Community detection method based on mixed-norm sparse subspace clustering

Community Detection Method Based on Mixed-norm Sparse Subspace Clustering

Bo Tian [1] *, Weizi Li [2]

[1] School of Information Management & Engineering, Shanghai University of Finance and Economics, Shanghai, 200433;
E-Mail: youngtb@sina.com
[2] Informatics Research Centre, University of Reading, Reading RG6 6UD, UK;
E-Mail: weizi.li@henley.ac.uk

Abstract: Community or group is an important structure in disciplines such as social networks, biology gene expression, and physics systems. Community detections for different types of networks have attracted considerable interest. However, it is still challenging to find meaningful community structures in various networks. In particular, accurate community description and implementation of effective detection algorithms with huge datasets are still not solved. In this paper, we present a novel community detection algorithm based on the theory of sparse subspace clustering (SSC) with mixed-norm constraints. Inspired by the sparse representation of subspace, each community in a given network can span a subspace in some similarity measure space. If the basis of subspaces can be solved, all of the nodes can be represented as a linear combination of the nodes that span the same subspace. By introducing a novel mixed-norm constraint in SCC, the connections of nodes among different communities are modeled as noise to improve the clustering accuracy. The formulation of the basis of subspaces is derived from the self-representation property of data by using SSC. Then, the alternating directions method of multipliers (ADMM) framework is used to solve the formulation. Finally, communities are detected by subspace clustering method. The proposed method is compared with state-of-the-art algorithms on synthetic networks and real-world networks. The experimental results show the effectiveness of the proposed algorithm in accurately describing the community. The results also show that the mixed-norm SSC is a practical approach for detecting communities in huge datasets.

Keywords: Community detection, sparse subspace clustering, sparse representation, mixed-norm, similarity measure

1 Introduction

Networks and graph theory have made significant contributions to the understanding of complex systems such as human social networks. A complex system can be represented as a network or graph with a set of vertices joined in pairs by edges. The Pareto principle (a.k.a. the 80–20 rule) [1] exists almost everywhere. It has been found that different types of networks are usually inhomogeneous, that is, they do not consist of undifferentiated vertices but rather of distinct groups. In other words, typically, not all the vertices or autos play the same roles. Within the groups, many edges exist between vertices but fewer exist between groups. Community detection in large and complex networks is essential to understanding the structure of real networks because intrinsic community structures influence the dynamics of many real-world networks [2-4]. For example, an Internet community could consist of several websites that address related topics. Communities in biochemical networks or electronic circuits refer to functional units. Social networks such can be represented as graphs in which a node represents a user and an edge represents the user's affiliation with another user. These affiliations can represent friendships, followings, or information flows. Nodes with similar affiliations tend to group into dense communities, which further formulate the network structures. Three characteristics of social network structures have been found: the small world phenomenon, the power structure, and the community structure [5].

Community detection is an important task in different research fields such as social networks, computer science, biology gene expression, and physics systems. The rapid growth of big data theory over the past decade has caused community detection to become prominent in terms of both theory and practice. However, detecting meaningful, inherent and hidden community structures in networks remains a challenge work. Although cluster theory was initially adopted to address the community detection problem, no unified quantitative definition has thus far emerged for the conception of communities [2]. Because network structures can usually easily be represented as graphs, community detection can also be formulated as a graph partition problem in the fields of graph theory [2, 5]. Communities are usually considered as groups of nodes in which intra-group connections are much denser than inter-group connections. Therefore, the problem of community detection can be simply defined as clustering the

vertices of a given underlying graph of the network into groups based on certain predefined similarity measures. Many solutions have emerged in the literature to solve this problem from various disciplines and perspectives. Community structure analysis was first proposed by Weiss and Jacobson in 1955 [6]. In 2002, Girvan and Newman proposed a new iterative algorithm to identify the edges between different communities [3]. Then, Newman separated the methods of community detection into two categories: sociology approaches and top-down computer science approaches [7]. The author observed that both sociological and computer science approaches performed well, but most algorithms are computationally expensive. In addition, their scalability is limited within few thousands nodes.

Community detection can reveal the relationships of nodes in a network such as common interests between people or similar structures between genes. Community detection is intuitively easy at first glance but proves to be an intricate problem [8]. In undirected and un-weighted networks, the adjacency matrix of the corresponding graph is determined by the nodes of networks using one-to-one mapping. Most existing community detection methods operate on the adjacency matrix and use modularity maximization optimization techniques [4]. If there are N nodes in a network, the dimensionality of the adjacency matrix is N*N. The dimensionality of the eigenvectors of the adjacency matrix is N. All the nodes in the network are presented in N-dimensional feature spaces from the original adjacency matrix. Most similarity measures are ineffective in high-dimensional feature spaces because the differences between the nearest and farthest neighbor of one data point becomes negligible[9]. High dimensionality not only increases computational time but also adversely affects performance, which is commonly referred to as the "curse of dimensionality"[10-11]. The nodes in a community are clearly not uniformly distributed in the ambient space. Nodes in networks with high-dimensionality must lie in some low-dimensional subspaces determined by nodes in the same community structures. The actual dimensions of nodes could be significantly smaller than they seem to be if there are several communities. Low-dimensional representation of the data on one hand could reduce computation cost; on the other hand, it could reduce the impact of high-dimension while improve recognition performance. If the nodes can be represented by other nodes in the same community, the dimensions of the nodes are then dependent upon the size of the same community spanned by those nodes.

Based on the sparse representation theory, we proposed a novel community detection method by using the mixed-norm sparse subspace clustering (SSC). The contributions of this paper are as follows: 1) the problem of community detection is formulated as detecting low-dimensional subspace from original dataset space based on SSC; 2) by using the alternating directions method of multipliers (ADMM) framework, each community is solved in the similarity measure space; 3) by introducing a novel mixed-norm constraint in SCC, the connections of nodes between different communities are modeled as noises to improve the clustering accuracy. The remainder of this paper is organized as follows. State-of-the-art studies on community detection and sparse subspace clustering are reviewed in Section 2. The proposed method is described into three parts in Section 3:1) similarity measures for network nodes derived from the adjective matrix; 2) vertices sparsely represented using the sparse representation theory; 3) the community detection method based on mixed-norm SSC. Experiments on synthetic and real networks are reported in Section 4, followed by the conclusions in Section 5.

2 Related works

The state-of-the-art in community detection and SSC is reviewed in this section. Community detection methods are discussed with a focus on graph-based methods. Furthermore both theoretical studies and application of SSC are discussed in this section .

2.1 Community detection methods

A network is a collection of nodes that represent individual objects with different relationships among them. It can be represented by a graph G (V, E), where V is the set of vertices representing individuals in the network, and E is the set of edges representing the relationships between vertices. Networks are normally analyzed on the micro-, meso- or macroscopic levels [12-13]. At the microscopic level, the properties of edges between any two or more vertices in the network are investigated. At the macro level,

the degree of distribution or the diameter of the network are investigated. At the meso level, it is considered that how the network is structured. For example, how vertices group together into dense clusters, which are known as communities. The vertices in one cluster or community are formulated such that more edges join the vertices of that cluster, whereas fewer edges join vertices in different clusters. The aim of community detection is 1) to divide the vertex set of a network into subsets whose internal connections are denser than those of other subsets and 2) to identify these modules and their hierarchical organization using the information encoded in the graph topology [14]. Therefore, community detection is an important means of analyzing networks at the mesoscopic level.

A network that contains nodes with various connections can be represented as a graph-based structure. There is no fixed order or form to network structures because they arise in different shapes and sizes [2]. The sizes of networks can be extremely variable, and a network can be sparse or dense. How to detect communities accurately remains a challenge [15]. Main community detection algorithms include label propagation method [2], density analysis method [16], spectral bisection method [17], clique percolation method [18], modularity measure method [2], cut- and conductance-based method [19-20], spectral clustering method [21-22], (α, β)-clustering method [23], topic modeling method [24], and K-means method [25].. From the community coverage point of view, the existing algorithms can be classified into local and global community detection methods [12, 26]. Dynamic community detection has also been researched with techniques such as dynamic clustering methods [27], the objective-function optimization method [28], and the dynamic probability modeling method [29-30]. Overlapping community detection methods have also been studied [31].

Graph theory is a mature and useful tool for community detection. Most community detection algorithms associated with graph theory operate directly on the adjacency matrix of the networks. Lim et al. proposed a seed-centric community detection algorithm based on the clique percolation method [32]. Palsetiay et al. proposed an improved global community based CNM algorithm [33]. Correa et al. proposed a local community detection method based on modularity optimization from a graph [34]. Dang et al. proposed a k-nearest-neighbor-based vertex similarity approach to partition a graph into a network [35]. Natarajan et al. proposed user-specific interest identification based on interest similarity for community detection [36]. Dongen proposed a Markov cluster algorithm based on the idea of current flow in graphs [37]. Girvan and Newman proposed an edge betweenness-based community detection algorithm [4]. Deritei et al. represented the distance between two nodes using an edge-clustering coefficient and constructed Voronoi diagrams for community detection [25]. Methods of community detection in graphs can be divided into five major categories: graph partitioning, hierarchical clustering, partition clustering, spectral clustering, and divisive algorithm based on factors such as vertex similarity, edge density and distance between vertices [2, 12, 15, 17, 27-28].

The graph-based methods for community detection are intuitive and have a solid theoretical base. In most clustering-based methods for community detection, networks are not globally mapped to a space, which can cause issues in algorithm implementation. Following problems should be considered when these algorithms are applied in community detection: 1) how to detect and describe a community accurately when the difference between the degrees of internal and external nodes does not exceed the detection threshold; 2) how to address the imbalance problem in modularity maximization algorithm when the sizes of communities vary significantly; and 3) how to address the algorithm scalability when community detection algorithms are used in practice with huge datasets. Sparse subspace representation methods based on compressed sensing theory provide opportunities to address those above issues. A novel community detection algorithm based on SSC theory with mixed-norm constraints is proposed to solve these problems in this work. If nodes of networks are mapped to some similarity measure space, communities in networks can be refined as sets of nodes that span a same subspace. The formulation of subspaces is derived from the self-representation property of dataset by using SSC. Then the ADMM framework is used to solve the formulation, and communities are detected by clustering method.

