Neurocomputing 378 (2020) 153–165

Contents lists available at ScienceDirect

Neurocomputing

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

Robust optimal graph clustering

Fei Wang[a], Lei Zhu[a], Cheng Liang[a,*], Jingjing Li[b], Xiaojun Chang[c], Ke Lu[b]

[a] School of Information Science and Engineering, Shandong Normal University, China
[b] School of Computer Science and Engineering, University of Electronic Science and Technology of China, China
[c] School of Information Technology, Monash University, Australia

ARTICLE INFO

Article history:
Received 21 December 2018
Revised 4 July 2019
Accepted 6 July 2019
Available online 18 October 2019

Communicated by Choon Ki Ahn

Keywords:
Graph-based clustering
Sparse reconstruction
Similarity graph
Adaptive neighbors

ABSTRACT

Most graph-based clustering methods separate the graph construction and clustering into two independent processes. The manually pre-constructed graph may not be suitable for the subsequent clustering. Moreover, as real world data generally contains noises and outliers, the similarity graph directly learned from them will be unreliable and further impair the subsequent clustering performance. To tackle the problems, in this paper, we propose a novel clustering framework where a robust graph is learned with noise removal, and simultaneously, with desirable clustering structure. To this end, we first learn a discriminative representation of data samples via sparse reconstruction. Then, a robust graph is automatically constructed with adaptive neighbors to each data sample. Simultaneously, a reasonable rank constraint is imposed on the Laplacian matrix of similarity graph to pursue the ideal clustering structure, where the number of connected components in the learned graph is exactly equal to the number of clusters. We finally derive an alternate optimization algorithm guaranteed with convergence to solve the formulated unified learning framework to achieve better prediction accuracy. Experiments on both synthetic and real datasets demonstrate the superior performance of the proposed method compared with several state-of-the-art clustering techniques.

1. Introduction

Clustering partitions the data samples into disjoint clusters, where samples from the same cluster share high similarity with each other, and vice versa. It is a fundamental topic in both data mining and pattern recognition fields. Due to the effectiveness of capturing the complex structure hidden in the data, graph-based clustering methods have been widely investigated [1–4]. In general, traditional approaches are performed by first constructing a weighted undirected graph to measure the pair-wise similarities of data samples, and then accomplishing the clustering based on spectral graph analysis. Under such circumstance, the similarity graph derived from raw data will remain constant in the subsequent clustering process. However, the pre-constructed graph with unguided manual parameter setting may not be appropriate for clustering, which will lead to suboptimal clustering performance.

The key point of graph-based clustering performance is to construct a high-quality similarity graph that could accurately capture the intrinsic sample relations. Recently, varieties of graph-based clustering algorithms have been proposed. For example, Clustering with Adaptive Neighbors (CAN) [5] learns the data similarity matrix and clustering structure simultaneously. The data similarity matrix is learned by adaptive neighbors assignment. Besides, a rank constraint is imposed on the Laplacian matrix of the data similarity matrix to obtain a desirable clustering structure. The Constrained Laplacian Rank (CLR) [6] learns a block diagonal data similarity matrix such that the clustering results can be immediately obtained. Multi-View Clustering and Semi-Supervised Classification with Adaptive Neighbors (MLAN) [7] performs clustering/semi-supervised classification and local structure learning simultaneously. The obtained graph from raw multi-view features can be partitioned into specific clusters. Locally Consistent Concept Factorization (LCCF) [8] models the data space as a submanifold embedded in the ambient space and performs the concept factorization on this manifold in question. As a result, LCCF has more discriminating power than the ordinary NMF [9] which only considers the Euclidean structure of the data. More recently, Orthogonal and Nonnegative Graph Reconstruction (ONGR) [10] is designed based on the graph reconstruction. It imposes orthogonal and nonnegative constraints on Normalized Cut [3], so that the reconstructed graph can possess clear cluster structure. Learning with Adaptive Neighbors for Image Clustering (LAN) [11] is proposed to learn a graph based on the given data graph such that the new obtained graph is more suitable for the

[*] Corresponding author.
E-mail address: alcs417@sdnu.edu.cn (C. Liang).

https://doi.org/10.1016/j.neucom.2019.07.102
0925-2312/© 2019 Elsevier B.V. All rights reserved.

clustering task. All these methods have made important progress on enhancing the quality of the constructed similarity graph. Nevertheless, as real data generally contains noises and outliers, directly constructing graph on raw data may damage the intrinsic data structure and thus impair the clustering performance.

In this paper, we propose a novel clustering model, called *robust optimal graph clustering* (ROGC). We aim to learn a robust structured graph with noise removal, and simultaneously, with desirable clustering structure. Specifically, we first identify a set of basis vectors by sparse reconstruction to represent the raw data as the linear combination of the basis vectors. In this way, the corresponding reconstruction coefficient matrix is robust to adverse noises and outliers. It possesses enhanced discriminative capability and is determined as the new representation of raw data. Then, we learn an optimal graph by assigning the adaptive neighbors of each data point in the new transformed space. Simultaneously, a reasonable rank constraint is imposed on the Laplacian matrix of the graph to pursue the ideal clustering structure, where the learned graph contains exactly the same number of connected components as the number of clusters. The main contributions of this paper are summarized as follows:

- We propose a unified clustering framework to adaptively learn an optimal graph with desirable clustering structure in the robust representation space. And with the rank constraint, this graph can be directly used for clustering without requiring any post-processing to calculate the cluster indicators.
- We transform the challenging optimization problem into an equivalent one that can be tackled more easily. An alternate optimization method is proposed to iteratively solve the problem.
- Extensive experiments on both synthetic and real datasets demonstrate the superior performance of our method, and also validate the advantages of robust representation learning, as well as optimal graph learning on the clustering performance.

The remainder of this paper is organized as follows: We give details on the objective function and optimization algorithm in Section 2. Computational complexity analysis and convergence analysis are presented in Section 3. Section 4 provides the experimental configuration. Experimental results are presented in Section 5. Section 6 finally concludes the paper.

2. The proposed method

In this section, we detail the proposed method. We first briefly introduce the relevant notations used in this paper. Next, we formulate the overall objective function and present an alternate optimization algorithm to solve it.

2.1. Notations

Throughout the paper, all the matrices are written in uppercase and italic. For an arbitrary matrix M, the ith row (with transpose) and the (i, j)th element are denoted by m_i and m_{ij}, respectively. M^T is the transpose of M and $Tr(M)$ is the trace of M. The Frobenius norm of M is denoted by $\|M\|_F$. $\mathbf{1}$ denotes a column vector with all elements equal to one. We summarize the main notations in Table 1.

2.2. Objective formulation

Most graph-based clustering methods simply adopt the fixed graph pre-constructed from raw data samples to perform clustering. The manually constructed graph without any guidance may not be appropriate for clustering. Moreover, the noises and outliers existed in the raw data may damage the intrinsic graph structure and impair the clustering performance. In order to alleviate the impacts of adverse noises and obtain a robust structured graph for clustering, we develop a unified learning framework which simultaneously reconstructs data with noise removal and performs the structured graph learning. The framework is expected to satisfy the following objectives, i.e., minimizing the impact of adverse noises, preserving the local structure of the data, and learning a well structured graph that can be used for clustering directly.

Table 1
Main notations used in this paper.

Notation	Description
$X=[x_1,\ldots,x_n]\in R^{d\times n}$	Raw data matrix
$B\in R^{d\times m}$	Basis matrix
$S\in R^{m\times n}$	Coefficient matrix
$W\in R^{n\times n}$	Weight matrix (similarity matrix)
$F\in R^{n\times c}$	Cluster indicator matrix
$x_i\in R^{d\times 1}$	The ith data point
G	The learned graph
d	Dimension of raw data
n	The number of data samples
m	The number of basis vectors
k	The number of neighbors
c	The number of clusters

Given a data matrix $X=[x_1, x_2, \ldots, x_n]\in\Re^{d\times n}$, where each x_i represents a data point. We devote to learn a basis matrix $B=[b_1, b_2, \ldots, b_m]\in\Re^{d\times m}$, where each b_i represents a basis vector, and a coefficient matrix $S=[s_1, s_2, \ldots, s_n]\in\Re^{m\times n}$, where each s_i is a sparse representation for a data point. Each data point x_i can be represented as a sparse linear combination of basis vectors, and the coefficient matrix can be considered as a new representation of the raw data. Specifically, the part of data reconstruction is formulated as

$$\min_{S,B} \|X-BS\|_F^2+\beta\sum_{i=1}^{n}\|s_i\|_1 \quad s.t. \forall i, \|b_i\|^2\le c \tag{1}$$

where β is a regularization parameter and it is used to avoid the issue of overfitting, $\|\cdot\|_F$ measures the data reconstruction error, and $\sum_{i=1}^{n}\|s_i\|_1$ measures the sparseness of s_i. In addition, $\|b_i\|^2\le c$ is to prevent B from having arbitrarily large values, which would lead to very small values of S.

