mathbf{1}=\mathbf{1},\ B_{ij}\geq 0, \quad (4c)$$

where $\tilde{\mathbf{L}}_B=\frac{\lambda_1}{K}\mathbf{L}_B-\mathbf{X}^T\mathbf{X}$. Given $\mathbf{B}$, (4) is reduced to an eigenvalue optimization problem, whose optimal solution is given by the eigen vectors corresponding to $\left\{\sigma_i(\tilde{\mathbf{L}}_B)\right\}_{i=1}^{K}$.

B. ADMM

Problem (4) is not convex in terms of $\mathbf{V}$ and $\mathbf{B}$. In this subsection, we shall develop an ADMM to tackle (4). To this end, we first introduce an auxiliary variable $\mathbf{Y}\in\mathbb{R}^{N\times K}$ in (4) to facilitate the subsequent optimization in each iteration

$$\min_{\mathbf{V},\mathbf{Y},\mathbf{B}} \mathrm{Tr}\left\{\mathbf{V}^T\tilde{\mathbf{L}}_B\mathbf{V}\right\} + \lambda_2\|\mathbf{B}\|_F^2 \quad (5a)$$

$$\text{s.t. } \mathbf{V}=\mathbf{Y}, \quad (5b)$$

$$\mathbf{Y}^T\mathbf{Y}=\mathbf{I},\ \mathbf{Y}\in\mathbb{R}^{N\times K}, \quad (5c)$$

$$\mathbf{B}^T\mathbf{1}=\mathbf{1},\ B_{ij}\geq 0. \quad (5d)$$

Then, the augmented Lagrangian is given by

$$\begin{aligned}\mathcal{L}(\mathbf{V},\mathbf{Y},\mathbf{B};\Pi) = {}& \mathrm{Tr}\left\{\mathbf{V}^T\tilde{\mathbf{L}}_B\mathbf{V}\right\} + \lambda_2\|\mathbf{B}\|_F^2 \\ & + \mathrm{Tr}\left\{\Pi^T(\mathbf{V}-\mathbf{Y})\right\} + \frac{\theta}{2}\|\mathbf{V}-\mathbf{Y}\|_F^2,\end{aligned} \quad (6)$$

where Π is the Lagrangian multiplier. Here, since constraints (5c) and (5d) are not relaxed, they do not appear in (6). The major steps of the ADMM are summarized below.

1) Update $\mathbf{V}$

$$\begin{aligned}\mathbf{V}^{(l+1)} = \arg\min_{\mathbf{V}} {}& \mathrm{Tr}\left\{\mathbf{V}^T\tilde{\mathbf{L}}_B^{(l)}\mathbf{V}\right\} \\ & + \frac{\theta}{2}\left\|\mathbf{V}-\mathbf{Y}^{(l)}+\frac{\Pi^{(l)}}{\rho^{(l)}}\right\|_F^2,\end{aligned} \quad (7)$$

whose optimal solution is given by

$$\mathbf{V}^{(l+1)} = \left(2\tilde{\mathbf{L}}_B^{(l)}+\theta^{(l)}\mathbf{I}\right)^{-1}\left(\theta^{(l)}\mathbf{Y}^{(l)}-\Pi^{(l)}\right). \quad (8)$$

2) Update $\mathbf{Y}$

$$\mathbf{Y}^{(l+1)} = \arg\min_{\mathbf{Y}^T\mathbf{Y}=\mathbf{I}}\left\|\mathbf{V}^{(l+1)}+\frac{\Pi^{(l)}}{\theta^{(l)}}-\mathbf{Y}\right\|_F^2. \quad (9)$$

whose optimal solution $\mathbf{Y}^{(l+1)}$ is achieved by replacing nonzero singular values of $\mathbf{V}^{(l+1)}+\frac{\Pi^{(l)}}{\theta^{(l)}}$ by ones.