2.2 Sparse subspace clustering

Compared with compression and reconstruction by transform coding with some known transform, compressed sensing uses fewer measures to infer more details of compressible objects even when the objects are under-sampled [38-39]. Many high-dimensional datasets, such as social networks, images and video, and DNA micro-array data, lie close to low-dimensional structures such as communities, moving objectives and a functional mass of genes [2-4]. SSC clusters datasets that lie in some low-dimensional subspaces [40]. By using sparse representation theory, a node in a network can be sparsely represented by other nodes from the same subspace based on the self-representation ability of data. The problem of sparse representation is formulated as an NP-hard sparse optimization program, and convex relaxation is used to approximate its solution [41-42]. Based on subspace clustering algorithms, , SSC is effective in sparse representations and reconstruction under appropriate conditions with respect to the arrangement of subspaces.

Subspace clustering (SC) provides primary clustering methods used in SSC-based community detection. An early review of SC methods was performed in the data mining community in 2004 [11]. Review of SC methods in the machine learning and computer vision community was written in 2011 [40]. In [40], SC methods are classified into algebraic algorithms [43-44], iterative algorithms [45], statistical algorithms [46], and spectral clustering algorithms [47-48]. Algebraic approach such as generalized principal component analysis fits the data with some polynomial where gradient at a point provides the normal vector to the subspace containing that point [49]. Iterative approaches such as k-subspaces alternates between assigning data points to subspaces and fitting a subspace to each cluster [50-51]. Statistical approach such as agglomerative lossy compression seeks to segment the data by minimizing the coding length to fit the points with a mixture of degenerate Gaussians [52]. In spectral clustering, local spectral clustering approach such as local similarity subspace and local linear manifold clustering use local information around each point to build a similarity between pairs of points [10]. Global spectral clustering approach solves the issue by building similarity between data points using global information [53]. Spectral curvature clustering uses multi-way similarity that captures the curvature of a collection of points in an affine subspace [48]. SC approaches give computation techniques for SSC.

SSC methods adopt sparse representation of high-dimensional data. The reason is that high-dimensional data usually lie in the unions of low-dimensional subspaces. The subspace clustering result can then be obtained through standard clustering method. By using sparse representation [54], low-rank reconstruction [55-56], low-rank recovery [57], low-rank subspace clustering [58], and SSC [10, 59], the problem of cluster identification can be formulated into finding the sparse or low-rank representation of the data by the data itself. Then, global optimization algorithms are adopted to build a similarity graph from which the data segmentation is obtained. In recent few years, simultaneously sparse coding and low-rank representation method has attracted much attention [60-64]. Parallel SSC [65], structured SSC [66], structured sparse representation [67] and local constrained low-rank representation [68] were further proposed. SSC has been successfully applied to different pattern recognition fields such as face recognition [62-63, 69], motion detection [70-71], gene expression clustering [72], system identification [10].SCC related methods have advantages in addressing noisy datasets as well as dimensions-and number of subspaces not necessarily to be known -.

Communities are usually defined as groups of nodes having more intra-group and fewer inter-group links. Thus, the value of similarity measure between two nodes in the same community will be greater than that of two nodes in two different communities. Therefore, in some similarity measure space, each community will span a subspace. In [73], each node is represented by a column of an adjacency matrix. Then, several eigenvalues and eigenvectors of Laplacian matrix are computed, and communities are determined based on a complete link-clustering algorithm. In [74], each node is represented by a vector of the geodesic distances with other nodes. The similarity matrix is computed by using sparse linear coding. The community is then detected using the spectral clustering algorithm. We think each community spans a low-dimensional subspace with some similarity measure. Different similarity measure can be adopted to present the relationship of each node to other nodes in a network. All the

nodes are represented by the linear combination of the other nodes that span the same subspaces. By introducing a novel mixed-norm constraint, the coefficients of node representation can be used to cluster different communities, in which the connections of nodes between different communities are modeled as noises to improve the clustering accuracy. By using SCC theory, each community is identified as a concrete low-dimensional subspace in measure space. Experiments in section 4 demonstrate the effectiveness of the proposed method.

3 The proposed method

3.1 Outline

Subspace theory provides a new direction for pattern recognition, where a low-dimensional representation can be found for high-dimensional dataset. For example, the principal component analysis (PCA) method assumes that all the data are drawn from a single low-dimensional subspace of a high-dimensional space. However, different subspaces are needed to describe different objects in practical uses such as detecting multiple moving objects from a video sequence [71]. Due to data points in each subspace being distributed randomly most of the time, standard clustering method that adopts spatial proximity measure to identify the data points in each cluster is not suitable to subspace clustering anymore. Therefore, the dataset should be simultaneously clustered into multiple subspaces, and each subset should be fitted by a low-dimensional subspace. SSC addresses the clustering problem by finding sparse representations of the dataset itself.

The proposed method is outlined as Figure 1. A community is considered as a group of nodes that have more intra-group and fewer inter-group link similarities, which can be denoted as different subspace. To this end, a similarity measure between all pairs of nodes in the network is initially adopted to formulate the problem. The similarity measure is smaller between nodes in a same community than that between nodes in two different communities. By mapping the adjacency matrix to similarity measure space, each community will span a different subspace. Based on the similarity measure, the apparent dimensionality of each node is as same as the total number of nodes in the network, which is represented by a vector determined by the similarity measure between each node and all the other nodes in the network. Actual dimensionality of a node in the corresponding subspace may be smaller than its apparent dimensionality. A node can be represented as a linear combination of other nodes which spanning the subspace of the same community. If a network is decomposed into several communities, the coefficients of the vector representation in different other subspaces will become zeros. But different communities in a network are usually not separated rigidly. Linear decomposition approaches using base functions with least square error may result in some non-zero coefficients of small magnitudes for some nodes in different communities. In order to deal with the problem, the connections of nodes between different communities are modeled as noise. Community detection is formulated as an estimation of the coefficients of sparse linear representation, which minimizes representation error by the mixed-norm constraints. Then, each node in the network is represented as a linear combination of all the other nodes, and the coefficients of measures make up a similarity matrix. Finally, spectral clustering is used to partition the graph represented by the similarity matrix into several clusters. The number of clusters is estimated by the reduction rate of clustering error. In reality, complex networks show the small world phenomenon [2], which makes it more challenging to identify the accurate community. To address this challenge, mixed-norm SSC with noise is proposed to improve the accuracy of community detection.

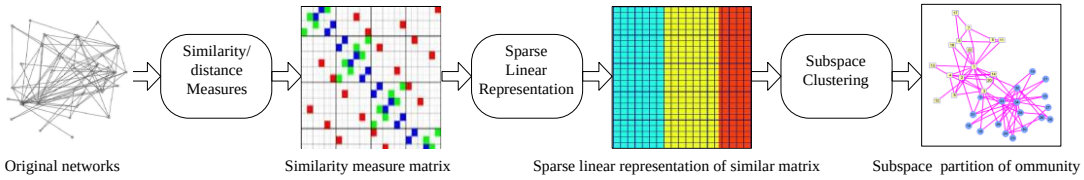


Fig. 1 Outline of the proposed community detection method

3.2 Similarity measure matrix

Almost all clustering algorithms rely on similarity or distance measures to determine the assignment of vertices to a cluster. Similarity measures are the basis of community detection methods such as hierarchical, partition and spectral clustering. Similarity can be computed based on some reference property of the measures such as Euclidean distance, Manhattan distance, maximum distance, or angle cosine similarity.

Considering an un-weighted and undirected network G with n vertices and m edges denoted by an adjacency matrix $\mathbf{A}^{n\times n}$, if there exists an edge between two vertices $\{v_i, v_j\}$, then $\mathbf{A}_{i,j}=1$; otherwise, $\mathbf{A}_{i,j}=0$. Adjacency matrix $\mathbf{A}$ is symmetric, and each vertex $v_i \in G$ is determined by the corresponding column or row vector of the adjacency matrix in $\{0,1\}^n, i=1,2,...,n.$ Different similarity measures derive different spaces. Distance measures usually satisfy the following three properties: non-negativity, symmetry and triangular inequality. Similarity between a pair of vertices usually is the function of their distance. Given two vertices v_i and v_j denoted by $V_i=(a_{i,1},a_{i,2},...,a_{i,n})^T$ and $V_j=(a_{j,1},a_{j,2},...,a_{j,n})^T$, norm $\mathbf{L}_m, m\in Z^+$ can be used to define a distance such as the Euclidean distance ($\mathbf{L}_2$-norm)

$$d^E_{v_iv_j}=(\sum_{k=1}^{n}(a_{i,k}-a_{j,k})^2)^{1/2}, \tag{1}$$

the Manhattan distance ($\mathbf{L}_1$-norm)

$$d^M_{v_iv_j}=\sum_{k=1}^{n}|a_{i,k}-a_{j,k}|, \tag{2}$$

and the $\mathbf{L}_\infty$-norm

$$d^\infty_{v_iv_j}=\max_{k\in[1,n]}|a_{i,k}-a_{j,k}|. \tag{3}$$

Another similarity measure is angle similarity, defined as

$$\rho_{v_iv_j}=\frac{\sum_{k=1}^{n}a_{i,k}a_{j,k}}{(\sum_{k=1}^{n}(a_{i,k})^2)^{1/2}(\sum_{k=1}^{n}(a_{j,k})^2)^{1/2}}, \tag{4}$$

where the variable $\rho_{v_iv_j}$ is in the range $[0,\pi)$.

Based on the concept of structural equivalence [2], a distance measure can be notated as

$$d_{v_iv_j}=(\sum_{k\neq i,j}(a_{i,k}-a_{j,k})^2)^{1/2}. \tag{5}$$

The measure (5) denotes that vertices with large degrees and different neighbors are considered quite far from each other, while two vertices are structurally equivalent if they have the same neighbors, even when they are not adjacent.