Minimizing the objective function in Eq. (1) can reduce the impact of adverse noises and enhance the discriminative capability of data representation. Therefore, the coefficient matrix S can be considered as a more discriminative representation of raw data. Motivated by its effectiveness, in ROGC, we consider learning a graph with desirable structure on the obtained sparse representation S. Specifically, we aim to learn a nearest neighbor graph G with n vertices, where each vertex represents a data point s_i and the weight of each edge w_{ij} is the possibility of connecting s_i and s_j. Such probability can be regarded as the similarity between two data points. For simplicity, we use the Euclidean distance $\|s_i-s_j\|_2^2$ as the similarity measure. Generally, a smaller distance should be assigned with a larger probability w_{ij}, and vice versa. Thus, the similarity matrix $W\in\Re^{n\times n}$ can be determined by solving the following problem

$$\min_{W}\sum_{i,j=1}^{n}(\|s_i-s_j\|_2^2 w_{ij}+\gamma w_{ij}^2)$$
$$s.t. \forall i, w_i^T\mathbf{1}=1, 0\le w_{ij}\le 1 \tag{2}$$

where γ is the regularization parameter. The regularization term $\sum_{i,j=1}^{n} w_{ij}^2$ is used to avoid the trivial solution that two points which are nearest to each other become neighbor with probability 1. In spectral analysis, $L_W=D-W$ is called Laplacian matrix, where the degree matrix D is defined as a diagonal matrix whose ith diagonal element is $\sum_j(w_{ij}+w_{ji})/2$. According to the definition of Laplacian matrix, we have

$$Tr(SL_WS^T) = Tr(SDS^T) - Tr(SWS^T) = \sum_{i=1}^{n} d_i s_i^T s_i - \sum_{i,j=1}^{n} w_{ij} s_i^T s_j$$
$$= \frac{1}{2}(\sum_{i=1}^{n} d_i s_i^T s_i - 2\sum_{i,j=1}^{n} w_{ij} s_i^T s_j + \sum_{j=1}^{n} d_j s_j^T s_j)$$
$$= \frac{1}{2}\sum_{i,j=1}^{n} \|s_i - s_j\|_2^2 w_{ij}. \tag{3}$$

Then, Eq. (2) can be rewritten as:

$$\min_W 2Tr(SL_WS^T) + \gamma \sum_{i,j=1}^{n} w_{ij}^2$$
$$s.t. \forall i, w_i^T \mathbf{1} = 1, 0 \le w_{ij} \le 1. \tag{4}$$

By comprehensively considering the above two parts, we derive the preliminary objective formulation of ROGC as

$$\min_{S,B,W} \|X - BS\|_F^2 + 2\alpha Tr(SL_WS^T) + \beta \sum_{i=1}^{n} \|s_i\|_1 + \gamma \sum_{i,j=1}^{n} w_{ij}^2$$
$$s.t. \forall i, w_i^T \mathbf{1} = 1, 0 \le w_{ij} \le 1, \|b_i\|^2 \le c \tag{5}$$

where $\alpha,\ \beta,\ \gamma \ge 0$ are the regularization parameters. Via Eq. (5), the two parts can enhance with each other. On the one hand, the structured graph learning part can guide the data reconstruction process to provide robust representation. On the other hand, the data reconstruction part could learn a discriminative representation that improves the capability of the graph on revealing the intrinsic data structure.

To ensure that the learned graph can be directly used for clustering, the graph is supposed to contain exactly c connected components. However, the graph obtained by solving Eq. (5) does not achieve this ideal state. In order to formulate a clustering objective based on this target, we start with the following theorem. If the similarity matrix W is nonnegative, then its Laplacian matrix has an important property as follows [12].

Theorem 1. *The multiplicity c of the eigenvalue 0 of the Laplacian matrix L_W (nonnegative) is equal to the number of connected components in the graph with the similarity matrix W.*

According to the above theorem, we can easily obtain that if $rank(L_W) = n - c$, the undirected graph corresponding to this similarity matrix W will contain c connected components. Thus, we add the constraint into Eq. (5) to properly guide the graph learning process, then the overall objective function of ROGC is reformulated as

$$\min_{S,B,W} \|X - BS\|_F^2 + 2\alpha Tr(SL_WS^T) + \beta \sum_{i=1}^{n} \|s_i\|_1 + \gamma \sum_{i,j=1}^{n} w_{ij}^2$$
$$s.t. \forall i, w_i^T \mathbf{1} = 1, 0 \le w_{ij} \le 1, rank(L_W) = n - c, \|b_i\|^2 \le c. \tag{6}$$

Via Eq. (6), each data point will be assigned with adaptive neighbors and the learned graph will be updated accordingly until it contains c connected components.

2.3. Alternate optimization

Eq. (6) is difficult to solve due to the challenging rank constraint. In this paper, we first transform the challenging optimization problem into an equivalent one that can be tackled more easily.

Denoting $\sigma_i(L_W)$ as the ith smallest eigenvalue of L_W, we know $\sigma_i(L_W) \ge 0$ since L_W is positive semi-definite. It can be verified that the constraint $rank(L_W) = n - c$ in Eq. (6) will be satisfied if $\sum_{i=1}^{c} \sigma_i(L_W) = 0$. According to *KyFan's* Theorem [13], we have

$$\sum_{i=1}^{c} \sigma_i(L_W) = \min_{F^TF=I} Tr(F^T L_W F) \tag{7}$$

where $F = [f_1, f_2, \dots, f_n] \in \Re^{n \times c}$ is an indicator matrix. Therefore, Eq. (6) is equivalent to the following equation

$$\min_{S,B,F,W} \|X - BS\|_F^2 + 2\alpha Tr(SL_WS^T) + 2\lambda Tr(F^T L_W F)$$
$$+ \beta \sum_{i=1}^{n} \|s_i\|_1 + \gamma \sum_{i,j=1}^{n} w_{ij}^2$$
$$s.t. \forall i, w_i^T \mathbf{1} = 1, 0 \le w_{ij} \le 1, F^TF = I, \|b_i\|^2 \le c. \tag{8}$$

When λ is large, the optimal solution to Eq. (8) will make the equation $\sum_{i=1}^{c} \sigma_i(L_W) = 0$ hold, and thus the rank constraint can be satisfied accordingly. As shown in Eq. (8), W is constructed based on B and S, while S also depends on B and W. A natural approach to solve this problem is to iteratively optimize the function by minimizing over one variable while keeping the others fixed.

Updating S: When W, F and B are fixed, the irrelevant items $Tr(F^T L_W F)$ and $\sum_{i,j=1}^{n} w_{ij}^2$ can be removed and Eq. (8) is transformed into

$$\min_S \|X - BS\|_F^2 + 2\alpha Tr(SL_WS^T) + \beta \sum_{i=1}^{n} \|s_i\|_1. \tag{9}$$

Note that Eq. (9) is independent between different i, thus we rewrite Eq. (9) in the vector form to optimize each s_i

$$\min_{s_i} \sum_{i=1}^{n} \|x_i - Bs_i\|^2 + 2\alpha \sum_{i,j=1}^{n} L_{ij} s_i^T s_j + \beta \sum_{i=1}^{n} \|s_i\|_1. \tag{10}$$

When updating each s_i, the other vectors $\{s_j\}_{j \ne i}$ are fixed. Therefore, we get the following optimization equation

$$\min_{s_i} f(s_i) = \|x_i - Bs_i\|^2 + 2\alpha L_{ii} s_i^T s_i + s_i^T h_i + \beta \sum_{j=1}^{m} |s_i^{(j)}| \tag{11}$$

where $h_i = 4\alpha(\sum_{j \ne i} L_{ij} s_j)$ and $s_i^{(j)}$ is the jth coefficient of s_i. Eq. (11) can be solved by the algorithm proposed by Zheng et al. [14].

Updating B: When W, F and S are fixed, Eq. (8) is transformed into

$$\min_B \|X - BS\|_F^2 \quad s.t. \forall i, \|b_i\|^2 \le c. \tag{12}$$

Let $\theta = [\theta_1, \dots, \theta_m]$ and θ_i be the Lagrange multiplier associated with the ith inequality constraint $\|b_i\|^2 - c \le 0$, then the Lagrange dual function of Eq. (12) can be obtained by

$$g(\theta) = \inf_B L(B, \theta) = \inf_B \Big(\|X - BS\|_F^2 + \sum_{i=1}^{m} \theta_i (\|b_i\|^2 - c) \Big). \tag{13}$$

Let Λ be the $m \times m$ diagonal matrix whose diagonal entry $\Lambda_{ii} = \theta_i$ for all i. Then $L(B, \theta)$ can be written as

$$L(B, \theta) = \|X - BS\|_F^2 + Tr(B^T B \Lambda) - cTr(\Lambda)$$
$$= Tr(X^T X) - 2Tr(B^T X S^T) + Tr(S^T B^T B S)$$
$$+ Tr(B^T B \Lambda) - cTr(\Lambda). \tag{14}$$

The optimal solution B^* can be obtained by setting the first-order derivative of Eq. (14) equal to zero

$$B^* SS^T - XS^T + B^* \Lambda = 0. \tag{15}$$

Then we have

$$B^* = XS^T (SS^T + \Lambda)^{-1}. \tag{16}$$

Substituting Eq. (16) into Eq. (14), the Lagrange dual function becomes

$$
\begin{aligned}
g(\lambda) = {} & Tr(X^TX) - 2Tr\Big(XS^T(SS^T+\Lambda)^{-1}SX^T\Big) - cTr(\Lambda) \\
& + Tr\Big((SS^T+\Lambda)^{-1}SX^TXS^T\Big) = Tr(X^TX) - cTr(\Lambda) \\
& - Tr\Big(XS^T(SS^T+\Lambda)^{-1}SX^T\Big). \qquad (17)
\end{aligned}
$$

This leads to the following Lagrange dual function

$$\min_{\Lambda} Tr\Big(XS^T(SS^T+\Lambda)^{-1}SX^T\Big) + cTr(\Lambda)\ s.t.\theta_i \geq 0, i = 1, \ldots, m. \qquad (18)$$

Eq. (18) can be solved by conjugate gradient method [15]. Let Λ^* be the optimal solution, then $B^* = XS^T(SS^T+\Lambda^*)^{-1}$ is the optimal solution for B.

Updating F: When B, S and W are fixed, Eq. (8) is transformed into

$$\min_{F^TF=I} Tr(F^TL_WF). \qquad (19)$$

The optimal solution F of Eq. (19) is formed by the c eigenvectors of L_W corresponding to the c smallest eigenvalues.