3) Update $\mathbf{B}$

$$\mathbf{B}^{(l+1)} = \arg\min_{\mathbf{B}\in\mathcal{C}} \mathrm{Tr}\left\{\mathbf{V}^{(l+1)T}\mathbf{L}_B\mathbf{V}^{(l+1)}\right\} + \alpha\|\mathbf{B}\|_F^2, \quad (10)$$

where $\alpha=\frac{K\lambda_2}{\lambda_1}$, and $\mathcal{C}$ represents the set of left stochastic matrices, that is, $\mathcal{C}=\left\{\mathbf{A}\,\middle|\,\mathbf{A}^T\mathbf{1}=\mathbf{1}\,,\mathbf{A}\in\mathbb{R}^{N\times N}\right\}$. In practice, the above problem can be decomposed into a set of subproblems, that are solved independently

$$\min_{\mathbf{b}_j} \sum_{i=1}^{n} d_{ij}^{(l+1)}B_{ij} + \alpha B_{ij}^2 \quad (11a)$$

$$\text{s.t. } \mathbf{b}_j^T\mathbf{1}=1,\ \mathbf{b}_j\geq 0, \quad (11b)$$

where $d_{ij}^{(l+1)} = \frac{1}{2}\left\|\mathbf{v}_i^{(l+1)}-\mathbf{v}_j^{(l+1)}\right\|_2^2$, $\mathbf{b}_j$ is the jth column of $\mathbf{B}$, and $\mathbf{v}_i^{(l+1)}$ denotes the ith row of $\mathbf{V}^{(l+1)}$.

Further defining $\mathbf{d}_j = [d_{1j} \ \ldots \ d_{Nj}]^T$, the above problem can be cast as

$$\mathbf{b}_j^{(l+1)} = \underset{\mathbf{b}_j^T\mathbf{1}=1,\, \mathbf{b}_j \geq 0}{\operatorname{argmin}} \left\| \mathbf{b}_j + \frac{1}{2\alpha}\mathbf{d}_j \right\|_2^2, \tag{12}$$

whose optimal solution is given by [11], [14]

$$B_{ij}^{(l+1)} = \max\left\{ -\frac{1}{2\alpha} d_{ij}^{(l+1)} + \eta_j^{(l+1)}, 0 \right\}, \tag{13}$$

where $\eta_j^{(l+1)} = \frac{1}{k} + \frac{1}{2k\alpha}\sum_{i=1}^{k} \tilde{d}_{ij}^{(l+1)}$, and $\tilde{d}_{ij}^{(l+1)}$ is the ith element of $\mathbf{d}_j^{(l+1)}$ arranged in the ascending order. Here, each graph node is supposed to connect to the k nearest neighbors, yielding a sparse $\mathbf{B}$.

4) Update Π and θ

$$\Pi^{(l+1)} = \Pi^{(l)} + \theta^{(l)}\left(\mathbf{V}^{(l+1)} - \mathbf{Y}^{(l+1)}\right) \tag{14}$$

$$\theta^{(l+1)} = \min\left\{\rho\theta, \theta_{\max}\right\} \tag{15}$$

where $\rho > 1$, and $\theta_{\max}$ is the maximum value of θ.

The iterative procedure of the ADMM is terminated when $\frac{\left\|\mathbf{V}^{(l+1)} - \mathbf{V}^{(l)}\right\|_F}{\left\|\mathbf{V}^{(l)}\right\|_F} \leq \delta$ or $l > l_{\max}$, where δ and $l_{\max}$ are both predefined parameters.

III. Experimental Results

To evaluate the effectiveness of the proposed algorithm, several experiments using synthetic data and real benchmark image databases are presented in this section. For simplicity, we always choose $\lambda_1 = \lambda_2 = \lambda$ varies within $[10^{-6}, 1]$, parameter k is chosen from 2 to 10 , and set $\rho = 1.1$, $\theta^{(0)} = 1.01$, $\theta_{\max} = 10^{10}$, $\delta = 10^{-6}$, and $l_{\max} = 200$. Standard clustering accuracy (ACC) and normalized mutual information (NMI) are employed as evaluation metrics [17]. Because spectral clustering relies on the K-means method, whose performance is affected by initial points randomly selected, average ACC and NMI are computed by clustering results obtained by 50 independent implementations of spectral clustering for a given $\mathbf{B}$.