Other measures of vertex structural equivalence include the Pearson correlation between the columns or rows of the adjacency matrix, the number of edge- (or vertex-) independent paths between two vertices, and so on [14]. Geodesic distance is used to represent the similarity of vertices in [74]. The similarities of each vertex to all others are represented by a column in a matrix of similarity scores in which geodesic distances are mapped to a similarity measure by a Gaussian kernel function.

In community detection, distance (or similarity) measures are deduced from the adjacency matrix of a network. Let the vector $d_i=(d_{i,1},d_{i,2},...,d_{i,n})^T$ be the set of distance measures for v_i from all v_j in network G. In general, the map from the adjacency matrix to similarity measures matrix one-by-one. However, the latter provides additional clearly inferred information about the structure of the dataset. The similarity measures matrix can be easily constructed. For example, in our work, similarity measure $s_{i,j}$ between v_i and v_j deduced from $d_{i,j}$ is

$$s_{i,j} = 1 - \frac{d_{i,j}}{n}, \tag{6}$$

where n is the number of vertices in network G. And $d_{i,i}$ is defined as zero because it is assumed there are no self-loops in the network.

Subspace method provides a new direction for community detection of networks. Although any one-to-one mapping can affect the form of the original dataset, the intrinsic information will not change. An adjacency matrix of networks can initially be mapped into different similarity matrix in which each vertex is mapped to a unique point in a similarity measure space. Then, SSC is used to approximate each community by spanning subspace determined by the similarity measure. In SSC method, $\mathbf{L}_F$-norm is used to formulate the similarity measure spaces. Furthermore, all the nodes are represented by a linear combination of the nodes which spanning the same subspace by introducing a mixed-norm constraint. In the proposed method, $\mathbf{L}_1$-norm and $\mathbf{L}_F$-norm of matrix $\mathbf{A}$ are denoted as

$$\|\mathbf{A}\|_1 = \max\{\sum_{i=1}^{n}|\mathbf{A}_{i,1}|, \sum_{i=1}^{n}|\mathbf{A}_{i,2}|, ..., \sum_{i=1}^{n}|\mathbf{A}_{i,n}|\} \tag{7}$$

and

$$\|\mathbf{A}\|_F = (\sum_{i=1}^{n}\sum_{j=1}^{n}\mathbf{A}_{ij}^2)^{\frac{1}{2}} \tag{8}$$

are adopted to formulate the sparse linear representation model, because $\mathbf{L}_1$-norm Eqw.(7) and $\mathbf{L}_F$-norm Eq.(8) have lower computation cost than $\mathbf{L}_2$-norm of matrix [75].

3.3 Sparse representation of similarity measure

If each node in a network is represented by a column vector in the similarity measure matrix $\mathbf{S}$, the problem of community detection in networks can be solved as the sparse subspace clustering formulation. The general representation-based clustering methods initially solve a coefficient matrix $\mathbf{C} \in \mathbf{R}^{n \times n}$ for data matrix $\mathbf{Y} = [\mathbf{y}_1, \mathbf{y}_2, ...\mathbf{y}_n] \in \mathbf{R}^{d \times n}$, where d denotes the feature dimension and n is the number of data points. The matrix $\mathbf{S}$, of which entries measure the similarity, is reconstructed. SC is then further performed on the coefficient matrix $\mathbf{C}$ to segment data. The coefficient matrix $\mathbf{C}$ can be obtained by solving the following general minimization problem [71]

$$\min f(\mathbf{C}) + \lambda g(\mathbf{Y} - \mathbf{YC}), \tag{9}$$

where $f(\mathbf{C})$ denotes a penalty function, $g(\mathbf{Y} - \mathbf{YC})$ corresponds to a loss function, and λ is a parameter used to trade off the two terms. The domain of the values of parameter λ is estimated in [76] for a concrete linear case of f and g.

In community detection, SSC algorithm is used as clustering similarity measure of multi-subspaces by using sparse linear representation. The similarity measure of each pair of nodes is assumed to lie in a union of linear subspaces. Let $\{\mathbf{S}_k\}_{k=1}^{N}$ be an arrangement of N linear subspaces of $\mathbf{R}^n$ with the dimensions $\dim_k$, $k = 1, 2, ..., N$. Consider a given matrix of similarity measure $\mathbf{S} \in [0,1]^{n \times n}$; the linear representation formulation of the similarity measure $\mathbf{S}$ is

$$\mathbf{S}@[y_1, y_2, ...y_n] = [\mathbf{Y}_1, \mathbf{Y}_2, ..., \mathbf{Y}_N]\mathbf{C}, \tag{10}$$

where $\mathbf{Y}_k \in \mathbf{S}_k$ is a rank $\dim_k$ matrix of the similarity measures that lie in $\mathbf{S}_k$, $k = 1, 2, ..., N$ and $\mathbf{C}$ is an unknown coefficient matrix. In community detection, neither a priori information about the basis of the subspaces nor the data belong to the subspaces is known. The formulation of the subspace clustering method for community detection is intended to find the number of subspaces, their dimensions, a basis for each subspace, and the segmentation of the dataset $\mathbf{S}$.

To solve the community detection problem by SSC, the similarity measure of each data is initially

represented by a few other similarity measure vectors that belong to the same subspace. To this end, a global sparse optimization program is adopted. Then, a spectral clustering framework is used to infer the clustering of the matrix of similarity measure. By taking the advantage of self-representation property of the data in SSC, the data for each community are assumed to lie in a union of subspaces where each data point can be efficiently represented by a combination of other points in the dataset [10]. Let each data be denoted by $y_i \in \{\mathbf{S}_k\}_{k=1}^{N}, i = 1, 2, ..., n$, and written as

$$y_i = \mathbf{Y}\mathbf{C}_i, \quad c_{ii} = 0 \tag{11}$$

where $\mathbf{C}_i = [c_{i1}, c_{i2}, ..., c_{in}]^T$ and $\mathbf{Y} = [\mathbf{Y}_1, \mathbf{Y}_2, ..., \mathbf{Y}_N]$. The constraint $c_{ii} = 0$ eliminates the trivial solution of representing a node as a linear combination of itself. The similarity measure matrix $\mathbf{S}$ is a self-representation dictionary where each similarity measure vector is a linear combination of other points. The representation of y_i in the dictionary $\mathbf{Y}$ is not generally unique because the number of data points in a subspace is often greater than its dimensions. In other words, a nontrivial null-space can provide an infinite number of representations of each data point. However, by using sparse subspace representation theory, a data point y_i that lies in subspace $\mathbf{S}_k$ with dimensions $\dim_k$ can be written as a linear combination of other points from $\mathbf{S}_k$. Then, Eq. (11) can be solved by minimizing $\mathbf{L}_q$-norm objective function such as

$$\min |\mathbf{C}_i|_q \quad \text{s.t.} \quad y_i = \mathbf{Y}\mathbf{C}_i, \quad c_{ii} = 0. \tag{12}$$

Different values of q have different effects on the solution. Because the values of q decrease from infinity toward zero, then the sparsity of the solution increases. The $\mathbf{L}_0$-norm counts the number of nonzero elements of the solution and corresponds to finding the sparsest representation, which is generally an NP-hard problem [69, 71]. In practice, the convex relaxation formulation $\mathbf{L}_1$-norm of the $\mathbf{L}_0$-norm is adopted to approximate the nontrivial sparse representation of y_i in subspace $\mathbf{S}_k$ such as

$$\min \|\mathbf{C}_i\|_1 \quad \text{s.t.} \quad y_i = \mathbf{Y}\mathbf{C}_i, \quad c_{ii} = 0 \tag{13}$$

Eq. (13) can be solved efficiently by using convex programming [64-67, 69] and can be rewritten as the $\mathbf{L}_2$-norm of the representation error

$$\min |y_i - \mathbf{Y}\mathbf{C}_i|_2 \quad \text{s.t.} \quad |\mathbf{C}_i|_1 \le \beta, c_{ii} = 0 \tag{14}$$

where β is positive constant. Ideally, the solution of Eq. (14) corresponds to the subspace sparse representations of the matrix of similarity measures. However, in the problem of community detection with real networks, some connections always exist between different communities. We modeled the clustering of different communities which are contaminated with sparse noise entries denoted as the connections between different communities. Let error-free point y_i lie perfectly in some subspace shown as

$$y_i^0 = \sum_{j \ne i} c_{ij} y_j^0, \tag{15}$$

and y_i be a node that is obtained by corrupting an error-free point y_i^0 with a vector of sparse noise entries $\mathbf{E}_i$ that has only a few large nonzero elements. Then, we have

$$y_i = \sum_{j \ne i} c_{ij} y_j^0 + \mathbf{E}_i. \tag{16}$$

This situation in community detection corresponds to the case that data are contaminated by the noise from connections between different communities. SSC does not attempt to present a node based on a linear combination of other nodes. Instead, a penalty noted with $\mathbf{L}_2$-norm of the error is added to the $\mathbf{L}_1$ norm of coefficient. The corresponding objective function is

$$\min |\mathbf{C}_i|_1 + \lambda |\mathbf{E}_i|_2 \quad \text{s.t. } y_i = \mathbf{Y}\mathbf{C}_i, \ c_{ii} = 0, \tag{17}$$

where λ is a Lagrange parameter. Then, Eq.(17) can be rewritten as the $\mathbf{L}_2$ -norm of the representation error

$$\min \|y_i - \mathbf{Y}\mathbf{C}_i\|_2 \quad \text{s.t. } |\mathbf{C}_i|_1 \le \beta_1, \ |\mathbf{E}_i|_2 \le \beta_2, \ c_{ii} = 0 \tag{18}$$

where β_1 and β_2 are positive constants. Eq. (13) and Eq.(17) can be rewritten in matrix form as

$$\min \|\mathbf{C}\|_1 \quad \text{s.t. } \mathbf{Y} = \mathbf{Y}\mathbf{C}, \operatorname{diag}(\mathbf{C}) = 0 \tag{19}$$

and

$$\min \|\mathbf{C}\|_1 + \lambda \|\mathbf{E}\|_F \quad \text{s.t. } \mathbf{Y} = \mathbf{Y}\mathbf{C} + \mathbf{E}, \operatorname{diag}(\mathbf{C}) = 0. \tag{20}$$