Updating W: When B, S and F are fixed, the optimization formula for updating W becomes

$$
\begin{aligned}
& \min_{W} \sum_{i,j=1}^{n} \Big(\|s_i - s_j\|_2^2 w_{ij} + \frac{\gamma}{\alpha} w_{ij}^2 + \frac{\lambda}{\alpha}\|f_i - f_j\|_2^2 w_{ij}\Big) \\
& s.t.\forall i, w_i^T\mathbf{1} = 1, 0 \leq w_{ij} \leq 1. \qquad (20)
\end{aligned}
$$

We can update each vector w_i individually while holding all the other vectors constant. Thus Eq. (20) can be written as follows

$$
\begin{aligned}
& \min_{w_i} \sum_{j=1}^{n} (\|s_i - s_j\|_2^2 w_{ij} + \tfrac{\gamma}{\alpha} w_{ij}^2 + \tfrac{\lambda}{\alpha}\|f_i - f_j\|_2^2 w_{ij}) \\
& s.t.\forall i, w_i^T\mathbf{1} \ = 1, 0 \leq w_{ij} \leq 1.
\end{aligned} \qquad (21)
$$

Denote $d_{ij}^s = \|s_i - s_j\|_2^2$, $d_{ij}^f = \|f_i - f_j\|_2^2$, and $d_i \in R^{n\times 1}$ as a vector with the jth element as $d_{ij} = d_{ij}^s + \frac{\lambda}{\alpha} d_{ij}^f$, then Eq. (21) can be transformed as

$$\min_{w_i^T\mathbf{1}=1, 0\leq w_i\leq 1} \left\| w_i + \tfrac{\alpha}{2\lambda} d_i \right\|_2^2 \qquad (22)$$

The solution of Eq. (22) can be obtained by existing quadratic programming solvers. The key procedures of ROGC are summarized in Algorithm 1.

Algorithm 1 Robust optimal graph clustering.

```
Input: Data matrix $X \in R^{d\times n}$,the dictionary matrix $B$,parameters $\alpha$,
  $\beta$, $\gamma$,the number of basis vectors $m$ and the number of clusters
  $c$.
Output: Optimal similarity matrix $W$ with $c$ connected compo-
  nents.
  Initialize $W$ by the optimal solution to Eq. (4).
  repeat
     Learn $S$ by solving Eq. (1).
     Update $B$ by solving Eq. (18).
     Update $F$ by solving Eq. (19).
     Update $W$ by solving Eq. (22).
  until converge
```

3. Theoretical analysis

3.1. Computational complexity analysis

In this subsection, we analyze the computational complexity of the proposed ROGC. As shown in Algorithm 1, the solution of the proposed method can be divided into several iterative optimization steps. In each iteration, S is learned by Eq. (11), it is easy to find that its computational complexity is $O(m)$. Each s_i needs to be calculated n times, so the complexity of Eq. (11) is $O(n \times m)$. B is updated by solving Eq. (18), the computational complexity of this process is $O(m^3)$, where m is the size of B. F is obtained by solving Eq. (19), its cost is $O(n^3)$. W is updated by Eq. (22), we need $O(n)$ to compute d_i. In addition, each w_i needs to be calculated n times, so the complexity of Eq. (22) is $O(n^2)$. ROGC requires multiple iterations to obtain the optimal solution. Therefore, the total computational complexity is $O(T \times n^3)$, where T is the number of iterations. The computation complexity of ROGC is comparable to other state-of-the-art graph-based clustering methods [5,6,16].

3.2. Convergence analysis

In this subsection, we theoretically analyse the convergence of the proposed alternate optimization of Eq. (6) in Algorithm 1. The convergence can be proved according to the following theorem.

Theorem 2. *The iterative optimization process for solving Eq. (6) will monotonically decrease the objective function value until convergence.*

Proof. By fixing other variables and updating S, we adopt an optimization method based upon coordinate descent to solve this problem. It is easy to see that Eq. (9) is convex [17], thus, the global minimum can be achieved. We use $S^{(t)}$ and $S^{(t+1)}$ to denote the updated S in two adjacent iterations. According to the optimization of S, we know that $S^{(t+1)}$ makes the objective of Eq. (9) have smaller value than $S^{(t)}$. Then, we arrive at

$$\mathbf{\Omega}(S^{(t)}, B, F, W) \geq \mathbf{\Omega}(S^{(t+1)}, B, F, W). \qquad (23)$$

By fixing other variables and updating B, the problem becomes a least squares problem with quadratic constraints [14]. It is straightforward to check that Eq. (18) is a convex function [18]. Therefore, we can obtain that

$$\mathbf{\Omega}(S, B^{(t)}, F, W) \geq \mathbf{\Omega}(S, B^{(t+1)}, F, W). \qquad (24)$$

By fixing other variables and updating F, the objective function in Eq. (19) is convex (the Hessian matrix of the Lagrangian function of Eq. (19) is positive semi-definite [12]). Therefore, we can obtain that

$$\mathbf{\Omega}(S, B, F^{(t)}, W) \geq \mathbf{\Omega}(S, B, F^{(t+1)}, W). \qquad (25)$$

By fixing other variables and updating W, optimizing Eq. (22) is a typical quadratic programming problem. The Hessian matrix of the Lagrangian function of Eq. (22) is positive semi-definite. Therefore, we can obtain that

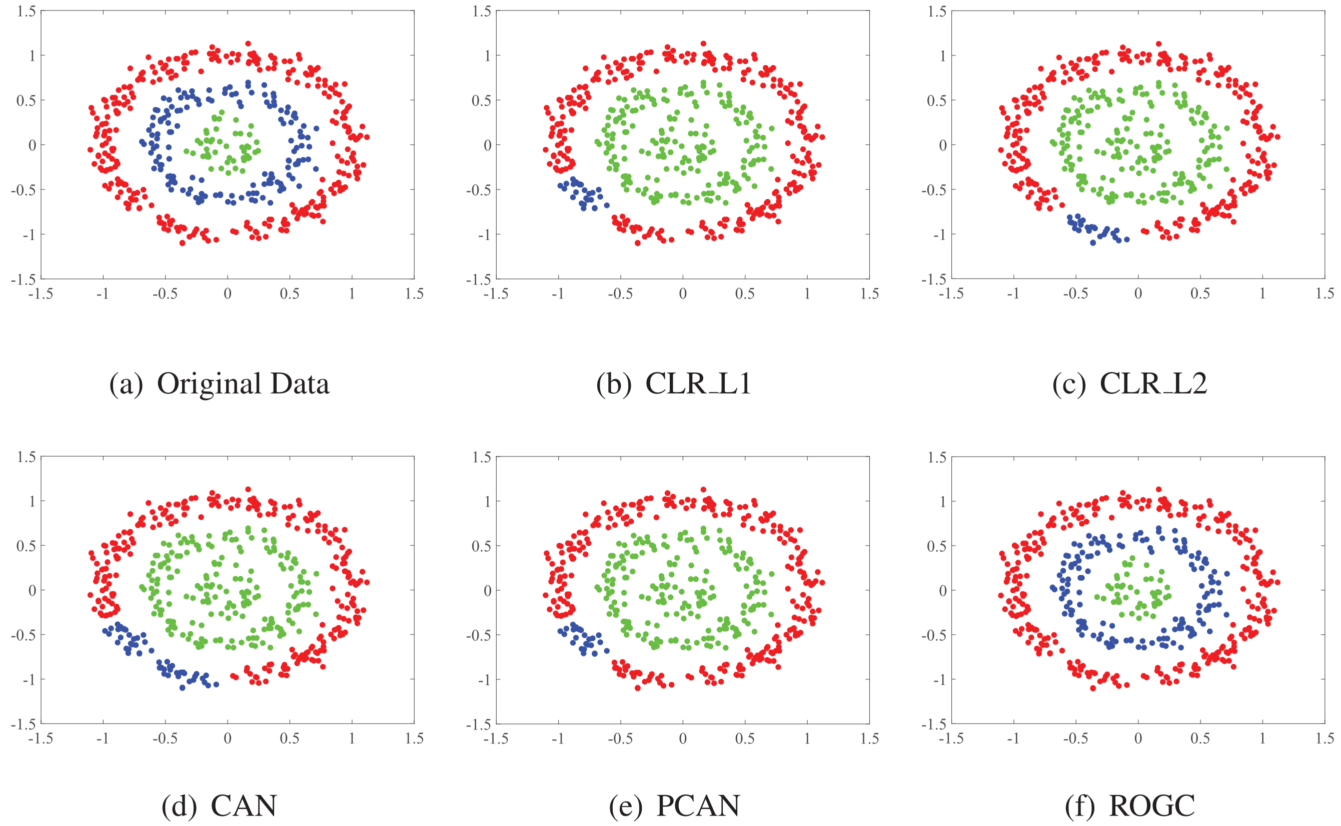
$$\mathbf{\Omega}(S, B, F, W^{(t)}) \geq \mathbf{\Omega}(S, B, F, W^{(t+1)}). \qquad (26)$$

As shown in the above analysis, the iterative optimization in Algorithm 1 can monotonically decrease the objective function of Eq. (6) in each iteration. Therefore, after several iterations, the convergence of Algorithm 1 can be achieved. □

4. Experimental configuration

4.1. Experimental datasets

We evaluate the proposed clustering method on 13 real datasets: Glass, Control, Yeast, Waveform, USPS, Palm, Ecoli, Dermatology, Solar, MSRA, COIL20, Yale and ORL. USPS is an image



(a) Original Data

(b) CLR_L1

(c) CLR_L2

(d) CAN

(e) PCAN

(f) ROGC

Fig. 1. Clustering results on three-ring dataset. The colors of data points indicate their cluster.

Table 2
Descriptions of 13 real datasets.