A. Synthetic Data

In this subsection, two clusters of data points are randomly distributed in the two-moon shape (see Fig. 1a). Each cluster consists of 100 samples. The coordinate of each data point is corrupted by white Gaussian noise $0.125 \cdot \mathcal{N}(0, 1)$. Both the proposed method and k-NN are used to cluster data points. Graphs constructed by k-NN using squared Euclidean pairwise distances and the proposed method are shown, respectively, in Figs. 1b and 1c. In our evaluation, parameters λ, k, and K are set to 1, 10, and 2, respectively. The number of the nearest neighbors in k-NN is chosen as 10, which leads to the best ACC. Clustering result of the proposed method is illustrated in Fig. 1d. It can be observed that the graph learned by our method has less edges connecting two different clusters than the one constructed simply by squared Euclidean pairwise distances. This indicates that graph learning to extract underlying structure embedded in data samples can facilitate data segmentation.

Table I: Description of real benchmark databases

Databases	# of samples	# of features	# of clusters
ExYaleB	2414	1024	38
USPS	1000	256	10
COIL100	1000	1024	100
AR	1400	2200	100

B. Image Clustering

In this subsection, four real benchmark databases including Extended Yale B (ExYaleB for short) [18], USPS [19], COIL100 [20], AR [21] are used to further evaluate clustering performance of the proposed method. Table I summaries the number of samples (N), the number of features (d) and the number of clusters (c). Six state-of-the-art approaches (i.e., CLR[1] [2], SSC [7], LRR[2] [8], LRSR [9], and SC-LALRG [13]) are also employed in our evaluation for performance comparison. Except CLR, all the approaches depend on spectral clustering to fulfill final data segmentation. The proposed approach and SC-LALRG employ the simultaneous learning strategy to obtain affinity matrices, while the others build $\mathbf{B}$ directly by $\mathbf{Z}$.

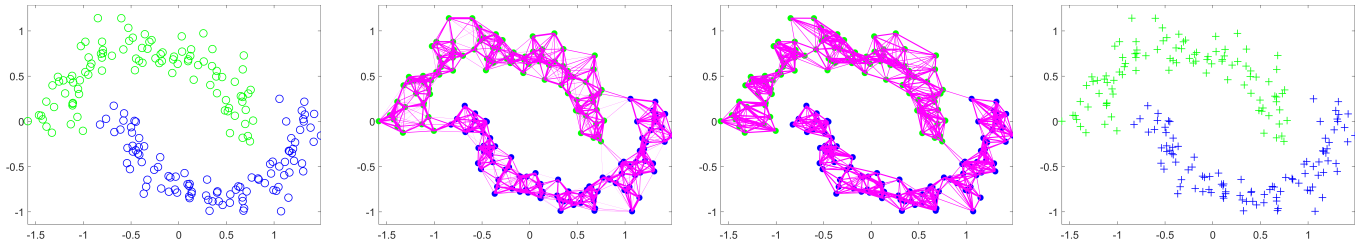
Table II shows experimental results in terms of ACC and NMI. Parameters of each SC approach are appropriately selected to achieve best performance. Best results are marked in bold font in Table II. Triplets (λ, k, K) of the best parameters for different databases are also listed in Table II. It can be observed that both SC-LALRG and the proposed approach generally outperform the others, because embedded graph terms are incorporated in their objective functions to learn underlying graph structures from data samples. Instead of the whole representation matrix $\mathbf{Z}$ as in SC-LALRG, our approach uses only one low-rank component of $\mathbf{Z}$, when constructing the embedded graph term in (2a). This strategy can avoid generating over-smoothed graphs and, thus, promote the reliability of graph segmentation by spectral clustering. In addition, the computation of the ADMM is carried directly on the orthogonal low-rank component $\mathbf{V}$, rather than $\mathbf{Z}$. Therefore, compared to some other approaches, its computational complexity is much lower. For instance, in the experiment of the COIL100 database, the CPU time of the proposed algorithm is 64.26s, that is much lower than that of LRR (218.25s), LRSR (375.06s) and SC-LALRG (170.05s).