Following, the ADMM framework is used to solve Eq. (20) or Eq. (19), which iteratively solves convex optimization problem by introducing auxiliary variables [77]. The solution of optimal problem Eq. (20) is formulated as

$$\arg\min_{\mathbf{C},\mathbf{E},\mathbf{U},\mathbf{Z}} \frac{\lambda}{2}\|\mathbf{Y} - \mathbf{Y}\mathbf{C} - \mathbf{E}\|_F^2 + \|\mathbf{Z}\|_1 + \lambda_e \|\mathbf{U}\|_1 \quad \text{s.t. } \mathbf{C} = \mathbf{Z}, \mathbf{E} = \mathbf{U}, \operatorname{diag}(\mathbf{C}) = 0 \tag{21}$$

where $\mathbf{Z}$ and $\mathbf{U}$ are auxiliary variables. And the augmented Lagrangian function $f(\mathbf{C},\mathbf{E},\mathbf{U},\mathbf{Z})$ is denoted as

$$f(\mathbf{C},\mathbf{E},\mathbf{U},\mathbf{Z}) = \frac{\lambda_{\mathbf{Y}}}{2}\|\mathbf{Y} - \mathbf{Y}\mathbf{C} - \mathbf{E}\|_F^2 + \|\mathbf{Z}\|_1 + \lambda_e \|\mathbf{U}\|_1 + \langle \mathbf{A}_{\mathbf{C}}, \mathbf{C} - (\mathbf{Z} - \operatorname{diag}(\mathbf{Z}))\rangle$$

$$+ \frac{\lambda_{\mathbf{C}}}{2}\|\mathbf{C} - (\mathbf{Z} - \operatorname{diag}(\mathbf{Z}))\|_F^2 + \langle \mathbf{A}_{\mathbf{E}}, \mathbf{E} - \mathbf{U}\rangle + \frac{\lambda_{\mathbf{E}}}{2}\|\mathbf{E} - \mathbf{U}\|_F^2 \tag{22}$$

where $\mathbf{A}_{\mathbf{C}}$ and $\mathbf{A}_{\mathbf{E}}$ are the multiplier matrixes, constant $\lambda_{\mathbf{Y}}$, $\lambda_{\mathbf{C}}$, λ_e and λ_E are positive penalty parameters and $\langle \mathbf{A}_1, \mathbf{A}_2 \rangle$ is the trace $\mathbf{A}_1^T \mathbf{A}_2$. The alternating directions iterations of multipliers are described as the following updating steps.

Step one: Updating $\mathbf{C}$

When other parameters are given in the k th iteration, and the objective function $f(\mathbf{C},\mathbf{E},\mathbf{U},\mathbf{Z})$ is minimized with respect to $\mathbf{C}$. The linear equation of the $\mathbf{C}^{(k+1)}$ can be given as

$$\left(\lambda_{\mathbf{Y}}\mathbf{Y}^{\mathbf{T}}\mathbf{Y} + \lambda_{\mathbf{C}}\mathbf{I}\right)\mathbf{C}^{(k+1)} = \lambda_{\mathbf{Y}}\mathbf{Y}^{\mathbf{T}}\left(\mathbf{Y} - \mathbf{E}^{(k)}\right) + \lambda_{\mathbf{C}}\left(\mathbf{Z}^{(k)} - \operatorname{diag}\left(\mathbf{Z}^{(k)}\right)\right) - \mathbf{A}_{\mathbf{C}}^{(k+1)} \tag{23}$$

where $\mathbf{I}$ is the suitable identity matrix. If the k th value of other parameters are given, the $k+1$ th value $\mathbf{C}^{(k+1)}$ of $\mathbf{C}$ can be computed by the operation of matrix inversion or conjugate gradient methods.

Step two: Updating $\mathbf{E}$

Similar with Eq.(23), when other parameters are fixed, and the objective function $f(\mathbf{C},\mathbf{E},\mathbf{U},\mathbf{Z})$ is minimized with respect to $\mathbf{E}$. The $k+1$ th value $\mathbf{E}^{(k+1)}$ is computed as

$$\mathbf{E}^{(k+1)} = \left(\lambda_{\mathbf{Y}} + \lambda_{\mathbf{E}}\right)^{-1}\left(\lambda_{\mathbf{Y}}\mathbf{Y} - \lambda_{\mathbf{Y}}\mathbf{Y}\mathbf{C}^{(k+1)} + \lambda_{\mathbf{E}}\mathbf{U}^{(k)} - \mathbf{A}_{\mathbf{E}}^{(k)}\right) \tag{24}$$

Step three: Updating auxiliary variable $\mathbf{Z}$

The value of variable $\mathbf{Z}$ is computed by the shrinkage-thresholding operator for each element of the given matrix [69, 77] as

$$\mathbf{Z}^{(k+1)} = J - \operatorname{diag}(J) \tag{25}$$

where

$$J = \Phi_{\frac{2}{\lambda_{\mathbf{C}}}}(\mathbf{C}^{(k+1)} + \frac{2}{\lambda_{\mathbf{C}}}\mathbf{A}_{\mathbf{c}}^{(k)}) \tag{26}$$

and $\Phi_{\tau}(\mu) := \max\{|\mu| - \tau, 0\}\operatorname{sgn}(\mu)$ is the shrinkage-thresholding operator.

Step 4: Updating auxiliary variable $\mathbf{U}$

Similarly with variable $\mathbf{Z}$, the value of variable $\mathbf{U}$ is computed as

$$J = \Phi_{\frac{\lambda_e}{\lambda_{\mathbf{E}}}}(\mathbf{E}^{(k+1)} + \frac{2\lambda_e}{\lambda_{\mathbf{E}}}\mathbf{A}_{\mathbf{E}}^{(k)}) \tag{27}$$

Step 5: Updating the multipliers matrix $\mathbf{A}_{\mathbf{C}}$ and $\mathbf{A}_{\mathrm{E}}$

The multipliers matrix $\mathbf{A}_{\mathbf{C}}$ and $\mathbf{A}_{\mathrm{E}}$ are updated by gradient ascent method such as

$$\mathbf{A}_{\mathbf{C}}^{(k+1)} = \mathbf{A}_{\mathbf{C}}^{(k)} + \lambda_{\mathbf{C}}(\mathbf{C}^{(k+1)} - \mathbf{Z}^{(k+1)}) \tag{28}$$

$$\mathbf{A}_{\mathbf{E}}^{(k+1)} = \mathbf{A}_{\mathbf{E}}^{(k)} + \lambda_{\mathbf{E}}(\mathbf{E}^{(k+1)} - \mathbf{U}^{(k+1)}) \tag{29}$$

The solution of Eq.(19) is a simple version of that of Eq. (20). Furthermore the solutions of Eq. (19) and Eq.(20) correspond to subspace sparse representation coefficients of the similarity measure matrix, which will be used to infer the clustering.

3.4 SC of the sparse representation coefficients

After solving the optimization program of Eq. (19) or Eq. (20), a sparse representation for each node is obtained in which the nonzero elements correspond to nodes from the same subspace. Then, the data are divided into different subspaces using the sparse coefficients. A community containing a node y_i is a set of nodes containing the support set of y_i as its subset. The support set is the collection of nodes that correspond to those large coefficients. Two nodes belong to the same community if their similarity measure is greater than some given thresholds. Therefore, all nodes sharing a larger similarity measure correspond to the common community. Community is then the union of the support set of each pair of nodes. To detect different communities in a network, the sparse linear coefficient vectors of $\mathbf{C}$ are clustered.

Because c_{ij} may be slightly different from those of c_{ji}, the normalization and symmetry operations are adopted to modify the representation matrix $\mathbf{C}$ as

$$\mathbf{F}_{i,j} = \frac{1}{2}(\frac{c_{ij}}{|\mathbf{C}_i|} + \frac{c_{ji}}{|\mathbf{C}_j|}), \tag{30}$$

where c_{ij} and c_{ji} are the j th and i th entry of the i th and j th row $\mathbf{C}_i$ and $\mathbf{C}_j$ of $\mathbf{C}$. Then, $\mathbf{F}$ is a symmetric matrix, and the SC method is adopted to detect communities. For this purpose, a degree matrix $\mathbf{D}$ is computed as

$$\mathbf{D}_{i,j} = \begin{cases} \sum_{i=1}^{n}\mathbf{F}_{i,j} & if\ \ i = j \\ 0 \quad \text{if} \ \ i \neq j \end{cases}. \tag{31}$$

Using $\mathbf{F}$ and $\mathbf{D}$, a Laplacian matrix $\mathbf{L}_s$ is computed as

$$\mathbf{L}_s = \mathbf{I} - \mathbf{D}^{1/2}\mathbf{F}\mathbf{D}^{1/2}. \tag{32}$$

The eigenvectors of $\mathbf{L}_s$ embed the vertices of a graph into the Euclidean space. The second-least-significant eigenvector of $\mathbf{L}_s$ is the Fiedler vector. Based on the NCut criterion [2, 4, 78], the network can be divided into two partitions. The Fiedler vector is used to discover the hierarchical structure of networks. The above process can be repeated to divide each part into two new partitions. Alternatingly, one can select the k least-significant eigenvectors of $\mathbf{L}_s$ and directly compute k clusters.

3.5 Details of community detection

From pervious analysis, the proposed community detection method can be drawn as the following algorithm.

