Signal Processing 220 (2024) 109436

Contents lists available at ScienceDirect

Signal Processing

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

A game model for semi-supervised subspace clustering with dynamic affinity and label learning

Tingting Qi, Xiangchu Feng [*], Weiwei Wang

School of Mathematics and Statistics, Xidian University, Xi'an 710071, China

ARTICLE INFO

Keywords:
Semi-supervised learning
Subspace clustering
Game theory
Graph convolutional network

ABSTRACT

With the aid of partial supervised information, semi-supervised subspace clustering methods aim to obtain affinity matrices directly derived from raw data, and then those affinity matrices are utilized to get assignment matrices. These affinity matrices are susceptible to disturbances such as noise and outliers, which could significantly impact their quality. To mitigate this, it becomes essential to dynamically update the affinity matrix using the clustering result, reducing the high dependency on raw data. This paper presents a Game Model for Semi-supervised Subspace Clustering (GMSSC). The first submodule of GMSSC utilizes a Graph Convolutional Network to learn a mapping from raw data to the assignment matrix with the assistance of the affinity matrix. The second submodule constructs a semi-supervised self-expressive model to learn a discriminative affinity matrix. By integrating these submodules into a game model, GMSSC achieves a Nash Equilibrium during adversarial training, resulting in stable and robust affinity and assignment matrices. Additionally, we propose an iterative algorithm based on data-driven and model-driven approaches to solve the model. Experimental results on four open real-world datasets demonstrate that the proposed method not only achieves elegant results but also outperforms state-of-the-art semi-supervised subspace clustering algorithms.

1. Introduction

High-dimensional data is prevalent in many real-world domains. As an unsupervised learning method, clustering plays a crucial role in the processing of such data. Practical applications often yield a scarcity of labeled information for data samples due to inefficient manual annotation, leading to the emergence of semi-supervised learning [1–16]. This approach guides machine learning models to predict labels for all data with minimal supervisory information. In contrast to unsupervised learning, semi-supervised learning achieves more accurate and stable results with only a small amount of labeled data and thus is widely applied in various fields. A notable and efficient technique within semi-supervised clustering methods is the graph-based semi-supervised subspace clustering [17–27].

Graph-based semi-supervised subspace clustering considers data samples as nodes within a graph, where the edge weights represent the similarity between samples. This constructed graph unveils the underlying geometric structure of the data, thereby affecting the performance of subspace clustering methods. The quality of the affinity matrix, constituting the constructed graph, significantly influences the efficacy of these methods. In subspace clustering, it is assumed that high-dimensional data resides within the union of multiple low-dimensional subspaces, each associated with a distinct category or class. The concept of self-expressiveness signifies that a sample can be represented linearly or affinely using other samples from the same subspace. Encoded such property, the self-expressive matrix can be utilized to construct the affinity matrix directly, as seen in traditional algorithms such as Sparse Subspace Clustering (SSC) [28], Low-Rank Representation (LRR) [29], and Block Diagonal Representation (BDR) [30]. Typically, the self-expressive matrix functions as the affinity matrix when it is nonnegative [22]. For ease of reference, we will consistently denote the self-expressive matrix as the affinity matrix. Graph-based semi-supervised subspace clustering entails two modes of learning for the affinity matrix: static learning and dynamic updating.

Static learning of a graph utilizes the raw data to build model directly and learns the graph without additional adjusting to the affinity matrix based on clustering results. These methods, categorized by the type of supervision, are typically divided into two groups: methods based on Pairwise Constraints (PC) [17–20] and those relying on Gaussian Fields and Harmonic Functions (GFHF) [21–27].

PC-based methods utilize the set of a little class attributes of the raw data as weak supervision to guide the subspace clustering. These PC-based techniques establish two sets of edges based on sample class membership: must-links (MLs) and cannot-links (CLs). MLs consist of edges connecting samples from the same class, while CLs encompass

[*] Corresponding author.
E-mail addresses: ttqi@stu.xidian.edu.cn (T. Qi), xcfeng@mail.xidian.edu.cn (X. Feng), wwwang@mail.xidian.edu.cn (W. Wang).

https://doi.org/10.1016/j.sigpro.2024.109436
Received 13 December 2023; Received in revised form 13 February 2024; Accepted 15 February 2024
Available online 24 February 2024

edges between samples from different classes. Typically, these methods assign weights to the edges in MLs and CLs, with different definitions across various approaches. For instance, SSLRR [17] sets the weight of edges in CLs to 0 and enforces this constraint on the self-expressive matrix in LRR. DPLRR [18] assigns a weight of 1 to edges in CLs, constructing a disaffinity matrix D. It augments the affinity matrix learning in LRR by integrating adversarial learning between similarity and dissimilarity matrices. While these methods mainly emphasize CLs, SSC_TLRR [19] constructs a matrix considering both MLs and CLs. It assigns weights of −1 to the edges in CLs and 1 to the edges of MLs. Within the LRR framework, SSC_TLRR stacks this matrix with the affinity matrix into a 3D tensor and learns an optimal affinity matrix by applying low-rank constraints to this tensor.