C. Effects of Parameters on Clustering Accuracy

In the proposed algorithm, three parameters (i.e., λ, k, K) have to be appropriately set up. Three sets of experiments are conducted using ExYaleB database to analyze their effects on clustering performance. In each of them, we vary two parameters to evaluate clustering performance of the proposed method, while fixing the other one. The variations of ACC are depicted in Fig. 2. It can be noticed that inappropriate

[1]Two versions (denoted by CLR_1 and CLR_2) of CLR are used in our evaluation. They adopt ℓ_1 and ℓ_2 norms, respectively.

[2]Two versions of LRR using ℓ_1 and $\ell_{2,1}$ norms of $\mathbf{E}$ are developed in [8]. In our experiment, only the one using $\ell_{2,1}$ norm is employed in our evaluation for comparison.

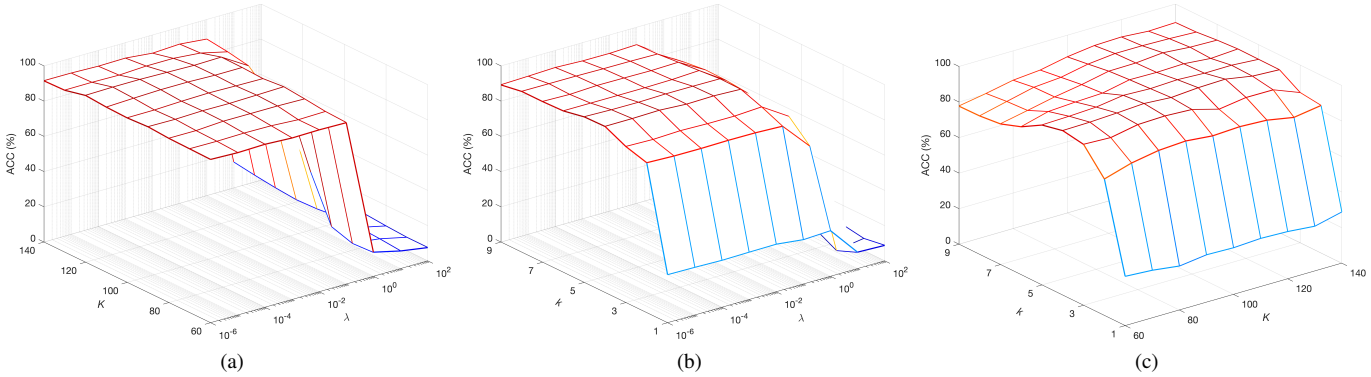


(a) (b) (c) (d)

Figure 1: Example of clustering result using synthetic data. (a) Original data points; (c) Graph learned by the k-NN algorithm; (c) Graph learned by the proposed algorithm; (d) Clustering result by the proposed algorithm.

Table II: Experimental results in terms of ACC and NMI (%) on real benchmark databases