Algorithm 1: Community detection based on the mixed-norm SSC

Input: $\mathbf{A}^{n\times n} \in \{0,1\}^n \times \mathrm{L} \times \{0,1\}^n$ adjacency matrix of network

Step 1: Compute the similarity measure matrix $\mathbf{S}$ from the adjacency matrix $\mathbf{A}$;

Step 2: Solve the coefficients matrix $\mathbf{C}$ of sparse linear representation by the optimization program (19) or (20);

Step 3: Normalize and symmetry the matrix $\mathbf{C}$ to obtain the symmetric linear coefficients matrix $\mathbf{F}$;

Step 4: Apply spectral clustering to the similarity matrix $\mathbf{F}$.

Output: Segmentation of the original dataset denoted by the network adjacency matrix

---

In the ideal situation that a network contains some isolated communities with no inter-community links, the subspace spanned by each community is independent. As a result, a node in a particular community has nonzero coefficients in those nodes within the same community. The coefficients corresponding to nodes in other communities are zero. In such cases, SSC associated with the optimization program Eq.(19) is adopted. However, for community detection in practical complex networks such as human relationships, the small-world phenomenon of dense networks generally makes accurate community detection challengeable or even impossible. Traditional spectral clustering algorithm is fused with the geodesic distance-based algorithm to mitigate the effect of inter-community links in [74]. However, information for the both comes from the adjacency matrix. Computation of double cluster and inverse exponential operation take a significant amount of time. In the proposed method, different distance measures can increase the flexibility of similar descriptions. Furthermore, $\mathbf{L}_F$-norm of error $\mathbf{E}$ is added to the $\mathbf{L}_1$-norm of the coefficients matrix $\mathbf{C}$, which improves the accuracy of community detection. In the case of community detection for networks in the following experiments, SSC associated with optimization program Eq.(20) is suitable. The proposed algorithms have shown excellent performance, particularly in computation cost. $\mathbf{L}_F$-norm of error $\mathbf{E}$ and $\mathbf{L}_1$-norm of the coefficients matrix $\mathbf{C}$ are used in the optimization program Eq.(20). Therefore, the proposed method is denoted as mixed-norm SSC.

Among the computation processes in the proposed Algorithm 1, step one and step three are facile. In step two, the sparse linear representation of Eq.(19) or Eq.(20) can be solved using implementations of various techniques such as ADMM, the augmented Lagrange multiplier (ALM) method, and the iterative shrinkage/thresholding (IST) method [41, 69]. In this work, ADMM method is adopted. ADMM iteratively solves convex optimization problems with a global solution by breaking them into smaller, easier-to-solve problems. The parameter λ in Eq.(20) balances the two terms in the objective function. The domains and the roles of the parameters in the Lagrange objective function are also estimated in [76]. MATLAB code of ADMM is available from us or in the SPArse Modeling Software (SPAMS) package [79]. In step four, SSC need not initially know the number of subspaces. We select the K least-significant eigenvectors of $\mathbf{L}_s$ to determine the number of clusters. The eigenvectors are normalized to be unit magnitude and then clustered by the linear clustering algorithm.

The average clustering error is used to determine an appropriate value for K. Let $\mathbf{m}_k$ be the center of cluster $\mathbf{C}_k$ and $\xi_i \in \mathbf{C}_k$ be the eigenvectors of cluster $\mathbf{C}_k$. The average error of clustering is

$$\varepsilon = \sqrt{\sum_{k=1}^{K} \sum_{i:\xi_i \in C_k} |\xi_i - \mathbf{m}_k|^2} \ . \tag{33}$$

When the values of K increase, the values of ε clearly decrease significantly at first. When K is 1, ε reaches maximum. When K is the number of nodes, ε is 0. Experimental results in the next section show that after a specific value of K, the error will decreases slowly. The error reduction rate gives the efficient number approximation of clustering.

4 Experimental Results and Discussion

Experiments are performed on real labeled networks, real unlabeled networks and synthetic networks to evaluate the efficiency of the proposed approach. Two versions of the proposed method, denoted as the noise-free SSC model Eq.(19) and the mixed-norm SSC model Eq.(20) are implemented. The proposed algorithms are compared with the state-of-the-art methods such as the sparse linear coding method (SLC) [74], the GN algorithm [3], the Newman fast greedy algorithm (FG) [8], and the Infomap algorithm (IM) [20]. The performance of the proposed algorithms is compared with those of the state-of-the-art methods on both standard benchmark networks with 4 real labeled networks, 2 real unlabeled networks and 2 synthetic networks such as GN Benchmark [4] and LFR Benchmark [81]. Information about the benchmark networks is shown in table 1. In the labeled networks, each node has a ground-truth community label. The detected communities are compared with the ground-truth communities using precision rate, recall rate and normalized mutual information (NMI). In the unlabeled networks, each node does not have a ground-truth label. The algorithm cannot be compared with existing algorithms directly based on precision rate and recall rate. Therefore, an evaluation of modularity is used. For most of the algorithms, experimental results are reported from the original authors or from comparative studies [5, 15]. Then experiments on synthetic networks of GN Benchmark mainly show how the number of clustering is determined by the reconstruction error of the community detection schemes. Experiments on LFR Benchmark show that the proposed algorithm is effective with different sizes of networks. At last, convergences of the noise-free SSC model Eq.(19) and the mixed-norm SSC model Eq.(20) are compared. All of the programming is processed using MATLAB 2013 on an Intel(R) Core(TM) i5 3.10 GHz CPU with 3 GB RAM.

Table 1 Benchmark networks used in the experiments

Network	Abbreviation	Vector	Edge	Label
Zachary's Karate club [74]	Karate	34	78	Yes
Dolphin social network [82]	Dolphin	62	159	Yes
American college football [15]	Football	115	616	Yes
Books about US politics [3]	Politics	105	441	Yes
Email communication network [83]	Email	1133	5451	No
Co-authorships in network science [84]	Net science	1589	2742	No
GN Benchmark [4]	GN	Not given	Not given	No
LFR Benchmark [81]	LFR	Not given	Not given	No

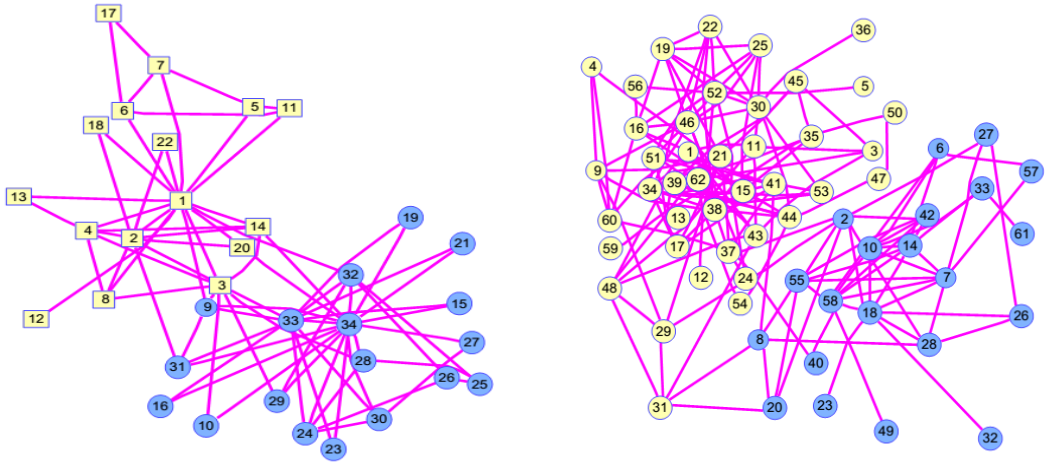
4.1 Results on labeled networks

In the datasets of labeled networks, Zachary Karate Club is a benchmark social network of friendships in a karate club. There are 34 nodes and 78 links in the network. The club splits into two new groups named Mr. Hi and John A [74]. The wide Kiss dolphin network provides the social relationships of 62 wide Kiss dolphins with 159 links. The network contains two small communities [82]. The American college football dataset is a network of football games between Division IA colleges during the regular season in 2000 [15]. There are 115 nodes and 616 links in the network. The network is divided into 12 groups, corresponding to the 12 leagues. The network of US political books is based on political book sales records from Amazon and includes 105 nodes and 441 edges [3]. In accordance with the political leanings of the books, the network is divided into three groups: freedom, neutrality and conservative. As

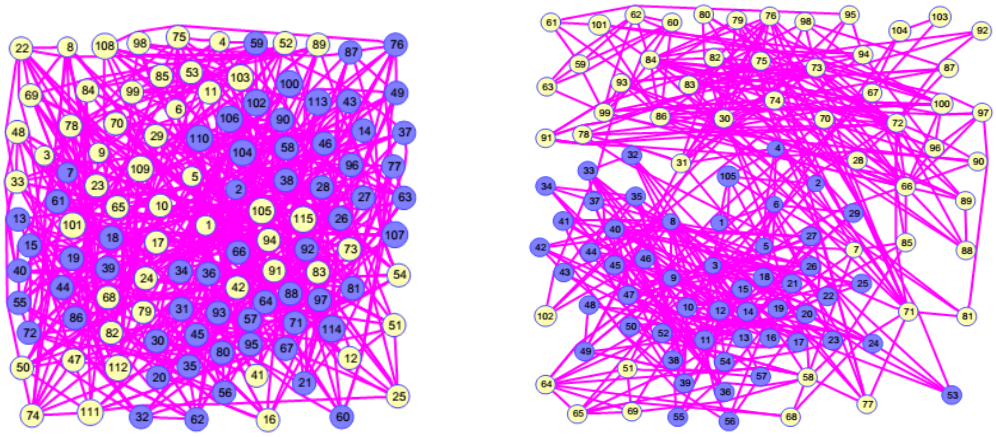
for the labeled networks, precision rate, recall rate and NMI are used to compare the proposed algorithms with the reference algorithms. Table 2 presents the community detection results of the proposed algorithm and the reference algorithms on the labeled networks. Fig. 2 shows the reconstruction results of two communities from four datasets through the proposed method using Eq.(20). Table 2 shows that all the algorithms perform better on the Karate and Dolphin datasets than on the Football and Politics datasets. Fig. 2 shows that the community structures of the first two networks are clearer than are those of the last two. The numerical comparing results are shown in table 2. There are fewer connections between communities in the first two networks than in the last two. We can see that the proposed method modeled with Eq.(20) performs well, particularly on the first dataset. The number of communities in these experiments is set in advance to be two. The results illustrate that the proposed method effectively addresses the problem of community detection in networks.