GFHF-based methods incorporate partially known labels as prior information into the model directly, primarily employing the Manifold Smoothing Regularity term (MSR) and label fitting term to infer the remaining unknown labels [27]. The MSR term delineates the local relationship between the assignment matrix and the affinity matrix. Meanwhile, the label fitting term ensures consistency between the predicted labels and the known labels for labeled samples. Typically, such semi-supervised subspace clustering methods integrate these terms with traditional self-expressive models for subspace clustering. For example, STSSL [21] integrates the MSR term into the conventional self-expressive model and iteratively updates the affinity matrix and predicted labels. NNLRR [22] unifies LRR and GFHF into a single model, facilitating concurrent learning of the assignment matrix and the affinity matrix. Building upon NNLRR, DCSSC [23] introduces a novel regularity that strengthens the local relationship between the assignment matrix and the affinity matrix. However, the raw data are often affected by noise, outliers, and other interferences, which leads to an inaccurate graph constructed by the raw data directly, thus affecting the final clustering results [31–35].

To tackle the aforementioned challenges, several researchers have proposed dynamic graph learning strategies, wherein the affinity matrix is adjusted dynamically based on the learned assignment matrix. For instance, [35] emphasized that it is necessary to construct an adaptive graph to handle specific features in hyperspectral images. Additionally, [31] suggested learning an adaptive graph to diminish the sensitivity of clustering results to the initial graph. Moreover, [33] utilized PC constraints to update the affinity matrix dynamically during the iterative process. These methodologies have showcased promising experimental outcomes. However, most of these approaches follow a two-stage process: [31,35] utilize the learned affinity matrix for spectral clustering, while [33] leverages the acquired low-dimensional features for K-means to obtain the final clustering assignments. Undoubtedly, these methodologies present a new perspective for enhancing the quality of the affinity matrix.

From the perspective of updating graph dynamically, this paper proposes a game-based semi-supervised subspace clustering model that synchronously learns the assignment matrix and the affinity matrix. Leveraging both the global geometric relationship and the local connections between the assignment matrix and the affinity matrix, we construct a game-based model comprising two primary submodules. To streamline the assignment matrix acquisition process and avoid the two-stage approach, the first submodule utilizes Graph Convolutional Network (GCN) [36] to achieve an end-to-end learning of the assignment matrix label self-expressiveness property. Continuously updating the affinity matrix based on the modified assignment matrix, this submodule ensures an iterative enhancement of the affinity matrix. Throughout the game training process, the assignment matrix and the affinity matrix cooperate and update each other dynamically, enhancing their quality until the model reaches a Nash Equilibrium, thereby attaining stable assignment and affinity matrices. Our primary contributions can be summarized as follows:

- Two submodules aimed at learning the assignment matrix and the affinity matrix based on label self-expressiveness and manifold smoothing regularity are constructed.
- A Game Model for Simultaneous Learning of the Assignment Matrix and Affinity Matrix is proposed.
- An Iterative Algorithm Combining Model-driven and Data-driven Approaches is proposed.

The structure of this paper is outlined as follows: Section 1 comprises the Introduction. Section 2, which delves into Related Work, encompasses various topics, including data self-expressiveness, label self-expressiveness, semi-supervised subspace clustering, graph convolutional networks, game models and Nash Equilibrium. Section 3 provides a detailed explanation of our game model, initiating with the Label Self-expressiveness based GFHF module and the Discriminative Affinity Learning module. These submodules are subsequently amalgamated to create a comprehensive game model. This section concludes by presenting the algorithm designed for resolving the proposed model. Section 4 shows the experimental results obtained from our method on four benchmark datasets. Lastly, Section 5 serves as a summary of the work detailed in this paper.