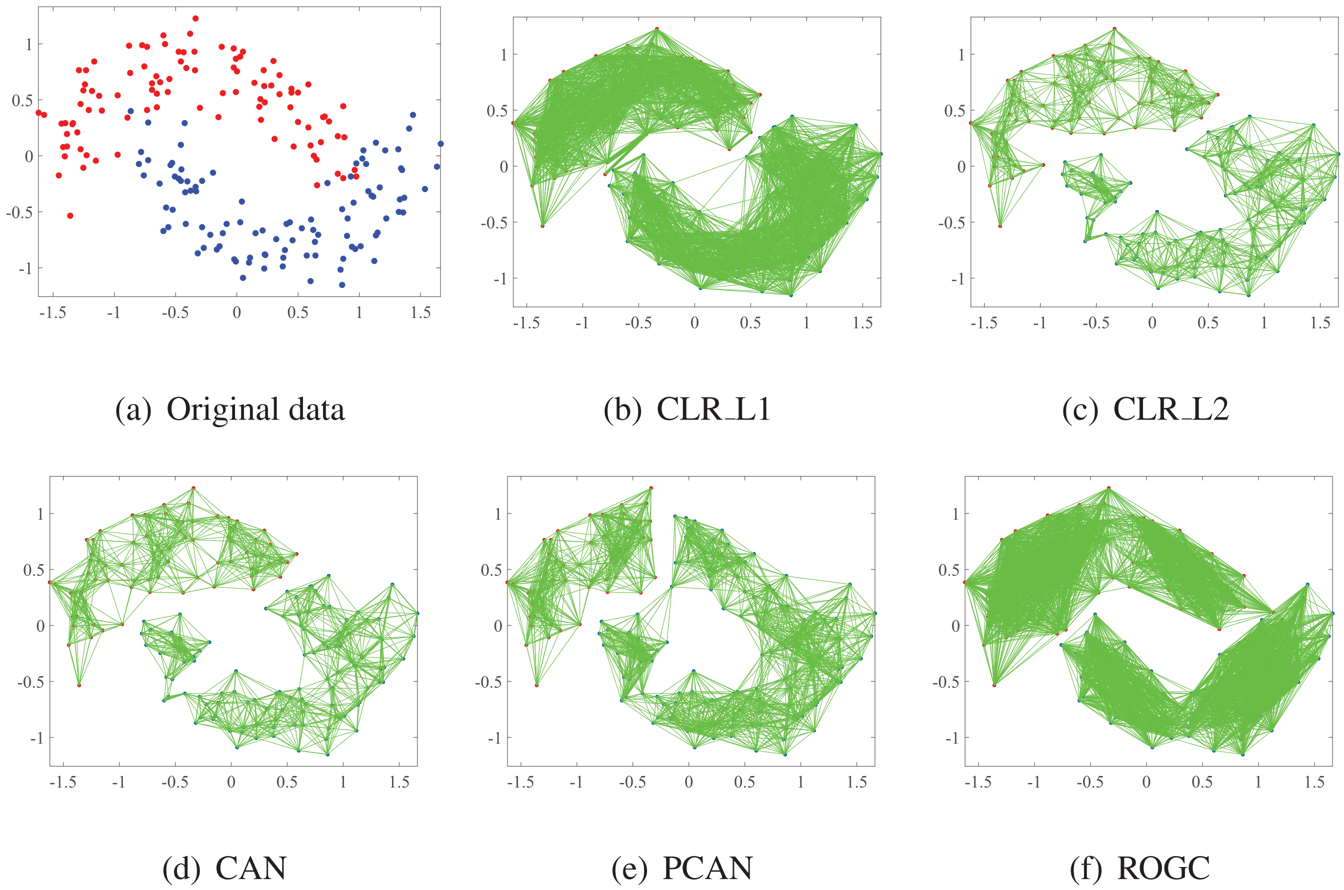
Datasets	Num of instances	Dimensions	Classes
Glass	214	9	6
Control	600	60	6
Yeast	1484	1470	10
Waveform (Wave)	2746	21	3
USPS	1854	256	10
Palm	2000	256	100
Ecoli	336	343	8
Dermatology (Derm)	366	34	6
Solar	323	12	6
MSRA	1799	256	12
COIL20	1440	1024	20
Yale	165	4096	3
ORL	400	4096	40

dataset [19], Palm is a digit dataset [20], COIL20 is an object dataset [21]. MSRA, Yale and ORL are face image datasets [22] and the other seven are biological datasets from UCI Machine Learning Repository [23]. The descriptions of these 13 datasets are summarized in Table 2.

4.2. Evaluation baselines

To demonstrate the performance of our method, we compare ROGC with 10 state-of-the-art clustering methods on the 13 real datasets mentioned above. The compared approaches are briefly introduced below:

- *K-means (KM)* [24]: *K*-means iteratively assigns each point into the cluster with the closest center based on certain similarity measurement (e.g., the Euclidean Distance) and then updates the center of each cluster.
- *Ratio Cut (RCut)* [1], *Normalized Cut (NCut)* [3]: RCut and NCut are two representative graph clustering methods. RCut cuts the graph by minimizing sub-graph cutting function and maximizing the number of data points lying in the sub-graph. NCut forces the sub-graph cutting function to be minimized and the weight of sub-graph to be maximized.
- *Non-negative Matrix Factorization (NMF)* [9]: NMF finds two nonnegative matrices whose product approximates the input one. *K*-means is performed on the reduced matrix to obtain the clustering results.
- *Constrained Laplacian Rank (CLR)* [6]: CLR learns a new data similarity matrix which has c connected components such that the clustering can be immediately obtained. Considering both L1 norm and L2 norm, this approach develops two variants of clustering objectives. To facilitate presentation, we abbreviate them as CLR_L1, and CLR_L2, respectively.
- *Clustering with Adaptive Neighbors (CAN)* [5]: CAN learns data similarity and the clustering structure simultaneously. The similarity matrix is learned by adaptive local structure learning. Rank constraint is imposed on the Laplacian matrix of the data similarity matrix to obtain the ideal clustering structure.
- *Projected Clustering with Adaptive Neighbors (PCAN)* [5]: PCAN is extended from CAN to handle high-dimensional data. It aims to find an optimal subspace on which the adaptive graph learning is performed.
- *Orthogonal and Nonnegative Graph Reconstruction (ONGR)* [10]: ONGR is proposed from the viewpoint of graph reconstruction. With orthogonal and nonnegative constraints, the reconstruction graph naturally has clear structure about the clusters and the post-processing is no longer needed.
- *Learning with Adaptive Neighbors for Image Clustering (LAN)* [11]: LAN tries to learn a new block diagonal data similarity matrix based on the given data graph so that the new graph is more suitable for the final clustering task.



(a) Original data

(b) CLR_L1

(c) CLR_L2

(d) CAN

(e) PCAN

(f) ROGC

Fig. 2. Clustering results on two-moon dataset. The colors of data points indicate their cluster and the width of the connecting line denotes the learned similarity of two corresponding points.

4.3. Evaluation metrics

The standard evaluation metrics Accuracy (ACC) [25], Normalized Mutual Information (NMI) [26] and Purity [16] are adopted to measure the clustering performance in our experiments.

- *ACC:* ACC is defined as follows

$$ACC(\Omega, G) = \frac{1}{n}\sum_{i=1}^{n}\sigma(s_i, t_i) \tag{27}$$

where $\Omega = [\omega_1, \dots, \omega_n]$ are the clusters. $G = [g_1, \dots, g_n]$ represent the ground truth classes. s_i indicates the cluster label of sample i, t_i is the ground truth label. $\sigma_i(s_i, t_i) = 1$ if $s_i = t_i$, otherwise $\sigma_i(s_i, t_i) = 0$.

- *NMI:* NMI is calculated by

$$NMI = \frac{MI(C, C^{'})}{\max(H(C), H(C^{'}))} \tag{28}$$

where C is a set of clusters obtained from the true labels, $C^{'}$ is a set of clusters obtained from the clustering algorithm. $H(C)$ and $H(C^{'})$ are the entropies of C and $C^{'}$, respectively, and $MI(C, C^{'})$ is the mutual information metric.

- *Purity:* Purity is defined as follows

$$Purity = \sum_{i=1}^{c}\frac{n_i}{n}P_{ij} \tag{29}$$

where c is the number of clusters and n is the number of members involved in the entire cluster partition. P_{ij} denotes the probability that a member of cluster i belongs to class j, $p_{ij} = \frac{n_{ij}}{n_i}$, where n_i is the number of all members in cluster i, and n_{ij} is the number of members in cluster i belonging to class j. In general, ACC, NMI and Purity are between 0 and 1, a larger value indicates better performance.

Table 3
Clustering ACC, NMI, Purity of five approaches on two-moon and three-ring datasets (%).

Methods	Three-ring dataset			Two-moon dataset		
	ACC	NMI	Purity	ACC	NMI	Purity
CLR_L1	84.80	74.95	90.00	89.00	51.75	89.00
CLR_L2	84.60	74.95	90.00	89.00	53.35	89.00
CAN	79.40	70.64	90.00	89.00	53.35	89.00
PCAN	84.80	74.95	90.00	76.00	27.51	76.00
ROGC	**99.80**	**98.51**	**99.80**	**94.00**	**67.36**	**94.00**

4.4. Implementation details

In experiments, we set the number of clusters to be the ground truth in each dataset. For those methods calling for an input of a similarity matrix, such as RCut, NCut, CLR_L1 and CLR_L2, the graph is constructed with the adaptive neighbors method [6]. For all the methods involving K-means, including K-means, RCut, NCut and NMF, since their performance is unstable with different initializations, we report their respective average result of 100 repetitions. The parameters in all compared methods are carefully adjusted to report their best performance. In the proposed method, the value of the dictionary size m is tested from 1 to 30. The regularization parameter α is tested in {0.001, 0.01, 0.1, 0, 1, 10, 100},

Table 4
Clustering ACC of compared approaches on 13 real datasets (%).

Methods	KM	Rcut	Ncut	NMF	CLR_L1	CLR_L2	CAN	PCAN	ONGR	LAN	ROGC
Glass	51.57	49.58	50.00	46.73	51.40	50.94	56.08	53.27	57.01	55.61	**57.94**
Control	60.81	62.72	63.30	60.04	60.83	60.83	62.50	71.50	88.00	68.67	**93.67**
Yeast	33.33	35.16	34.71	31.27	39.08	46.02	29.38	39.49	40.43	42.12	**49.26**
Wave	50.71	52.40	52.42	34.85	58.01	58.08	59.00	55.54	60.60	51.13	**70.28**
USPS	62.77	71.36	72.63	16.26	76.75	72.38	74.16	67.15	80.15	66.02	**83.60**
Palm	68.99	72.24	72.63	70.24	90.20	89.10	86.85	87.77	90.00	83.35	**98.55**
Ecoli	63.37	55.87	56.45	52.01	72.32	64.88	79.46	80.66	69.94	65.18	**86.31**
Derm	78.19	91.19	90.39	30.60	95.90	95.36	95.63	95.90	95.90	93.17	**97.27**
Solar	50.85	47.58	41.79	52.62	47.06	45.02	54.49	44.58	45.51	45.51	**56.35**
MSRA	48.86	55.49	56.28	45.28	57.37	59.14	57.75	59.26	61.81	54.75	**62.70**
COIL20	55.04	77.36	77.19	64.87	87.50	89.38	90.14	86.18	83.68	73.13	**98.75**
Yale	46.08	54.55	57.10	51.90	60.00	59.39	52.73	56.36	64.24	61.83	**66.06**
ORL	51.12	62.64	63.62	49.34	59.50	63.75	56.00	61.25	69.50	62.00	**71.75**

Table 5
Clustering NMI of compared approaches on 13 real datasets (%).