Table 2 Community detection results of the proposed algorithm and the state-of-art algorithms on labeled networks

Network	Evaluation	GN	FG	IM	SLC	Eq. (19)	Eq. (20)
Karate	precision	0.971	0.971	0.971	0.971	0.971	**1**
	recall	0.969	0.971	0.971	0.971	0.971	**1**
	NMI	0.836	0.837	0.837	0.837	0.837	**1**
Dolphin	precision	0.984	0.935	0.968	0.984	0.984	**0.968**
	recall	0.977	0.94	0.954	0.977	0.977	**0.954**
	NMI	0.89	0.652	0.814	0.89	0.89	**0.814**
Football	precision	0.834	0.573	0.930	0.878	0.887	**0.896**
	recall	0.772	0.49	0.891	0.816	0.832	**0.825**
	NMI	0.878	0.697	0.936	0.793	0.821	**0.865**
Politics	precision	0.857	0.838	0.848	0.857	0.876	**0.876**
	recall	0.765	0.639	0.721	0.747	0.765	**0.786**
	NMI	0.568	0.568	0.571	0.584	0.588	**0.597**



(a) (b)



(c) (d)

Fig. 2 Community structures of four benchmark networks: (a) Karate, (b) Dolphin, (c) Football and (d) Politics

4.2 Results on unlabeled networks

In the unlabeled real networks, there is no ground-truth label for each node in the networks. Therefore, comparison of different algorithms is difficult. Indexes such as precision rate, recall rate and NMI cannot be used to compare the performance of different algorithms. Comparisons have typically been made based on the modularity function [3]. The unlabeled networks contain a Spanish university email communication network and a network of web science collaboration. The email communication network consists of 1133 nodes and 5451 edges. The cooperative network describes the cooperative relationships of 1589 scientists from the field of web of Science, which consists of 1589 nodes and 2742 edges. In the dataset of the Email communication network, the number of communities is set as 11. In the dataset of the cooperative network, the number of communities is set as 12. Experimental results on the modularity of different algorithms for the two unlabeled networks are shown in Table 3. We see that the proposed algorithm is as good as the SLC algorithm from Table 3, and both outperform the others algorithms. For the dataset of cooperative network, the GN algorithm cannot divide the network societies because the network is not connected. The corresponding value of the modularity function is not shown.

Table 3 Values of modularity of community detection results on unlabeled networks

Network	GN	FG	IM	SLC	Eq. (19)	Eq. (20)
Email	0.532	0.506	0.52	0.545	0.538	**0.544**
Network science	-	0.955	0.931	0.957	0.951	**0.958**

4.3 Results on the GN benchmark

The modularity function may not be an especially appropriate measure for the effect of community detection, particularly in situations that the sizes of the communities vary significantly [5]. Therefore, apart from the modularity function, average error of clustering is used to evaluate the performance of the community detection algorithm. Experiments using the GN benchmark illustrate how to determine the number of communities for a given network. The GN benchmark has 128 nodes and four communities, each with 32 nodes. Each node has one probability of being connected to the nodes in the same community and another probability of being connected to the nodes of different communities. The total degree of each node is 16. The mixing parameter m is defined as the ratio of the external degree of a node to its total degree. The structure is well defined for small values of m [4]. When m is less than 6/16, almost all community detection algorithms yield a 100% correct result. However, when m is greater than 9/16, the community structure becomes subtle, and most algorithms cannot find any

meaningful communities. We report on the proposed method when m is 8/16. Average error of clustering along with the number of clustering using Eqs. (19) and Eq.(20) is shown in Fig. 3. The average error variation rate of clustering along with the number of clustering using Eq.(19) and Eq.(20) is shown in Fig. 4. Fig. 3 and Fig. 4 show that average error decreases with an increase in the number of communities. The rate of average error of clustering reduces significantly when the number is less than 4, which is the truth number of communities in the network. Experiments are repeated ten times. Fig. 3 and Fig. 4 show that the average error-reduction rate can be used as a measure to determine the number of communities. In addition, the mixed-norm of Eq. (20) yielded a slightly clearer result than did the basic model denoted as Eq.(19).

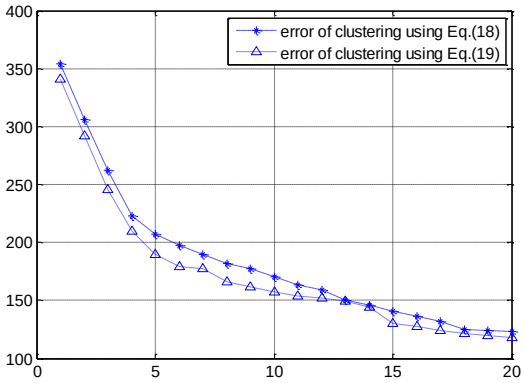


Fig. 3 Average error versus the number of clustering using Eq.(19) and (20) on the GN benchmark

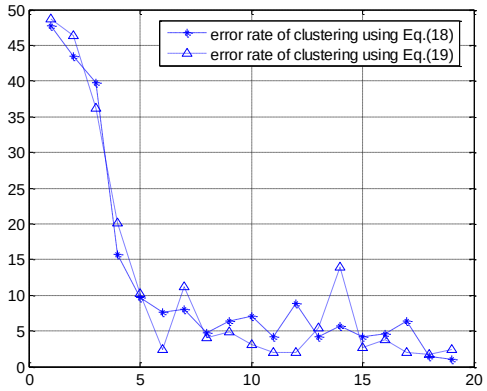


Fig. 4 Average error variation rate versus the number of clustering using Eq.(19) and (20) on the GN benchmark

4.4 Results on LFR benchmark

Experiments are performed on LFR synthetic benchmark networks to show the proposed method is effective for different scales of networks. The degree of distribution and community scale distribution of nodes in the LFR network follow the power-law distribution, which makes it closer to real networks [81]. The following parameters should be set to generate synthetic complex networks: the number of nodes in the network, n; the average degree of network nodes, k; the maximum degree of network nodes, $k_{\max}$; the network topological mixing parameter λ, which specifies the structure of the communities; the power law distribution parameter for the node degree, λ_1; the power law distribution parameter for community size index, λ_2; the minimum community size, $R_{\min}$, which specifies the minimum size of each generated community; and the maximum community size, $R_{\max}$, which designates the maximum size of each generated community. In the experiments, n ranges from 1000 to 10000 according to the interval of 1000, and k is 20, $k_{\max}$ is 50, λ is 0.6, λ_1 is -2, λ_2 is -1, $R_{\min}$ is 10, and $R_{\max}$ is 50, all of which are typical suggested values [81]. Fig. 5 and Fig. 6 show the average error and their

variation rate along with the number of clusters using Eqs. (19) and (20) when n is 1000. Other parameters are the same as description previously. We deduced that there are 12 communities in the networks. Fig. 7 shows the average error as n changes from 1000 to 10000 in intervals of 1000 when there are 12 communities in the networks. The result shows that the method can address different sizes of networks. The results from Eq. (20) of the proposed method are more accurate than are those from Eq. (19). Because of the performance limitations of our computer, how to test the proposed method on significantly larger networks is our future work. Sparse subspace representation theory and fast solution of a linear matrix equation provide useful tools and directions for community detection.

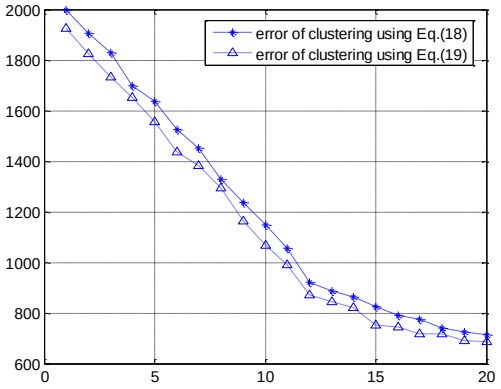


Fig. 5 Average error versus the number of clustering using Eq.(19) and (20) on LFR benchmark

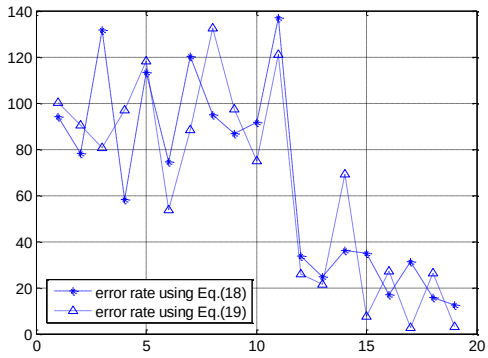


Fig. 6 Average error variation rate versus the number of clustering using Eq.(19) and (20) on LFR benchmark

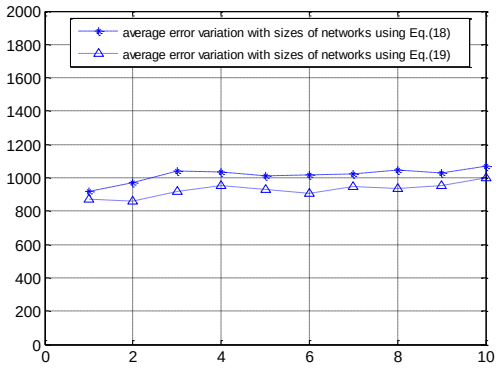


Fig. 7 Average error along with different network sizes

Finally, the time costs of two versions of the proposed method are compared with recently proposed similar method [74]. There are two versions marked as geodesic sparse subspace communities (GSSC) and sparse subspace communities with fusion (SSCF) in the reference method. Table 4 shows that the

time cost of the proposed method is superior to those of reference method because there is no kernel function mapping or nonlinear geometry distance computation in the proposed method, which is necessary in the reference. The experiment condition is as same as the previous when the number of nodes in the network n is 1000 for the LFR dataset.