2. Related work

Given the raw data $X \in \mathbb{R}^{d\times n}$, where d represents the dimensionality of each sample and n denotes the number of samples. It originates from the union of k subspaces $\bigcup_{l=1}^{k} S_l$, where S_l represents any of these subspaces. $X = [x_1, \dots, x_u, x_{u+1}, \dots, x_n]$, $x_1, \dots, x_u$ represent the samples with known labels, and the labels for $x_{u+1}, \dots, x_n$ are unknown. We denote the known assignment matrix as $Y = \left[y_1, \dots, y_u, y_{u+1}, \dots, y_n\right] \in \mathbb{R}^{k\times n}$, where $y_1, \dots, y_u$ correspond to the known labels. Each $y_i(i \in 1, \dots, u)$ is a column vector of dimensionality k, where only one element is 1, and the remaining elements are 0. When $y_{li} = 1$, it indicates that x_i belongs to the lth subspace. The unknown $y_{u+1}, \dots, y_n$ are entirely zero vectors. We denote the predicted assignment matrix as $Q \in \mathbb{R}^{k\times n}$ and the self-expressive matrix as $C \in \mathbb{R}^{n\times n}$.

2.1. Data self-expressiveness and label self-expressiveness

Traditional unsupervised subspace clustering methods [28–30,37], e.g., SSC, LRR, BDR, typically employ the raw data for modeling, and their optimization problems can be uniformly formulated as:

$$\begin{aligned} &\min_{C} \mathcal{L}(E) + \mathcal{R}(C) \\ &s.t.\ X = XC + E \end{aligned} \tag{1}$$

In this equation, $\mathcal{L}(\cdot)$ quantifies the reconstructed loss, measuring the self-expressive error of the raw data, and $\mathcal{R}(\cdot)$ is the regularization term of C. Traditional unsupervised subspace clustering methods construct an affinity matrix $A = \frac{1}{2}(|C| + |C^T|)$ upon the affinity matrix obtained from the optimization problem (1). Subsequently, they utilize spectral clustering methods such as N-Cut [38] to derive the final assignment matrix Q from the affinity matrix. From the aforementioned spectral clustering process, it is evident that the quality of the affinity matrix significantly influences the ultimate clustering results. Over the past twenty years, traditional unsupervised subspace clustering models have primarily differed in their selection of $\mathcal{L}(\cdot)$ and $\mathcal{R}(\cdot)$, aiming to identify suitable $\mathcal{L}(\cdot)$ and $\mathcal{R}(\cdot)$ to acquire C with high-quality.

Specifically, BDR introduces a block-diagonal regularization and demonstrates that the affinity matrix satisfying the conditions mentioned above also complies with the block-diagonal property. Consequently, the optimized affinity matrix obtained from (1) has the following form:

$$C^* = \begin{bmatrix} C_1^* & 0 & \cdots & 0 \\ 0 & C_2^* & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & C_k^* \end{bmatrix} \tag{2}$$

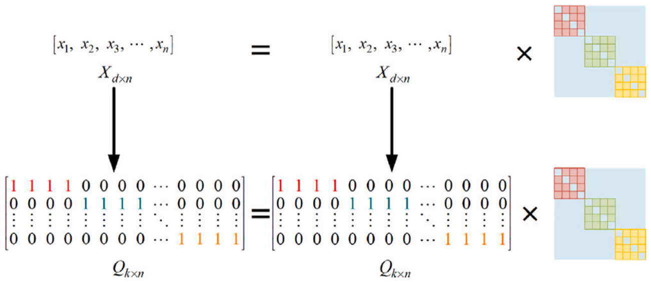


Fig. 1. The illustration of Label self-expressiveness. The corresponding assignment matrix should have the same self-expressive property (the lower part) when the raw data has a self-expressive property (the upper part). The self-expressive matrices are the same.

where $C_l^* \in \mathbb{R}^{n_l\times n_l}$ represents the affinity matrix corresponding to samples in the lth subspace. To address this, BDR proposes an intuitive k-block diagonal regularization as follows:

$$\|C\|_{\boxed{k}} = \sum_{i=n-k+1}^{n} \lambda_i(L_C) \tag{3}$$

L_C refers to the Laplacian matrix of C, $\|C\|_{\boxed{k}}$ is defined as the sum of the k smallest eigenvalues of L_C. BDR formulates the optimization problem for solving C as follows:

$$\begin{aligned}&\min_C \frac{1}{2}\|X - XC\|_F^2 + \gamma\|C\|_{\boxed{k}}\\ &s.t.\ diag(C) = 0, C \geq 0, C = C^T\end{aligned} \tag{4}$$

However, these methods are modeled directly based on the raw data, which makes them sensitive to noise, outliers, and other disturbances, which leads to an inaccurate affinity matrix. Hence, numerous algorithms have emerged, aiming to learn intrinsic features from raw data and establish self-expressive models within the feature domain. These include methods based on kernel functions [39–41] and deep learning approaches [42–44]. Such methods achieve self-expressive matrices with high quality by ensuring that the associated features satisfy the self-expressive property, thereby reducing the impact of noise and other disturbances. Specifically, each sample corresponds to a label which can be considered as a special feature for the sample.