Methods	KM	Rcut	Ncut	NMF	CLR_L1	CLR_L2	CAN	PCAN	ONGR	LAN	ROGC
Glass	36.14	35.20	35.11	26.10	41.75	39.98	34.27	37.27	36.96	41.32	**42.09**
Control	67.26	74.58	74.59	67.71	74.21	74.21	72.70	78.32	84.17	78.39	**89.64**
Yeast	16.94	16.65	20.56	0.57	20.29	23.65	17.27	15.94	19.58	21.17	**27.03**
Wave	35.75	36.47	36.47	0.07	37.25	37.39	38.07	21.48	38.62	19.94	**44.26**
USPS	61.74	77.19	75.67	0.39	79.11	76.87	76.53	72.73	78.04	65.02	**81.04**
Palm	89.10	90.89	90.89	88.60	97.03	96.55	95.47	95.01	96.20	90.43	**99.30**
Ecoli	43.09	41.51	41.62	43.96	62.42	48.96	66.91	66.12	46.97	49.93	**71.14**
Derm	85.45	87.71	87.09	1.20	92.00	92.19	92.10	93.53	91.53	84.12	**94.57**
Solar	35.28	34.72	23.82	42.13	38.29	35.80	38.70	38.96	36.00	36.00	**48.79**
MSRA	56.04	70.40	69.99	49.00	71.10	74.45	76.85	**79.68**	73.80	64.01	76.20
COIL20	70.74	90.02	87.92	73.62	94.50	94.50	95.29	91.09	92.44	77.97	**98.75**
Yale	52.34	58.07	58.95	54.68	59.86	59.21	64.05	53.49	63.86	58.23	**66.03**
ORL	71.89	80.13	80.77	69.84	76.48	80.29	75.11	76.86	82.86	72.90	**84.90**

Table 6
Clustering Purity of compared approaches on 13 real datasets (%).

Methods	KM	Rcut	Ncut	NMF	CLR_L1	CLR_L2	CAN	PCAN	ONGR	LAN	ROGC
Glass	57.35	61.90	61.79	51.64	63.08	62.62	64.02	55.61	64.49	65.42	**69.63**
Control	66.27	68.27	68.71	65.90	66.67	66.67	76.33	75.67	88.00	76.00	**97.67**
Yeast	44.84	42.62	49.27	31.74	43.26	48.11	39.02	41.04	44.61	47.91	**50.07**
Wave	53.33	52.40	52.42	34.89	58.01	58.08	58.99	55.54	60.60	51.13	**70.28**
USPS	70.13	79.46	79.24	16.72	79.29	77.99	77.35	76.11	84.03	73.41	**84.63**
Palm	74.39	76.84	77.12	74.84	92.20	91.10	89.35	95.15	91.30	86.35	**98.55**
Ecoli	67.76	69.30	69.50	76.97	79.76	70.54	82.14	81.85	72.62	75.30	**88.69**
Derm	87.41	91.81	91.40	31.42	95.90	95.36	95.63	95.90	95.90	93.17	**97.27**
Solar	57.81	57.71	50.71	61.72	58.82	56.04	60.37	57.59	56.35	56.35	**65.33**
MSRA	52.10	60.20	59.71	47.65	60.92	76.04	79.49	73.49	62.53	38.69	**84.71**
COIL20	59.13	81.73	80.67	66.03	90.00	90.00	91.74	86.81	86.87	74.86	**98.75**
Yale	48.39	56.40	58.17	52.99	60.00	59.39	**76.36**	56.36	64.85	62.42	66.06
ORL	56.10	67.27	67.81	53.40	67.50	71.25	64.25	69.50	72.75	67.75	**75.75**

the sparsity parameter β is tested in $\{0.001, 0.01, 0.1, 0, 1\}$. Besides, there are two parameters γ and λ. γ can be determined by the number of neighbors [5] and λ is set in a heuristic way. Specifically, we initialize $\lambda = \gamma$, if the connected components of W is less than c in each iteration, we multiply λ by 2, if it is greater than c, we divide λ by 2. We keep the best group of parameters for ROGC.

5. Experimental results

In this section, experimental results are divided into seven parts. In the first two parts, we will show the comparison results on both synthetic datasets and real datasets. The third part and the fourth part prove the effects of robust representation learning and optimal graph learning, respectively. The fifth part reports the results of parameter analysis and the sixth part empirically proves the convergence of the proposed alternate optimization method. The last part reports the running times of the algorithms that are being compared.

5.1. Comparison results on synthetic datasets

We adopt two synthetic datasets to measure the clustering performance of ROGC. On these two datasets, we compare our method with CLR_L1, CLR_L2, CAN and PCAN. The first toy dataset is a randomly generated three-ring data which contains three data clusters distributed in the ring shape. Through extensive experiments, we found that ROGC achieves better performance than the comparison methods on this dataset when the noise percentage is greater than 0.07. These experimental results will be presented in Section 5.4. In this subsection, we set the noise percentage of three-ring dataset as 0.07. Our goal is to recompute the data graph such that the learned graph has exactly three connected components and clustering results can be obtained from this graph directly. The comparison results are displayed in Fig. 1 and Table 3. The second toy dataset is a randomly generated two-moon data which contains two data clusters distributed in the moon shape. Each cluster has a volume of 100 samples, and the noise

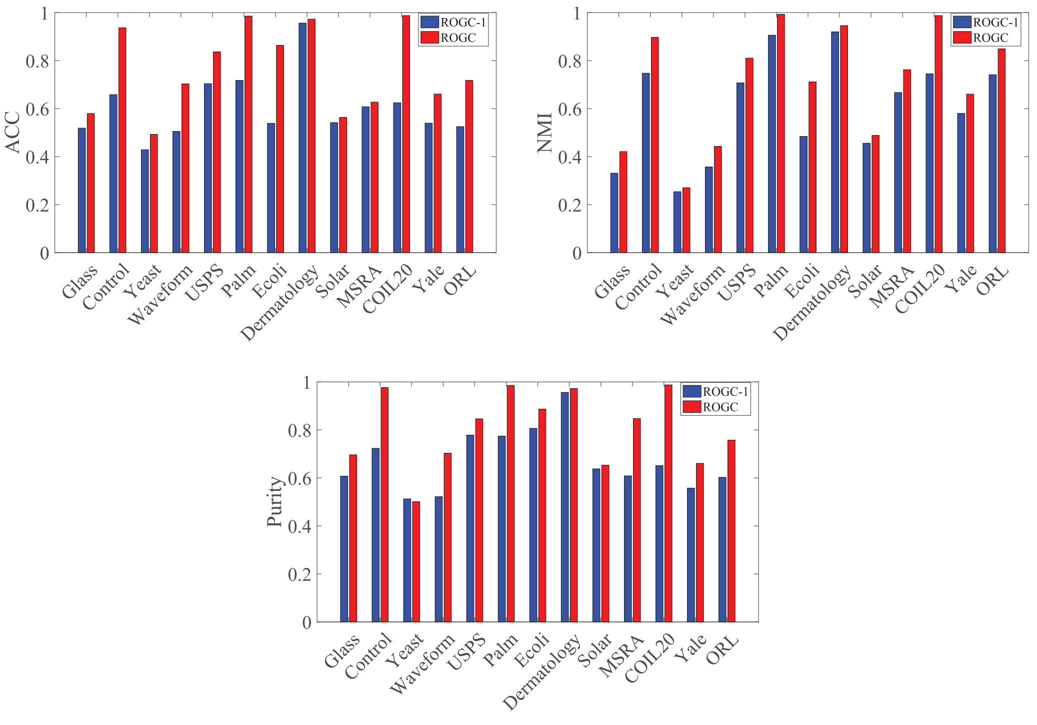


Fig. 3. Comparison results of ROGC and ROGC-1 on 13 real datasets. ROGC-1 is a variant of ROGC without optimal graph learning.

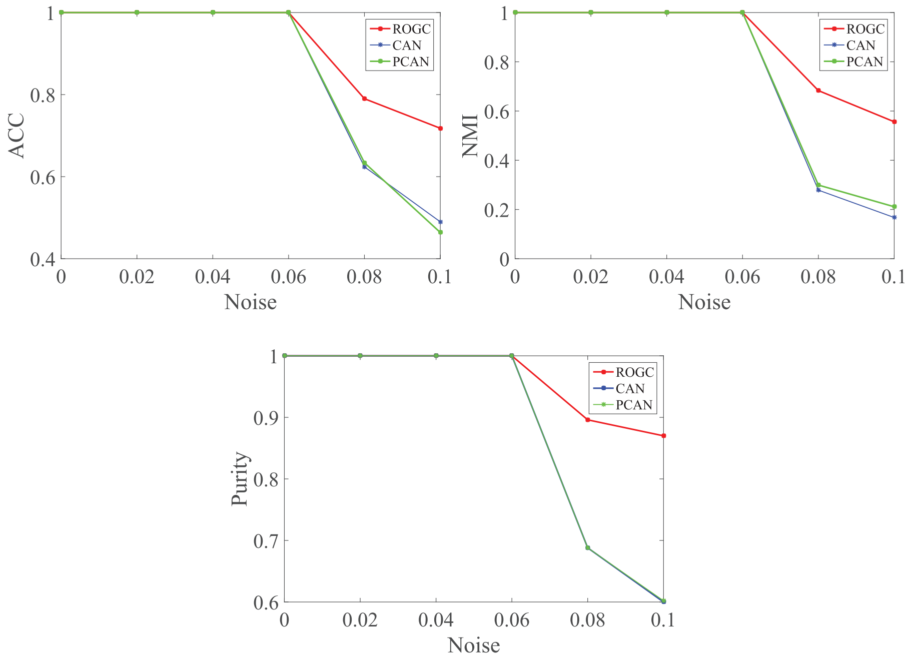


Fig. 4. Comparison results of CAN, PACN and ROGC on three-ring dataset with different noise.

percentage is set to be 0.18. The comparison results are displayed in Fig. 2 and Table 3. Figs. 1 and 2 show that more samples of the same category are clustered into the same component in ROGC. From Table 3, we also observe that ROGC achieves better performance under all evaluation metrics on two synthetic datasets.