Table 4 Time cost of the proposed method compared with similar method

Evaluation	GSSC of ref. [19]	SSCF of ref. [19]	Eq. (19)	Eq. (20)
Karate	4.41	4.67	4.06	**4.23**
Dolphin	6.14	6.85	5.48	**5.64**
Football	3.36	3.64	2.95	**3.18**
Politics	5.73	6.29	5.25	**5.59**
Email	89.47	95.16	82.92	**86.74**
Net science	103.52	118.73	94.96	**99.81**
GN	6.50	6.726	5.88	**6.02**
LFR	98.14	105.83	89.30	**92.76**

4.5 Convergence comparing

As previously discussed, key steps in the proposed method are to solve the coefficients matrix of the optimization program Eq.(19) or Eq.(20) and to apply spectral clustering to the similarity matrix. The latter has standard processes. We mainly verified that AMDD frameworks used in Eq.(19) or Eq.(20) is convergent, and the mixed-norm SSC model Eq.(20) is more accurate than the noise-free SSC model Eq.(19) in community detection. In fact, it is found that the ADMM framework can converge to modest accuracy within dozens of iterations even though it is slow to converge to high accuracy [69]. This property of ADMM can deal with community detection challenges with huge datasets. From experiments, we see that this level of accuracy is sufficient enough for community detection. Fig. 8 shows the values of the objective functions of Eq.(19) and Eq.(20) along with the number of iteration of the ADMM framework for the Zachary Karate Club dataset. From table 2, table 3, Fig. 4, Fig. 6 and Fig. 7, we see that average errors of clustering using Eq.(20) is smaller than those of using Eq.(19). From Fig. 8, we noticed that solutions of the mixed-norm SSC model Eq.(20) converge faster than those of the error-free SSC model Eq.(19). After 20 iterations, the values of the objective functions of Eq.(20) is stable between 0.723 ± 0.054. While after 31 iterations, the values of the objective functions of Eq.(19) is stable between 0.786 ± 0.052. Compared with the noise-free SSC model, the mixed-norm SSC model is more accurate for community detection because the connections of nodes between different communities in networks always exist and are modeled as noises suitably.

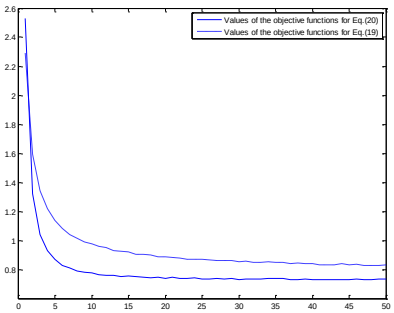


Fig. 8 Values of the objective functions of Eq.(19) and Eq.(20) versus iteration

5 Conclusions

Community detection is a challenging problem in various research fields such as social networks, biology

gene expressions, physics systems. A new community detection method based on the theory of SSC with mixed-norm constrains is proposed in this paper. By using the approach of sparse subspace representation theory, each community in a given network can be considered as a subspace in some similarity measure space. Moreover, each node can be represented by a linear combination of the other nodes in the same subspace. The connections of nodes between different communities are modeled as noises to improve the clustering accuracy. By introducing mixed-norm constraint condition, the representation coefficients of each node in the subspace are formulated. The ADMM framework is used to solve the formulation. Finally, the proposed community detection method is compared with the state-of-the-art algorithms on both labeled and unlabeled real benchmark networks and on synthetic networks. Experimental results show that the proposed method is effective. The self-representation ability of SSC provides a kind of suitable description of the community in the network. The proposed method gives a new way of addressing community detection challenges in different networks with huge datasets.

Acknowledgments

This research is supported by the National Natural Science Fund of China (71302080), the Shanghai Natural Science Fund (12ZR1409900), the Ministry of Education Research of Social Sciences Youth funded projects (13YJC630149) and the Special Research Fund for the Doctoral Program of Higher Education of China (20120078120001).

References

[1] BV. Pareto. Manual of political economy. New York: A. M. Kelly, 1971.

[2] Fortunato, Santo. Community detection in graphs. Physics, 486(3-5) (2010) 75-174.

[3] M Girvan , MEJ Newman. Community structure in social and biological networks. Proceedings of the National Academy of Sciences of the United States of America, 99(12) (2002) 7821-7826.

[4] MEJ Newman, M Girvan. Finding and evaluating community structure in networks. Physical Review E Statistical Nonlinear & Soft Matter Physics, 69(2 Pt 2) (2004) 026113-026113.

[5] A. Lancichinetti, S. Fortunato. Community detection algorithms: a comparative analysis. Physical Review E Statistical Nonlinear & Soft Matter Physics, 80(5 Pt 2) (2009) 2142-2152.

[6] Weiss, R. S., and E. Jacobson, A method for the analysis of the structure of complex organizations, Am. Sociol. Rev. 20 (1955) 661-668.

[7] Mark Newman. Detecting community structure in networks. Eur. Phys. 38(2) (2004) 321-330.

[8] A. Clauset, M E J. Newman, C. Moore. Finding community structure in very large networks. Physical Review E. 70(6 Pt 2) (2010) 264-277.

[9] Hans-Peter Kriegel, Peer Kröger, Arthur Zimek. Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. ACM Transactions on Knowledge Discovery from Data. 3(1): (2009) 337-348.

[10] E. Elhamifar, R. Vidal. Sparse subspace clustering: Algorithm, theory, and applications. IEEE Transactions on Pattern Analysis & Machine Intelligence. 35(11) (2012) 2765-81.

[11] L. Parsons, E. Haque, H. Liu. Subspace clustering for high dimensional data: A review. Acm Sigkdd Explorations Newsletter. 6(1) (2004) 90-105.

[12] Mats Julian Olsen. Community detection in large social networks. Institutt for Matematiske Fag, 2014.

[13] D. Hric, R. K. Darst, S. Fortunato. Community detection in networks: Structural communities versus ground truth. Physical Review E Statistical Nonlinear & Soft Matter Physics. 90(6) (2014) 062805-062805.

[14] Amit Dhumal, Pravin Kamde. Survey on community detection in online social networks. International Journal of Computer Applications. 121(9) (2015) 0975-8887.

[15] M. Newman. Networks: an Introduction. Astronomische Nachrichten. 327(8) (2010) 741-743.

[16] X Xu, N Yuruk, Z Feng, T A J Schweiger. SCAN: a structural clustering algorithm for networks. Acm Sigkdd International Conference on Knowledge Discovery & Data Mining, 2007: 824-833.

[17] Pan G, Zhang W, Wu Z, Li S. Online Community Detection for Large Complex Networks. PLoS ONE. 9(7) (2014) e102799. doi:10.1371/journal.pone. 0102799

[18] J. M. Kumpula, M. Kivel¨a, K. Kaski, J. Saram¨aki. Sequential algorithm for fast clique percolation. Physical Review E Statistical Nonlinear & Soft Matter Physics. 78(2) (2008) 1815-1824.

[19] G. Karypis, V. Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. Siam Journal on Scientific Computing. 20(1) (2006) 359--392.

[20] Wu P, Pan L. Multi-Objective community detection based on matrix algorithm. PLoS ONE 10(5) (2015) e0126845. doi:10.1371/journal. pone.0126845.

[21] S White, P Smyth. A spectral clustering approach to finding communities in graphs. In: Proceedings of the Fifth SIAM International Conference on Data Mining, vol. 119, p. 274 (2005)

[22] R. Andersen, F. Chung, K. Lang. Local graph partitioning using pagerank vectors. Annual Symposium on Foundations of Computer Science (2006) 475-486.

[23] J. He, J. E. Hopcroft, H. Liang, S. Supasorn, L. Wang. Detecting the structure of social networks using (α, β) -communities. In Proc. 8th Workshop on Algorithms and Models for the Web Graph (WAW), 2011.

[24] C. X. Lin, B. Zhao, Q. Mei, J. Han. Pet: a statistical model for popular events tracking in social communities. Siam International Conference on Data Mining (2010) 929-938.

[25] D. Deritei, Z. I. Lazar, I. Papp, F. Jarai-Szabo, R. Sumi, L. Varga, E.R.Regan, and M. Ercsey Ravasz, Community detection by graph Voronoi diagrams, New Journal of Physics, 16(6) (2014) 63007-63022.

[26] P. De Meo, E. Ferrara, G. Fiumara, A. Provetti. Mixing local and global information for community detection in large networks. Journal of Computer & System Sciences, 80(1) (2014) 72-87.

[27] Nathan Aston, Wei Hu. Community detection in dynamic social networks. Communications and Network, 06(02) (2014) 124-136.

[28] C.D. Wang, J.-H. Lai, P. Yu. Neiwalk: Community discovery in dynamic Content-based networks. IEEE Transactions on Knowledge & Data Engineering, 26(7) (2013) 1734-1748.

[29] Federico Ricci-Tersenghi, Adel Javanmard, Andrea Montanari. Performance of a community detection algorithm based on semi-definite programming. Journal of Physics Conference Series, 699(1) (2016) 012015.

[30] Y Xin，Z Q Xie，J Yang. An adaptive random walk sampling method on dynamic community detection. Expert Systems With Applications, 58(1) (2016) 10-19.

[31] L. Tang, H. Liu, J. Zhang. Identifying evolving groups in dynamic multimode networks, IEEE Transactions on Knowledge & Data Engineering, 24(1) (2011) 72-85.

[32] Kwan Hui Lim, Amitava Datta. A topological approach for detecting twitter communities with common interests. Springer, 2012.

[33] Palsetiay et al. User-interest based community extraction in social networks,SNAKDD'12, ACM 2012.

[34] Denzil Correa et al., iTop: interaction based topic centric community discovery on twitter, PIKM12, P 51-58,ACM 2012.