As mentioned by [45,46], for most cases in the real-world applications, there exists $D \in \mathbb{R}^{d\times k}, b \in \mathbb{R}^{k\times 1}$, and the assignment matrix Q can be expressed as a linear combination of the raw data:

$$Q = D^T X + b\mathbf{1}_n^T \tag{5}$$

When the raw data exhibits self-expressive property as $X = XC$, and the affinity matrix satisfies $C^T\mathbf{1}_n = \mathbf{1}_n$, where $\mathbf{1}_n$ is a column vector with full 1. This constraint implies that each sample can be affinely represented by other samples from the same subspace. At this point, we have

$$Q = D^T X + b\mathbf{1}_n^T = \begin{bmatrix} D^T & b \end{bmatrix}\begin{bmatrix} X \\ \mathbf{1}_n^T \end{bmatrix} = \begin{bmatrix} D^T & b \end{bmatrix}\begin{bmatrix} XC \\ \mathbf{1}_n^T C \end{bmatrix} = \begin{bmatrix} D^T & b \end{bmatrix}\begin{bmatrix} X \\ \mathbf{1}_n^T \end{bmatrix} C = QC \tag{6}$$

In other words, the assignment matrix Q also exhibits self-expressive property [9]:

$$Q = QC \tag{7}$$

From Eq. (7), it can be explicitly observed that it requires the label of a sample to be represented affinely by labels same to it. This establishes a global geometric relationship between the assignment matrix and the affinity matrix (see Fig. 1).

2.2. Semi-supervised subspace clustering

The semi-supervised subspace clustering based on GFHF typically incorporates a small number of labels into the model directly to enhance the learning of the affinity matrix. GFHF can be described as follows:

$$\begin{aligned}&\min_Q \frac{1}{2}\sum_{i,j=1}^{n}\|q_i - q_j\|_F^2 C_{ij} + \lambda_\infty \sum_{i=1}^{u}\|q_i - y_i\|_F^2\\ &s.t.\ Q \in \mathcal{Y},\ \mathcal{Y} = \{Q \in (0,1)^{k\times n} : Q^T\mathbf{1_n} = \mathbf{1_n},\ rank(Q) = k\}\end{aligned} \tag{8}$$

The first term is referred to as MSR, which represents the local relationship between the assignment matrix and the affinity matrix. The second term is known as the label-fitness term, which enforced q_i to approach y_i with a large λ_∞. The above equation can be simplified as:

$$\begin{aligned}&\min_Q \frac{1}{2}Tr(QL_CQ^T) + Tr((Q-Y)U(Q-Y)^T)\\ &s.t.\ Q \in \mathcal{Y}, \mathcal{Y} = \{Q \in \{0,1\}^{k\times n} : Q^T\mathbf{1}_n = \mathbf{1}_n, rank(Q) = k\}\end{aligned} \tag{9}$$

$U \in \mathbb{R}^{n\times n}$ is a diagonal matrix with diagonal elements given by:

$$U_{ii} = \begin{cases}\lambda_\infty, & i = 1,\dots,u\\ 0, & i = u+1,\dots,n\end{cases} \tag{10}$$

The semi-supervised subspace clustering based on GFHF often can be formulated as a joint optimization problem involving C and Q:

$$\begin{aligned}&\min_{C,E,Q} \frac{1}{2}Tr(QL_CQ^T) + Tr((Q-Y)U(Q-Y)^T) + \gamma\mathcal{R}(C) + \frac{\beta}{2}\mathcal{L}(E)\\ &\text{s.t. } X = XC + E, C \in \Omega\end{aligned} \tag{11}$$

where Ω represents the constraints imposed on C by different methods. The selection of $\mathcal{L}(\cdot)$ and $\mathcal{R}(\cdot)$ is similar to (1).