	ACC/NMI						
	CLR_1 (k)	CLR_2 (k)	SSC (α)	LRR (λ)	LRSR (λ_1, λ_2)	SC-LALRG (λ, k)	Proposed (λ, k, K)
ExYaleB	36.99/52.76 (5)	41.88/56.25 (5)	66.37/70.38 (100)	69.18/74.13 (2)	69.32/74.36 (0.01, 1)	88.91/90.00 (1, 5)	**95.03/94.81** (10^{-4}, 4, 112)
USPS	81.60/84.16 (8)	81.40/83.36 (10)	78.50/77.15 (5)	75.89/68.73 (0.1)	81.24/70.97 (0.01, 0.1)	89.10/82.10 (1, 4)	**90.43/83.57** (0.1, 8, 19)
COIL100	**94.70/98.83** (4)	94.70/98.83 (4)	85.57/96.69 (20)	76.44/92.73 (0.5)	77.78/93.29 (10, 10)	94.51/98.34 (1, 4)	94.60/98.42 (0.01, 3, 34)
AR	36.57/60.29 (6)	36.79/63.24 (10)	64.81/84.26 (100)	82.55/92.43 (3)	77.23/92.69 (10^{-3}, 1)	82.89/91.35 (0.1, 9)	**87.08/93.35** (0.1, 8, 91)



(a) (b) (c)

Figure 2: Parameter effects on ExYaleB database. (a) Variation of ACC w. r. t. λ and K with $k = 4$; (b) Variation of ACC w. r. t. λ and k with $K = 110$; (c) Variation of ACC w. r. t. K and k with $\lambda = 10^{-4}$.

parameters could lead to the degradation of clustering performance. But the fluctuation of clustering accuracy is generally small when parameters are chosen within an appropriate range. Fig. 2 shows that, for ExYaleB database, the proposed method achieves steady performance within $\lambda \in [10^{-6}, 10^{-1}]$, $k \in [4, 9]$, and $K \in [100, 140]$.

D. Convergence Analysis

Although there is no theoretical guarantee of global convergence for the ADMM, experimental results show that the proposed algorithm always converges. Fig. 3 illustrates the variation of relative update $\left\|\mathbf{V}^{(l+1)} - \mathbf{V}^{(l)}\right\|_F / \left\|\mathbf{V}^{(l)}\right\|_F$ with respect to iteration index in the experiment of the COIL100 database. Obviously, the violent fluctuation is only observed in the first few iterations, and then the iterative procedure graduately converges to the final solution.

IV. Conclusions

A novel subspace clustering model based on joint low-rank decomposition and graph learning has been proposed in this paper. An explicit low-rank decomposition $\mathbf{Z} = \mathbf{U}\mathbf{V}^T$ is adopted, where only low-rank component $\mathbf{V}$ is involved in the graph embedding term. For the ease of graph partition, orthogonality property is also imposed on $\mathbf{V}$. Compared to other graph-learning-based approaches, this strategy can avoid the risk of learning a over-smoothed graph structure, thus enhancing its clustering performance. The proposed model can be further simplified and the ADMM has been developed for this challenging problem. Experimental results obtained

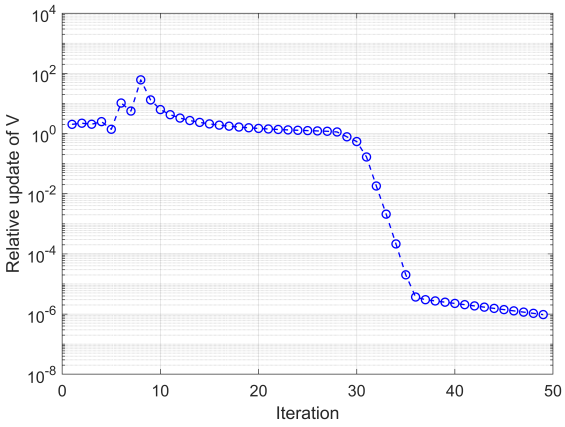


Figure 3: Variation of relative update of $\mathbf{V}$ w. r. t. iteration index on COIL100

on synthetic data and real benchmark databases have demonstrated the superior performance of our SC approach.

REFERENCES

[1] L. Parsons, E. Haque, and H. Liu, "Subspace clustering for high dimensional data: A review," *ACMSIGKDD Explorations Newslett*., vol. 6, no. 1, pp. 90-105, Jun. 2004.