5.2. Comparison results on real datasets

We compare ROGC with the 10 approaches mentioned above on 13 real datasets. As can be observed from Tables 4 to 6, ROGC consistently achieves better ACC, NMI and Purity under different circumstances. For example, ROGC gains 5.67%, 9.68%, 8.35%, 8.61% increment of ACC over the second best results on Control, Waveform, Palm and COIL20, 5.47%, 5.64%, 6.66% increment of NMI on Control, Waveform and Solar and 9.67%, 9.68%, 6.55%, 5.22% and 7.01% increment of Purity on Control, Waveform, Ecoli, MSRA and COIL20. These results clearly demonstrate the superior performance of ROGC over existing clustering methods. Our method reduces the impact of noises by robust representation learning which reconstructs the raw data with sparse representation. In addition, due to the rank constraint imposed on the Laplacian matrix of the similarity matrix, the learned graph has *c* connected components and becomes well structured.

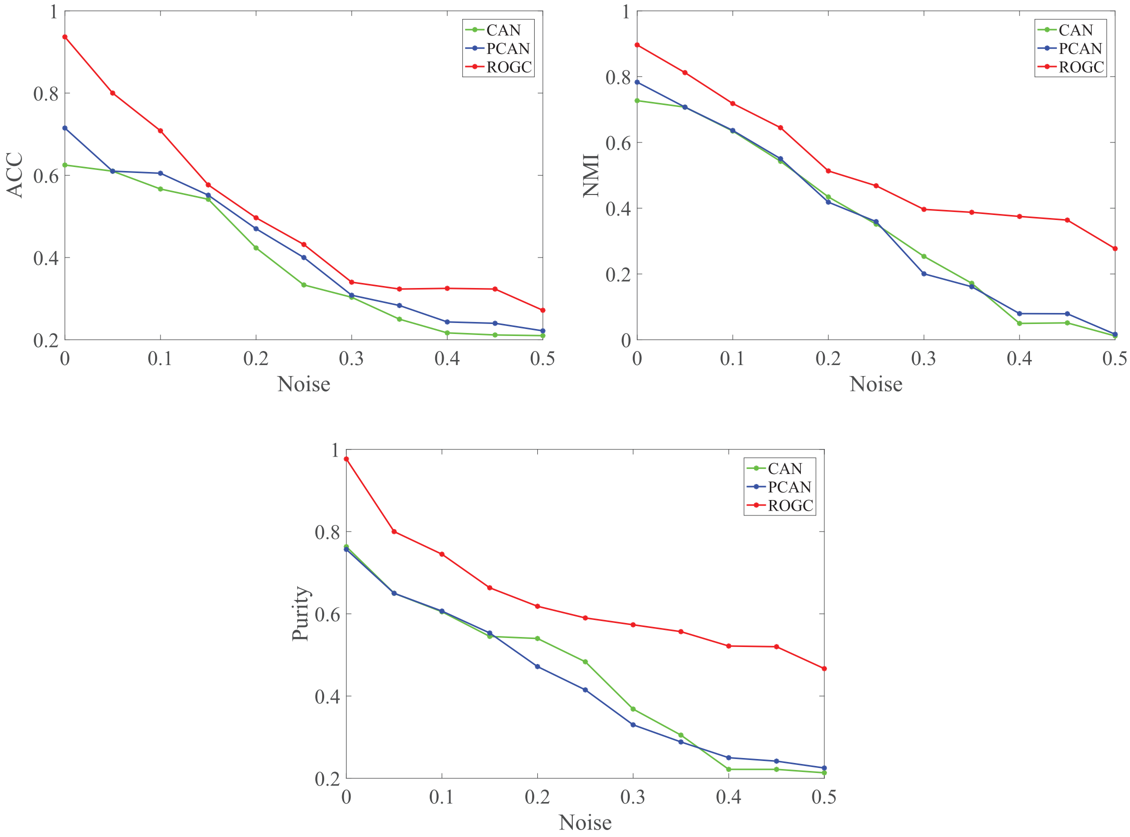
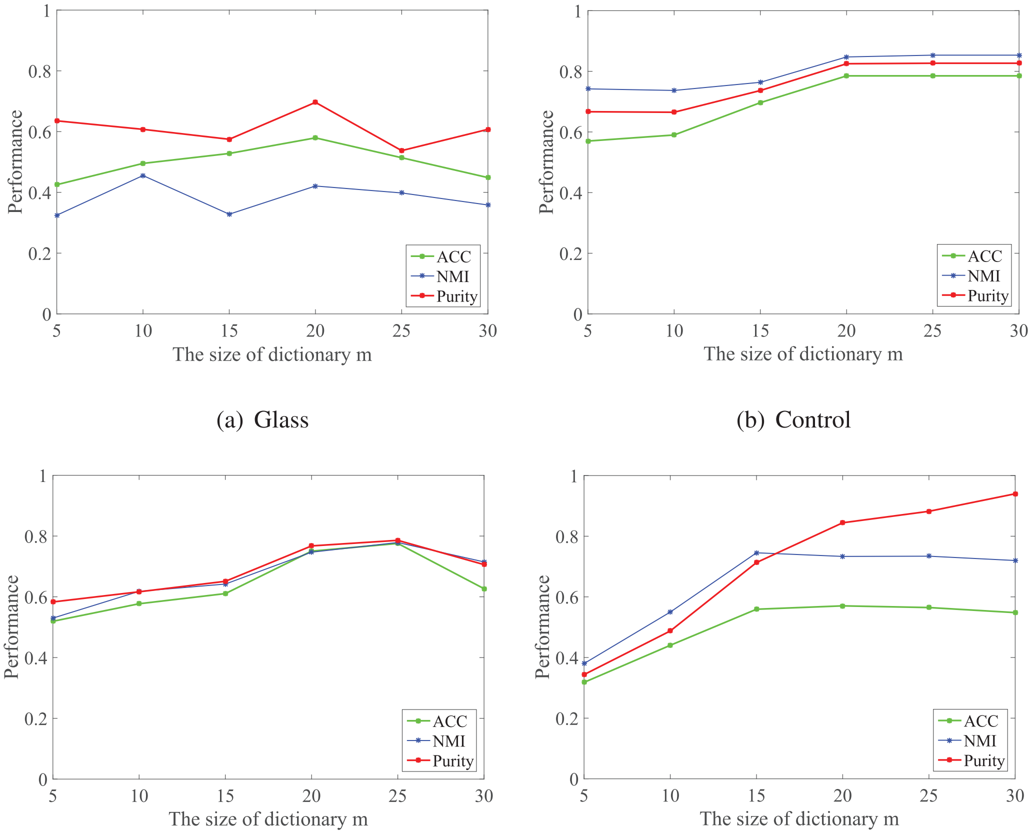


Fig. 5. Comparison results of CAN, PACN and ROGC on Control dataset with different noise.