Indeed, this kind of model has achieved outstanding performance in semi-supervised subspace clustering by directly utilizing limited label information to guide the learning of the affinity matrix. However, these methods rely on modeling with the raw data, and the high dimensionality of the raw data leads to high computational complexity. Additionally, similar to unsupervised subspace clustering, the influence of original noise and other interferences will be transmitted to the predicted labels through the affinity matrix.

2.3. Graph convolutional network

In recent years, GCN have been widely applied to classification and clustering tasks. Similar to Convolutional Neural Network (CNN), the forward propagation of GCN can be represented as:

$$H^{(l+1)} = \sigma(\Theta^{(l)}H^{(l)}\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}) \tag{12}$$

Here, $H^{(l+1)}$ represents the output of the lth layer, $\sigma(\cdot)$ denotes the activation function, $H^{(l)}$ stands for the input of the lth layer, $\Theta^{(l)}$ denotes the weights in the lth layer, $\tilde{A} = A + I_n$ is the sum of the initialized affinity matrix of an undirected graph and its self-connections, and I_n is the identity matrix. $\tilde{D}$ is the degree matrix constructed from $\tilde{A}$, specifically a diagonal matrix:

$$\tilde{D}_{ii} = \sum_{j=1}^{n}\tilde{A}_{ij} \tag{13}$$

However, many studies have discovered that multi-layer GCN tend to converge all nodes to identical values, known as the over-smoothing effect [15]. Therefore, considering the balance between computational complexity and accuracy, we utilize a two-layer GCN here to establish a parameterized representation of the assignment matrix(see Fig. 2), denoted as:

$$Q = G(X, A) = \mathrm{softmax}(\Theta^{(1)}\mathrm{Relu}(\Theta^{(0)}X\bar{A})) \tag{14}$$

with $\bar{A} = \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}$. $\Theta^{(0)} \in \mathbb{R}^{h\times d}, \Theta^{(1)} \in \mathbb{R}^{k\times h}$ represent the network parameters of GCN, h is the number of hidden units.

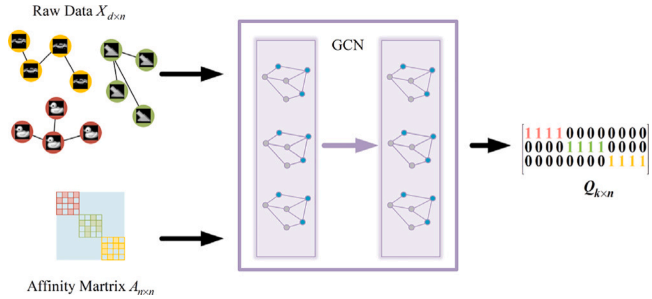


Fig. 2. The illustration of label learning via GCN. Both the raw data X and the initial affinity matrix A are input into a two-layer GCN to obtain the assignment matrix Q, while the activation functions of GCN are Relu in the first layer and softmax in the last layer, respectively. Finally, a post-processing step is applied to Q to obtain its one-hot encoding.

2.4. Game and Nash equilibrium

The game model involves a group of decision makers (or players) [47]. In our case, these players correspond to the ideal assignment matrix and affinity matrix. Let x, y represent two players, and $f(\cdot), g(\cdot)$ represent two loss functions related to the players.

$$\begin{aligned} &f : P^1 \times P^2 \to R \\ &g : P^1 \times P^2 \to R \end{aligned} \tag{15}$$

P^1 and P^2 represent the strategy spaces or feasible domains for two players, respectively. The entire game process can be described as follows:

$$\begin{aligned} &x^{k+1} = \arg\min_{x} f(x, y^k) \\ &y^{k+1} = \arg\min_{y} g(x^k, y) \end{aligned} \tag{16}$$

where (x^0, y^0) are the initial points. The game process continues until the updates for all players no longer reduce the loss functions. This naturally occurring endpoint is recognized as the Nash Equilibrium point, denoted as (x^*, y^*). In other words, for all (x, y), it holds that:

$$f(x^*, y^*) \le f(x, y^*); g(x^*, y^*) \le g(x^*, y) \tag{17}$$

Then (x^*, y^*) is named as a Nash Equilibrium solution of a pair of $(x \in P^1, y \in P^2)$. The game training process aims to find a Nash Equilibrium point. s

3. Method description

In this section, we primarily provide a detailed introduction to our game model. We begin by presenting two submodules separately: one is a semi-supervised module based on label self-expressiveness, and the other is a discriminative affinity learning module. Following this, these two submodules are integrated to construct the proposed game model. Finally, we comprehensively provide the optimization algorithm employed for solving this model.