[35] Anh Tang, Emmanuel Viennet. Community Detection based on Structural and Attribute Similarities. Achi, 2015:7-12.

[36] N Natarajan，P Sen，V Chaoji. Community detection in content-sharing social networks. IEEE/ACM International Conference on Advances in Social Networks Analysis & Mining, 2013:82-89.

[37] SMV Dongen. Graph Clustering by flow Simulation. Ph.D. dissertation, Dutch Nat. Res. Inst. Math. Comput. Sci., Univ. of Utrecht, Netherlands, JE Utrecht, Netherlands, 2000.

[38] D L Donoho. Compressed sensing. IEEE Transactions on Information Theory, 52(4) (2006) 1289-1306.

[39] M A Davenport，M F Duarte，Y C Eldar，G Kutyniok. Introduction to compressed sensing. Lecture Note, 2012, 93.

[40] Ren, Vidal. Subspace Clustering. IEEE Signal Processing Magazine, 2011, 28(2):52-68.

[41] M Elad. Sparse and redundant representations: from theory to applications in signal and image processing. Springer Publishing Company,Incorporated, 2010, 02(1):1094-1097.

[42] R Vidal, P Favaro. Low rank subspace clustering (LRSC). Pattern Recognition Letters, 2014, 43(7):47-61.

[43] J P Costeira, T Kanade. A multibody factorization method for independently moving objects. International Journal of Computer Vision, 1998, 29(3):159-179.

[44] R. Vidal, Y. Ma, S. Sastry. Generalized principal component analysis (GPCA). IEEE Trans. Pattern Anal. Machine Intell., 2005,27(12):1-15.

[45] T. Zhang, A. Szlam, G. Lerman. Median k-flats for hybrid linear modeling with many outliers. IEEE International Conference on Computer Vision Workshops, 2009, 122(3):361-367.

[46] Y. Ma, H. Derksen, W. Hong, J. Wright. Segmentation of multivariate mixed data via lossy coding and compression. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2007, 29(9):1546-62.

[47] G. Liu, Z. Lin, Y. Yu. Robust subspace segmentation by low-rank representation. International Conference on Machine Learning, 2010:663-670.

[48] G. Chen ,G. Lerman. Spectral curvature clustering (SCC).International Journal of Computer Vision, 2009, 81(3):317-330.

[49] R. Vidal, Y. Ma, S. Sastry. Generalized principal component analysis (GPCA). IEEE Trans. Pattern Anal. Machine Intell., 2005,27(12):1-15.

[50] P. Agarwal, N. Mustafa. K-means projective clustering. Acm Sigact-sigmod-sigart Symposium on Principles of Database Systems, 2004, 23:155-165.

[51] L. Lu, R. Vidal. Combined central and subspace clustering on computer vision applications. International Conference, 2006:593--600.

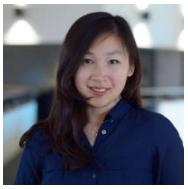
[52] J. Yan, M. Pollefeys. A general framework for motion segmentation: Independent, articulated, rigid, non-rigid, degenerate and non-degenerate. European Conference on Computer Vision, 2006, 3954:94--106.

[53] A. Goh, R. Vidal. Segmenting motions of different types by unsupervised manifold clustering. IEEE Conference on Computer Vision & Pattern Recognition, 2007:1-6.

[54] D. L. Donoho. For most large underdetermined systems of linear equations the minimal 1-norm solution is also the sparsest solution. Communications on Pure & Applied Mathematics, 2015, 59(6):797-829.
[55] E. Candes, X. Li, Y. Ma, J. Wright. Robust principal component analysis. Journal of the Acm, 2011, 58(3):1-73.
[56] B. Recht, M. Fazel, P. Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. Siam Review, 2010, 52(3):471-501.
[57] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, Y. Ma. Robust recovery of subspace structures by low-rank representation. Pattern Analysis & Machine Intelligence IEEE Transactions on, 2013, 35(1):171-184.
[58] P. Favaro, R. Vidal, A. Ravichandran. A closed form solution to robust subspace estimation and clustering. IEEE Conference on Computer Vision & Pattern Recognition. 2011, 42(7):1801-1807.
[59]M. Soltanolkotabi, E. J. Candes. A geometric analysis of subspace clustering with outliers. Annals of Statistics, 2012,40(4): 2195-2238.
[60] Abavisani, Mahdi; Patel, Vishal M. ,Multimodal sparse and low-rank subspace clustering, Information Fusion, 2018, 39(1): 168-177.
[61] Liu, Xiaolan; Yi, Miao; Han, Le; Deng, Xue, A subspace clustering algorithm based on simultaneously sparse and low-rank representation, Journal of Intelligent and Fuzzy Systems, 2017, 33(1): 621-633.
[62] Fan, Jicong; Chow, Tommy W.S. ,Sparse subspace clustering for data with missing entries and high-rank matrix completion, Neural Networks, 2017, 93(1): 36-44.
[63] Wang, Jun; Shi, Daming; Cheng, Dansong; Zhang, Yongqiang; Gao, Junbin, LRSR: Low-Rank-Sparse representation for subspace clustering, Neurocomputing, 2016, 214: 1026-1037.
[64] Fu, Yifan; Gao, Junbin; Tien, David; Lin, Zhouchen; Hong, Xia, Tensor LRR and Sparse Coding-Based Subspace Clustering, IEEE Transactions on Neural Networks and Learning Systems, 2016, 27(10): 2120-2133.
[65] Liu, Bo; Yuan, Xiao-Tong; Yu, Yang; Liu, Qingshan; Metaxas, Dimitris N., Parallel sparse subspace clustering via joint sample and parameter blockwise partition, ACM Transactions on Embedded Computing Systems, 2017, 16(3-75):1: 17.
[66] CG Li, C You, R Vidal. Structured sparse subspace clustering: a joint affinity learning and subspace clustering framework. IEEE Transactions on Image Processing, 2017, 26(6): 2988-3001.
[67] Yong Li, Wenrui Dai, J. Zou, Hongkai Xiong, and Y. F. Zheng, Structured Sparse Representation with Union of Data-driven Linear and Multi-Linear Subspaces Model for Compressive Video Sampling, IEEE Transactions on Signal Processing, 2017, 65, 19: 5062-5077.
[68] T. Lu, Z. Xiong, Y. Zhang, B. Wang, and T. Lu, Robust face super-resolution via locality-constrained low-rank representation, IEEE Access: Special Section on Advanced Data Analytics for Large-scale Complex Data environments, 2017, 5 : 13103-13117.
[69] Xin Zhang, Duc-Son Pham, Svetha Venkatesh, Wanquan Liu, Dinh Phung. Mixed-norm sparse representation for multi-view face recognition. Pattern Recognition, 2015, 48(9):2935-2946.
[70] J Xu， K Xu， K Chen, J Ruan. Reweighted sparse subspce clustering. Computer Vision & Image Understanding, 2015, 138: 25-37.
[71] Wanjun Chen, Erhu Zhang, Zhuomin Zhang. A Laplacian structured representation model in subspace clustering for enhanced motion segmentation. Neurocomputing, 2016, 208: 174-182.
[72] Gene expresion data clustering based on graph regularized subaspace segmentation. Neurocomputing, 2014 ,143(16):44-50.
[73] L. Donetti, M. A. Munoz. Detecting network communities: A new systematic and efficient algorithm. Journal of Statistical Mechanics Theory & Experiment, 2004(10):10012.
[74] Arif Mahmood, Michael Small. Subspace based network community detection using sparse linear coding. IEEE Transactions on Knowledge & Data Engineering, 2016:28(3): 801-812.
[75] GH Golub , CF Van Loan. Matrix computations (3rd ed.) Johns Hopkins University Press, 1996.
[76] Yiju Wang, Wanquan Liu, Louis Caccetta, Guanglu Zhou. Parameter selection for nonnegative l-1 matrix/tensor sparse decomposition. Operations Research Letters, 2015, 43(4): 423-426.
[77] S. Boyd, N. Parikh, E. Chu, B. Peleato and J. Eckstein, Distributed optimization and statistical learning via the alternating direction method of multipliers, Foundations and Trends in Machine Learning 3(1) (2010), 1–122.
[78] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2000, 22(8): 888-905.
[79] J. Mairal, F. Bach, J. Ponce, G. Sapiro. Online dictionary learning for sparse coding. International Conference on Machine Learning, 2009:689-696.
[80] B. Frey, D. Dueck. Clustering by passing messages between data points. Science, 2007, 315(5814): 972-976.
[81] A. Lancichinetti, S. Fortunato, and F. Radicchi. Benchmark graphs for testing community detection algorithms. Phys. Rev. E, 2008, 78(4): 046110.
[82] Lusseau D, Schneider K, Boisseau O J, et al. The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations. Behav Ecol Sociobiol, 2003, 54: 396-405.
[83] Guimera R, Danon L, Diaz-Guilera A, et al. Self-similar community structure in a network of human interactions. Phys Rev E, 2003, 68: 065103.

[84] Wang G X, Shen Y. Modularity matrix of networks and the measure of community structure. Acta Phys Sin, 2010, 59: 842–850.

Bo Tian, received her Ph.D. degree and M.S. degree in management science and engineering from the school of management, Xi'an Jiaotong University, Xi'an, P. R. China in 2009 and 2005 respectively. She visited Henley Business School, University of Reading from Feb, 2015 to 2016. She is currently an associate professor in the School of Information Management & Engineering, Shanghai University of Finance and Economics. And she is a master student supervisor in management science and information systems. Her main research interests include social computation, e-commerce, information systems, business intelligence and data mining.



Weizi Li, received her Ph.D. degree in management science and engineering from school of management and economics, Beijing Institute of Technology, China in 2010. She stated to work in Informatics Research Centre, Henley Business School, University of Reading since 2012. She is currently an associate professor of informatics at Henley Business School and a Fellow of Charted Institute of IT (FBCS). Her research focuses on information systems, data analytics and artificial intelligence especially in healthcare area.