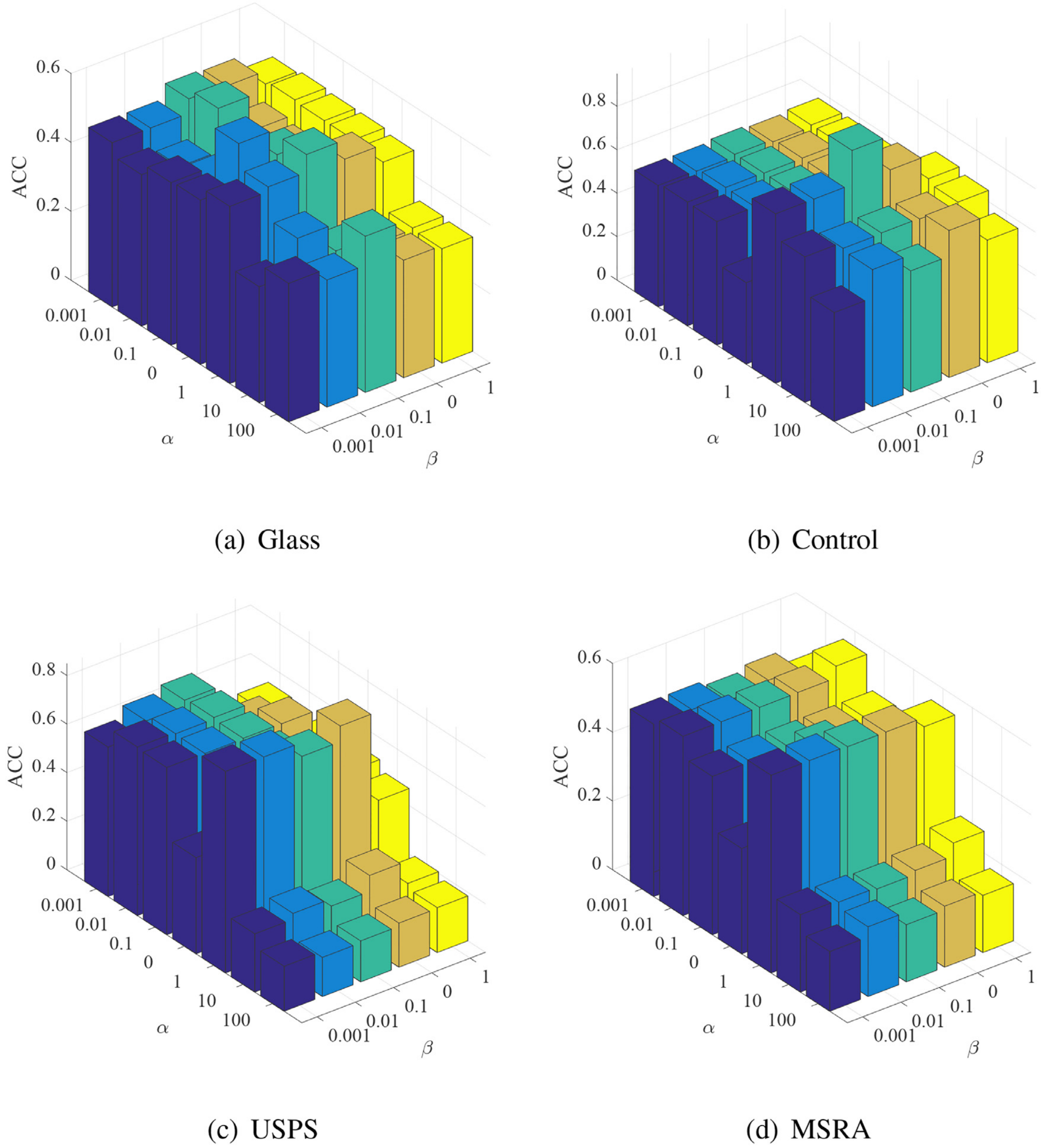
(a) Glass

(b) Control

(c) USPS

(d) MSRA

Fig. 6. Clustering performance variations with the size of dictionary m on Glass, Control, USPS and MSRA.



(a) Glass

(b) Control

(c) USPS

(d) MSRA

Fig. 7. Clustering ACC variations with different values of the regularization parameter α and sparsity parameters β on Glass,Control, USPS and MSRA.

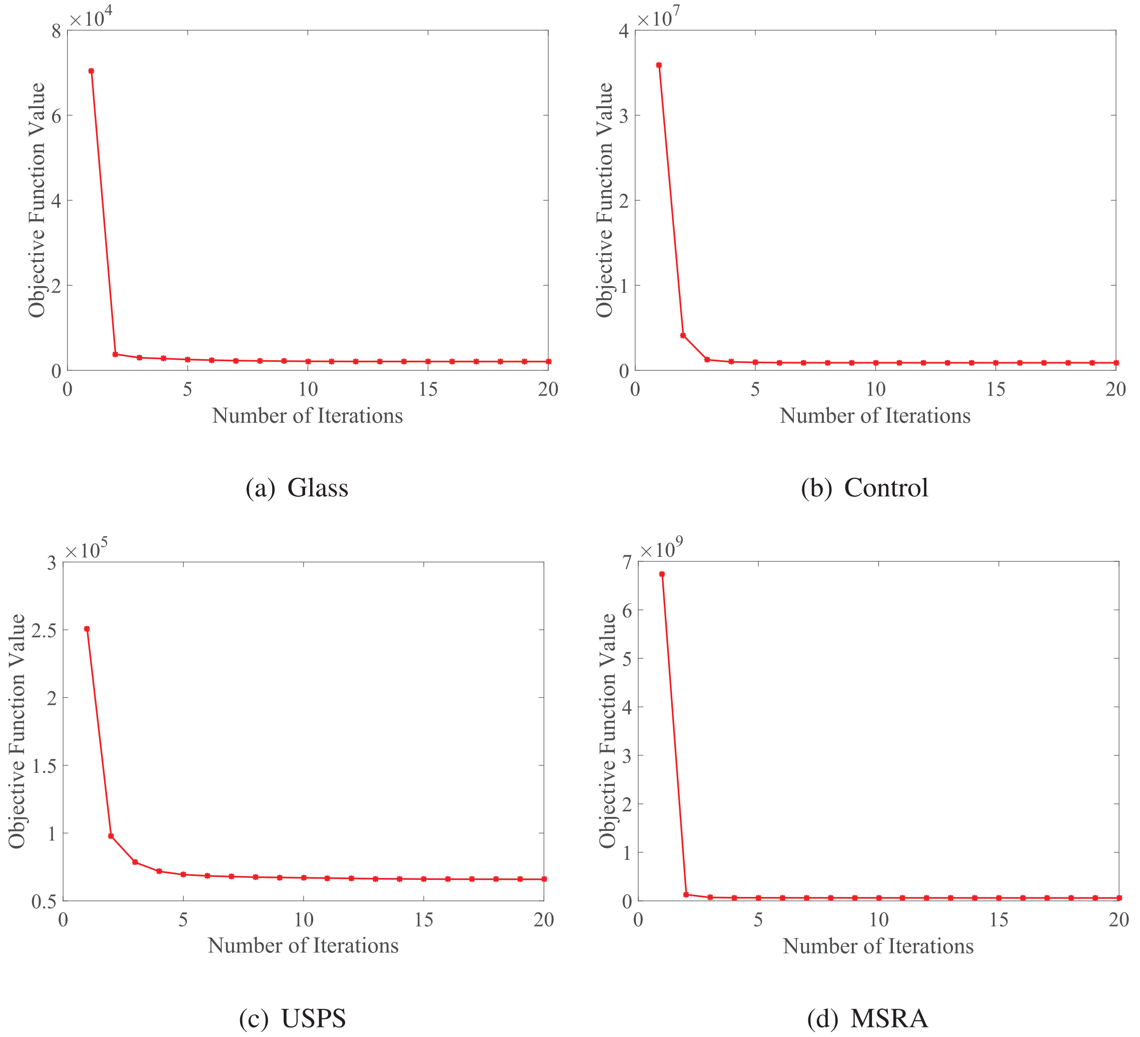
5.3. Effects of optimal graph learning

Our method learns a similarity graph from the learned sparse representation by assigning the adaptive neighbors for each data point. In this subsection, we conduct experiment to investigate the effects of optimal graph learning. Specifically, we first calculate a sparse representation of raw data by removing the part of optimal graph learning in Eq. (6). Then, the clustering results are obtained by performing the K-means on this sparse representation. We denote this variant as ROGC-1. Fig. 3 presents the comparison results between ROGC and ROGC-1. From it, we can observe that ROGC outperforms ROGC-1 under all three evaluation metrics only except the Purity on Yeast dataset. The potential reason for the improved performance is that the optimal graph learning preserves the intrinsic structure of raw data by learning similarity graph and the learned graph can be directly used for clustering by setting rank constraint.

5.4. Effects of robust representation learning

The key point of graph-based clustering performance is to construct a high-quality similarity graph. However, the similarity graph directly constructed from the raw features may be unreliable as real world data always involve adverse noises, outliers and irrelevant information. Our method reduces the impact of noises by robust representation learning which reconstructs the raw data with sparse representation. To evaluate the effects of robust representation learning, we design anti-noise experiments on synthetic datasets and real datasets, respectively. As a matter of fact, without data reconstruction, ROGC degenerates to CAN, so we compare our approach with CAN and PCAN.

For synthetic datasets, we take the three-ring dataset as an example. First, we add different ratios of Gaussian noise to this dataset. The noise percentage is set from 0 to 0.1 with an interval of 0.02, thus a set of the noisy datasets are constructed. We



(a) Glass

(b) Control

(c) USPS

(d) MSRA

Fig. 8. Variations of the objective function value in Eq. (6) with respect to the number of iterations on Glass, Control, USPS and MSRA.

compare the performance of CAN, PCAN and ROGC on these five datasets. Fig. 4 shows the comparison results.

For real datasets, we take Control dataset as an example. Suppose r is the ratio of random noise, and n is the number of original datasets, we randomly pick out $n \times r$ data points from the original datasets. Since the values of the internal elements of the data matrix are mostly between 30 and 50, we add a normal distribution with a mean of 40 and a standard deviation of 10 on the selected data. Thus, a set of the noisy datasets are formed with different r ranging from 0 to 0.5 with a step size of 0.05. We compare the clustering effects of CAN, PCAN and ROGC on these 11 datasets. Fig. 5 shows the comparison results. From the experimental results, we have the following observation. Although the performance of all the compared algorithms decreases as the noise ratio increases, the performance gain of the proposed method becomes more significant. This confirms the effect of robust representation learning.

5.5. Parameter analysis

There are five parameters m, α, β, γ, λ in our method. Previous methods [5,27] have proved that γ can be determined by the number of neighbors and thus we skip its analysis. The parameter λ can be obtained during the iteration as mentioned before. We conduct the experiments to observe the performance variations in terms of the different values of m on Glass, Control, USPS and MSRA. Detailed experimental results are presented in Fig. 6. Moreover, we investigate the influence of different choices of α and β. Fig. 7 shows their effects on the clustering performance (ACC), respectively. According to Fig. 7, we can see that the best performance is obtained when α is chosen from {0.001, 0.01, 0.1, 1} and β is chosen from {0.001, 0.01, 0.1, 0}, respectively.

5.6. Convergence analysis

In this subsection, we experimentally investigate the convergence of ROGC. Fig. 8 records the variations of the objective function value in Eq. (6) with respect to the number of iterations on Glass, Control, USPS and MSRA datasets. Similar results can be obtained on the other datasets. From Fig. 8, we can see that the objective function value decreases sharply at first and does not change significantly after about 10 iterations. The results show that our proposed approach is able to converge efficiently.

5.7. Running times comparison

In this subsection, we conduct experiments to compare the computational efficiency between ROGC and baselines. All our experiments are conducted on a computer with a 3.6 GHz Intel®Xeon(R) CPU E5-1650 v4. For all methods, we run them once and report their running times. Table 7 presents the comparison results on 13 real datasets. As can be observed, KM, Rcut, Ncut and NMF are significantly faster than the other methods. Besides,

Table 7
Running times (recorded in seconds) of compared approaches on 13 real datasets.