3.1. Label self-expressiveness based GFHF module

Based on GFHF (9), we utilize GCN to learn predicted labels here, which are constrained by the label self-expressiveness further to enhance its accuracy. The optimization is

$$\begin{aligned} &Q^* = \arg\min_{\theta} Tr((Q - Y)U(Q - Y)^T) + \frac{1}{2} Tr(Q L_C Q^T) \\ &\text{s.t. } Q = G_\theta(X, \hat{A}), Q = QC + E, Q \in \mathcal{Y} \end{aligned} \tag{18}$$

In (18), $Q = G_\theta(X, \hat{A})$ represents the label matrix Q obtained by subjecting the raw data to learning through a GCN. $Q = QC + E$ denotes the label self-expressiveness term with inherent self-expressive noise [9]. Simultaneously, Q must satisfy the conditions of $\mathcal{Y}$, where we define the final layer of the GCN as a softmax layer to ensure that Q complies with the condition $Q^T \mathbf{1}_n = \mathbf{1}_n, rank(Q) = k$. Finally, a post-processing step is applied to Q, setting the maximum value in each column to 1 and the remaining elements to 0, resulting in the learned assignment matrix.

3.2. Discriminative affinity learning module

To leverage the real-time updated assignment matrix for aiding in the learning of the affinity matrix, we construct the following self-expressive model based on the BDR. This model incorporates the local relationship and global geometric structure between the assignment matrix and the affinity matrix.

$$\begin{aligned} &C^* = \arg\min_{C} \frac{1}{2} Tr(Q L_C Q^T) + \gamma \|C\|_{\boxed{k}} \\ &\text{s.t. } Q = QC + E, C^T \mathbf{1}_n = \mathbf{1}_n, diag(C) = 0, C \ge 0, C = C^T \end{aligned} \tag{19}$$

where $Tr(Q L_C Q^T) = \sum_{i,j=1}^{n} \left\| q_i - q_j \right\|^2 C_{ij}$ characterizes the local relationship between the assignment matrix and the affinity matrix. It can be observed that for the given assignment matrix, $q_i \ne q_j$ if x_i and x_j belong to different classes. Minimizing this term ensures that the similarity C_{ij} corresponding to x_i and x_j is smaller. $\|C\|_{\boxed{k}}$ represents the block-diagonal constraint on the affinity matrix C. $Q = QC + E$ stands for the label self-expressiveness term containing self-expressive noise. $C^T \mathbf{1}_n = \mathbf{1}_n$ signifies the requirement of label self-expressiveness on the affinity matrix C, while $diag(C) = 0, C \ge 0, C = C^T$ signifies the block-diagonal regularization on the affinity matrix C.

In this model, the term $Tr(Q L_C Q^T)$ demands that the similarity between samples from different classes be minimized, while the label self-expressiveness $Q = QC + E$ requires that the similarity between samples from the same class is non-zero. From the aforementioned analysis, it is evident that this model is capable of learning a discriminative affinity matrix.

3.3. Game model for semi-supervised subspace clustering

The label self-expressiveness term characterizes the global structural relationship between the assignment matrix and the affinity matrix, while the MSR describes the local relationship between the assignment matrix and the affinity matrix. By incorporating these two terms, we construct a game model to integrate the aforementioned two submodules, facilitating mutual reinforcement in the learning process between the assignment matrix and the affinity matrix. The flowchart of the proposed model is shown in Fig. 3.

$$\begin{aligned} &\begin{cases} Q^* = \arg\min_{\theta} Tr((Q - Y)U(Q - Y)^T) + \frac{1}{2} Tr(Q L_{C^*} Q^T) \\ \text{s.t. } Q = G_\theta(X, \hat{A}), Q = QC^* + E, Q \in \mathcal{Y} \end{cases} \\ &\begin{cases} C^* = \arg\min_{C} \frac{1}{2} Tr(Q^* L_C Q^{*T}) + \gamma \|C\|_{\boxed{k}} \\ \text{s.t. } Q^* = Q^* C + E, C^T \mathbf{1}_n = \mathbf{1}_n, diag(C) = 0, C \ge 0, C = C^T \end{cases} \end{aligned} \tag{20}$$

Here it is abbreviated as:

$$\begin{cases} Q^* = \arg\min_{Q} f(C^*, Q) \\ C^* = \arg\min_{C} g(C, Q^*) \end{cases} \tag{21}$$