[2] F. Nie, X. Wang, M. I. Jordan, and H. Huang, "The constrained laplacian rank algorithm for graph-based clustering," *in Proc. AAAI*, 2016, pp. 1969-1976.

[3] J. P. Costeira and T. Kanade, "A multibody factorization method for independently moving objects," *Int. J. Comput. Vis.*, vol. 29, no. 3, pp. 159-179, 1998.

[4] S. Rao, R. Tron, R. Vidal, and Y. Ma, "Motion segmentation via robust subspace separation in the presence of outlying, incomplete, or corrupted trajectories," *in Proc. of 21th IEEE Conf. Comput. Vis. and Pattern Recognit.*, Jun. 2008, pp. 1-8.

[5] L. Lu and R. Vidal, "Combined central and subspace clustering for computer vision applications," *in Proc. of 23th Int. Conf. Machine Learn.*, Jun. 2006, pp. 593-600.

[6] U. von Luxburg, "A tutorial on spectral clustering," *Statist. Comput*., vol. 17, no. 4, pp. 395-416, 2007.

[7] E. Elhamifar and R. Vidal, "Sparse subspace clustering: Algorithm, theory, and applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 11, pp. 2765-2781, 2013.

[8] G. Liu, Z. Lin, S. Yan, J. Sun, and Y. Ma, "Robust recovery of subspace structures by low-rank representation," *IEEE Trans. Pattern Anal. Mach. Intell*., vol. 35, no. 1, pp. 171-184, Jan. 2013.

[9] J. Wang, D. Shi, D. Cheng, Y. Zhang, and J. Gao, "LRSR: low-rank-sparse representation for subspace clustering," *Neurocomputing*, vol. 214, pp. 1026-1037, 2016.

[10] C. Lu, H. Min, Z.-Q. Zhao, L. Zhu, D.-S. Huang, and S. Yan, "Robust and efficient subspace segmentation via least squares regression," *in Proc. Eur. Conf. Comput.Vis*., 2012, pp. 347-360.

[11] X. Guo, "Robust subspace segmentation by simultaneously learning data representations and their affinity matrix," *in Proc. 24th Int. Joint Conf. Artif. Intell.*, 2015, pp. 3547-3553.

[12] M. Yin, Z. Wu, D. Zeng, P. Li, and S. Xie, "Sparse subspace clustering with jointly learning representation and affinity matrix," *Journal of the Franklin Institute*, vol. 355, no. 8, 2018.

[13] M. Yin, S. Xie, Z. Wu, Y. Zhang, and J. Guo, "Subspace clustering via learning an adaptive low-rank graph," *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3716-3728, 2018.

[14] F. Nie, X. Wang, and H. Huang, "Clustering and projected clustering with adaptive neighbors," *in Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2014, pp. 977-986.

[15] G. B. Giannakis, Y. Shen, and G. V. Karanikolas, "Topology identification and learning over graphs," *Proc. IEEE*, vol. 106, no. 5, pp. 787-807, 2018.

[16] X. Dong, D. Thanou, M. Rabbat, and P. Frossard, "Learning graphs from data: A signal representation perspective," *IEEE Signal Process. Mag*., vol. 36, no. 3, pp. 44-63, 2019.

[17] M. Zheng *et al.*, "Graph regularized sparse coding for image representation," *IEEE Trans. Image Process.*, vol. 20, no. 5, pp. 1327-1336, May. 2011.

[18] K. C. Lee, J. Ho, and D. J. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting," *IEEE Trans. Pattern Anal. Mach. Intell*., vol. 27, no. 5, pp. 684-698, 2005.

[19] J. J. Hull, "A database for handwritten text recognition research," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 5, pp. 550-554, 1994.

[20] S. A. Nene, S. K. Nayar and H. Murase, "Columbia Object Image Library (COIL-100)," *Technical Report CUCS-006-96*, February 1996.

[21] A.M. Martinez and R. Benavente, "The AR face database," *CVC Tech. Report*, no. 24, 1998.