Methods	KM	Rcut	Ncut	NMF	CLR_L1	CLR_L2	CAN	PCAN	ONGR	LAN	ROGC
Glass	0.165	0.207	0.208	0.181	0.409	0.379	0.270	0.306	0.134	0.180	2.146
Control	0.166	0.261	0.237	0.186	0.948	0.729	1.406	0.816	0.157	0.562	2.271
Yeast	0.227	0.583	0.554	1.676	5.668	4.844	6.955	22.830	0.731	5.528	5.358
Wave	0.151	1.934	2.068	0.200	41.408	40.27	27.327	37.527	2.267	27.336	7.976
USPS	0.234	0.855	0.774	0.444	11.863	9.714	6.489	6.551	0.996	5.721	6.813
Palm	0.397	1.070	1.150	0.945	13.474	7.021	5.008	7.401	1.536	5.568	7.495
Ecoli	0.193	0.229	0.231	0.239	0.570	0.455	0.415	0.801	0.149	0.350	1.402
Derm	0.186	0.205	0.202	0.150	0.648	0.481	0.411	0.406	0.107	0.299	1.533
Solar	0.180	0.208	0.214	0.174	0.659	0.426	0.359	0.394	0.111	0.237	1.437
MSRA	0.215	0.792	0.688	0.652	0.742	0.722	13.756	23.834	0.919	4.487	2.840
COIL20	0.352	0.582	0.593	1.141	6.248	4.050	3.380	6.594	0.567	2.723	5.326
Yale	0.197	0.228	0.221	0.499	0.337	0.296	0.375	216.278	0.104	0.170	1.344
ORL	0.308	0.338	0.305	1.160	0.735	0.572	0.500	214.605	0.241	0.336	2.506

the running times of the CLR_L1, CLR_L2, CAN and LAN is significantly larger on Waveform and USPS datasets and it is similar when performing PCAN on Yale and ORL. These results indicate that their efficiency is limited when the number of sample points or the feature dimension is large. In contrast, the running times of ONGR and ROGC is relatively stable. Compared to ONGR, since our method first learns a discriminative representation of data samples via sparse reconstruction, it inevitably increases the running times of our method.

6. Conclusions

In this paper, we propose a robust optimal graph clustering model that performs robust representation learning and optimal graph learning simultaneously. Our method can reduce the impact of noise effectively and preserve the local structure of the data. A reasonable rank constraint is imposed on the Laplacian matrix of similarity matrix so that the learned graph has ideal structure and can well support the subsequent clustering. We derive an alternate algorithm to optimize the proposed challenging problem. Experimental results show that the proposed method achieves state-of-the-art clustering performance.

Declaration of Competing Interest

We declare that there is no conflict of interest with this submission.

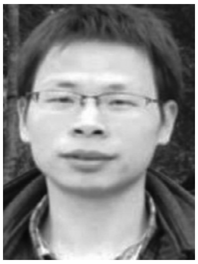
Acknowledgments

This work was supported National Natural Science Foundation of China under Grants 61802236, 61806039, 61602283, and U1836216, in part by the Natural Science Foundation of Shandong under Grant ZR2019QF002, in part by the Taishan Scholar Project of Shandong Province, China.

References

[1] L. Hagen, A.B. Kahng, New spectral methods for ratio cut partitioning and clustering, IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. 11 (9) (1992) 1074–1085.
[2] A.Y. Ng, M.I. Jordan, Y. Weiss, On spectral clustering: analysis and an algorithm, in: Proceedings of the 2002 NIPS, 2002, pp. 849–856.
[3] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 888–905.
[4] L. Zhang, Z. Chen, M. Zheng, X. He, Robust non-negative matrix factorization, Front. Electr. Electron. Eng. China 6 (2) (2011) 192–200.
[5] F. Nie, X. Wang, H. Huang, Clustering and projected clustering with adaptive neighbors, in: Proceedings of the 2014 ACM SIGKDD, 2014, pp. 977–986.
[6] F. Nie, X. Wang, J. Michael I, H. Huang, The constrained Laplacian rank algorithm for graph-based clustering, in: Proceedings of the 2016 AAAI, 2016, pp. 1969–1976.
[7] F. Nie, G. Cai, X. Li, Multi-view clustering and semi-supervised classification with adaptive neighbours, in: Proceedings of the 2017 AAAI, 2017, pp. 2408–2414.
[8] D. Cai, X. He, J. Han, Locally consistent concept factorization for document clustering, Trans. Knowl. Data Eng. 23 (6) (2011) 902–913.
[9] D. Cai, X. He, J. Han, T.S. Huang, Graph regularized nonnegative matrix factorization for data representation, IEEE Trans. Pattern Anal. Mach. Intell. 33 (8) (2010) 1548–1560.
[10] J. Han, K. Xiong, F. Nie, Orthogonal and nonnegative graph reconstruction for large scale clustering, in: Proceedings of the 2017 IJCAI, 2017, pp. 1809–1815.
[11] Y. Liu, Q. Gao, Z. Yang, S. Wang, Learning with adaptive neighbors for image clustering, in: Proceedings of the 2018 IJCAI, 2018, pp. 2483–2489.
[12] B. Mohar, Y. Alavi, G. Chartrand, O.R. Oellermann, The Laplacian spectrum of graphs, SIAM 11 (2) (1990) 218–238.
[13] F. Ky, On a theorem of Weyl concerning eigenvalues of linear transformations. I, Proc. Natl. Acad. Sci. 35 (11) (1949) 652–655.
[14] M. Zheng, J. Bu, C. Chen, C. Wang, L. Zhang, G. Qiu, D. Cai, Graph regularized sparse coding for image representation, IEEE Trans. Image Process. 20 (5) (2011) 1327–1336.
[15] T. Steihaug, The conjugate gradient method and trust regions in large scale optimization, SIAM 20 (3) (1983) 626–637.
[16] N. Zhao, L. Zhang, B. Du, Q. Zhang, J. You, D. Tao, Robust dual clustering with adaptive manifold regularization, IEEE Trans. Knowl. Data Eng. 29 (11) (2017) 2498–2509.
[17] X. Zhu, X. Li, S. Zhang, C. Ju, X. Wu, Robust joint graph sparse coding for unsupervised spectral feature selection, IEEE Trans. Neural Netw. Learn. Syst. 28 (6) (2016) 1263–1275.
[18] H. Lee, A. Battle, R. Raina, A.Y. Ng, Efficient sparse coding algorithms, in: Proceedings of the 2007 NIPS, 2007, pp. 801–808.
[19] J.J. Hull, A database for handwritten text recognition research, IEEE Trans. Pattern Anal. Mach. Intell. 16 (5) (1994) 550–554.
[20] P.N. Belhumeur, J.P. Hespanha, D.J. Kriegman, Recognition using class specific linear projection, IEEE Trans. Pattern Anal. Mach. Intell. 19 (7) (1997) 711–720.
[21] S.A. Nene, S.K. Nayar, H. Murase, Columbia Object Image Library (COIL-20), Technical Report, Columbia University, CUCS-005-96(1996).
[22] M.J. Lyons, J. Budynek, S. Akamatsu, Automatic classification of single facial images, IEEE Trans. Pattern Anal. Mach. Intell. 21 (12) (1999) 1357–1362.
[23] A. Frank, A. Asuncion, UCI machine learning repository, Natl. Acad. Sci. 35 (11) (2010) 652–655.
[24] M. James, Some methods for classification and analysis of multivariate observations, Berkeley Symp. Math. Stat. Prob. 1 (14) (1967) 281–297.
[25] W. Wang, Y. Yan, F. Nie, S. Yan, N. Sebe, Flexible manifold learning with optimal graph for image and video representation, IEEE Trans. Image Process. 27 (6) (2018) 2664–2675.
[26] F. Nie, D. Xu, I.W. Tsang, C. Zhang, Spectral embedded clustering, in: Proceedings of the 2009 IJCAI, 2009, pp. 1181–1186.
[27] F. Nie, W. Zhu, X. Li, Unsupervised feature selection with structured graph optimization, in: Proceedings of the 2016 AAAI, 2016, pp. 1302–1308.

Fei Wang is with the School of Information Science and Engineering, Shandong Normal University, China. Her research interest is in the area of big data mining.



Lei Zhu received the B.S. degree (2009) at WuHan University of Technology, the Ph.D. degree (2015) at Huazhong University of Science and Technology. He is currently a full Professor with the School of Information Science and Engineering, Shandong Normal University, China. He was a Research Fellow at the University of Queensland (2016–2017), and at the Singapore Management University (2015–2016). His research interests are in the area of large-scale multimedia content analysis and retrieval.

Cheng Liang received her Ph.D. from College of Computer Science and Electronic Engineering, Hunan University in 2015. She is currently an Assistant Professor in School of Information Science and Engineering, Shandong Normal University. She studied at Donnelly Centre, University of Toronto from 2012 to 2014 as a joint Ph.D. student. Her research interests include graph mining and bioinformatics.

Jingjing Li received his M.Sc. and Ph.D. degree in Computer Science from University of Electronic Science and Technology of China in 2013 and 2017, respectively. Now he is a national Postdoctoral Program for Innovative Talents research fellow with the School of Computer Science and Engineering, University of Electronic Science and Technology of China. He has great interest in machine learning, especially transfer learning, subspace learning and recommender systems.

Xiaojun Chang is a faculty member at Faculty of Information Technology, Monash University Clayton Campus, Australia. He is also affiliated with Monash University Centre for Data Science. He is an ARC Discovery Early Career Researcher Award (DECRA) Fellow between 2019–2021. Before joining Monash, he was a Postdoc Research Associate in School of Computer Science, Carnegie Mellon University, working with Prof. Alex Hauptmann. He has spent most of time working on exploring multiple signals (visual, acoustic, textual) for automatic content analysis in unconstrained or surveillance videos. He has achieved top performance in various international competitions, such as TRECVID MED, TRECVID SIN, and TRECVID AVS.

Ke Lu received the B.S. degree in thermal power engineering from Chongqing University, Chongqing, China, in 1996 and the M.Sc. and Ph.D. degrees in computer application technology from the University of Electronic Science and Technology of China, Chengdu, China, in 2003 and 2006, respectively. He is currently a Professor with the School of Computer Science and Engineering, University of Electronic Science and Technology of China. His research interests include pattern recognition, multimedia, and computer vision.