A Nash Equilibrium is considered stable when it is attainable as the limit from any initial point. It is locally stable when convergence holds for all initial conditions within an ϵ-neighborhood of the equilibrium solution. Obviously, it is difficult to seek the Nash solution of the above game model directly due to the nonlinearity of GCN. Hence we consider

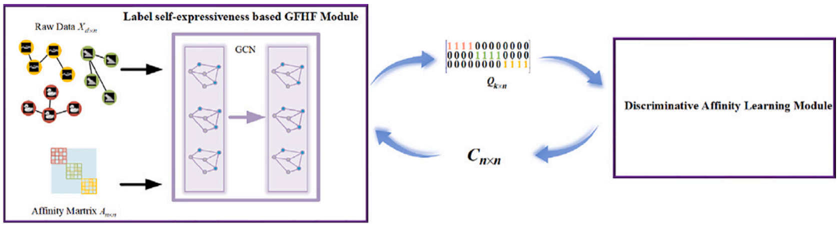

Label self-expressiveness based GFHF Module
Raw Data $X_{d\times n}$
GCN
Affinity Martrix $A_{n\times n}$
1111000000000
0000111100000
0000000001111
$Q_{k\times n}$
$C_{n\times n}$
Discriminative Affinity Learning Module

Fig. 3. The flowchart of the proposed method. The left part is Label self-expressiveness based GFHF Module designed to learn the assignment matrix Q with the affinity matrix C learned from the right part. The right part is the Discriminative Affinity Learning Module which aims to learn C with Q. Both submodules constitutes a game model to obtain the assignment matrix and the affinity matrix.

the locally stable Nash solution. Given t to be the iteration index, for some ε_f and ε_g, the Nash solution pair $\{Q^*, C^*\}$ satisfies:

$$f(C^*,Q^t)-f(C^*,Q^{t-1})\le\varepsilon_f \quad and \quad g(C^t,Q^*)-g(C^{t-1},Q^*)\le\varepsilon_g \tag{22}$$

3.4. Optimization algorithm

In this section, we show how to solve the game model. We alternately solve the two subproblems in (21) until its Nash Equilibrium point is found.

For Q_subproblem, or (18) in Module 1, we train the network by the following loss function with given C until it convergences.

$$loss=Tr((Q-Y)U(Q-Y)^T)+\frac{1}{2}Tr(QL_CQ^T)+\frac{\beta}{2}\|Q-QC\|_F^2 \tag{23}$$

For C_subproblem, or (19) in Module 2, we utilize the alternating minimization approach to update each variable with fixed Q. For ease of solving, we rearrange (19) first. Without altering the essence of the matrices, here the constraint $C^T\mathbf{1}_n=\mathbf{1}_n$ is transformed into $\mathbf{1}_n^TC=\mathbf{1}_n^T$. Thus, (19) can be equivalent to solving the following

$$\min_C \frac{1}{2}\sum_{i,j}^{n}\left\|q_i-q_j\right\|_F^2 C_{ij}+\gamma\|C\|_{\boxed{k}}+\frac{\alpha}{2}\left\|\mathbf{1}_n^TC-\mathbf{1}_n^T\right\|_F^2+\frac{\beta}{2}\|Q-QC\|_F^2$$
$$\text{s.t. } diag(C)=0, C\ge 0, C=C^T \tag{24}$$

When α and β are sufficiently large, these two optimization problems mentioned above become equivalent. Similar to [30], we introduce an auxiliary variable B to solve this optimization problem.

$$\min_{B,C}\frac{1}{2}\sum_{i,j}^{n}\left\|q_i-q_j\right\|_F^2 C_{ij}+\frac{\alpha}{2}\left\|\mathbf{1}_n^TC-\mathbf{1}_n^T\right\|_F^2+\frac{\beta}{2}\|Q-QC\|_F^2+\frac{\lambda}{2}\|B-C\|_F^2+\gamma\|B\|_{\boxed{k}}$$
$$\text{s.t. } diag(B)=0, B\ge 0, B=B^T \tag{25}$$

There is a Proposition regarding $\|B\|_{\boxed{k}}$:

Proposition 3.1 ([30]). *Let $L\in\mathbb{R}^{n\times n}$, and L be a positive semi-definite matrix, leading to the following :*

$$\|B\|_{\boxed{k}}:=\sum_{i=n-k+1}^{n}\lambda_i(L_B)=\min_W\langle L_B,W\rangle,\ s.t.\ 0\le W\le I, Tr(W)=k$$

Hence, (25) is equivalent to:

$$\min_{B,C,W}\frac{1}{2}\sum_{i,j}^{n}\left\|q_i-q_j\right\|_F^2 C_{ij}+\frac{\alpha}{2}\left\|\mathbf{1}_n^TC-\mathbf{1}_n^T\right\|_F^2+\frac{\beta}{2}\|Q-QC\|_F^2+\frac{\lambda}{2}\|B-C\|_F^2$$
$$+\gamma\langle Diag(B\mathbf{1})-B,W\rangle$$
$$\text{s.t. } diag(B)=0, B\ge 0, B=B^T, 0\le \mathrm{W}\le \mathrm{I}, \mathrm{Tr(W)=k} \tag{26}$$

Update the respective variables in an alternating manner as follows.

Update C :

$$C^{k+1}=\arg\min_C\frac{1}{2}\sum_{i,j}^{n}\left\|q_i-q_j\right\|_F^2 C_{ij}+\frac{\alpha}{2}\left\|\mathbf{1}_n^TC-\mathbf{1}_n^T\right\|_F^2+\frac{\beta}{2}\|Q-QC\|_F^2+\frac{\lambda}{2}\|B-C\|_F^2 \tag{27}$$

Setting the derivative of C equal to zero to obtain the solution:

$$C^{k+1}=(\lambda I_n+\beta Q^TQ+\alpha\mathbf{1}_n\mathbf{1}_n^T)^{-1}(-S+\lambda B+\beta Q^TQ+\alpha\mathbf{1}_n\mathbf{1}_n^T) \tag{28}$$

where $S_{ij}=\frac{\left\|q_i-q_j\right\|}{2}$.

Update B :

$$B^{k+1}=\arg\min_B\frac{1}{2}\|B-C\|_F^2+\frac{\gamma}{\lambda}\langle Diag(B\mathbf{1})-B,W\rangle, \text{s.t. } diag(B)=0, B\ge 0, B=B^T \tag{29}$$

The above is equivalent to:

$$B^{k+1}=\arg\min_B\frac{1}{2}\left\|B-C+\frac{\gamma}{\lambda}(diag(W)\mathbf{1}^T-W)\right\|_F^2, \text{s.t. } diag(B)=0, B\ge 0, B=B^T \tag{30}$$

Let $\Lambda=C-\frac{\gamma}{\lambda}(diag(W)\mathbf{1}^T-W)$

Algorithm 1 The Total Training.

```
Input :The raw data $X, Y, U, U, A, \alpha>0, \beta>0, \lambda>0, \gamma>0, \varepsilon_f, \varepsilon_g$
Initialize :$C^0 = B^0 = W^0 = 0, t = 0$
Initialize $Q$ by pretraining GCN;
while not converge do do
    Update $C$:
        Fix others and update $C$ via(28);
        Fix others and update $B$ via (31);
        Fix others and update $W$ via (33);
    Update $Q$ via training GCN via (23);
    Check the convergence conditions:
        $f(C^t,Q^t)-f(C^t,Q^{t-1})\le\varepsilon_f$ and $g(C^t,Q^t)-g(C^{t-1},Q^t)\le\varepsilon_g$
    $t = t + 1$;
end while
Output : Clustering Assignment $Q$ and Affinity Matrix $C$.
```

Proposition 3.2 ([30]). *Let $\Lambda\in\mathbb{R}^{n\times n}$, $\hat{\Lambda}=\Lambda-Diag(diag(\Lambda))$, The solution for*

$$\min_B\frac{1}{2}\|B-\Lambda\|_F^2, s.t.\ diag(B)=0, B\ge 0, B=B^T$$

is then given by:

$$B^*=[(\hat{\Lambda}+\hat{\Lambda}^T)/2]_+$$

where $[(\hat{\Lambda}+\hat{\Lambda}^T)/2]_+=\max(0,(\hat{\Lambda}+\hat{\Lambda}^T)/2)$. *Here, we simply denote it as:*

$$B^{k+1}:=B\_solver(C,W,\lambda,\gamma) \tag{31}$$

Table 1
The details of four datasets.

Dataset	Sample(n)	Dimension(d)	Cluster(k)	Source
Yale	165	1024	15	Face
USPS	1000	256	10	Digital
ORL	400	1024	40	Face
COIL20	1440	1024	20	Object

Table 2
The details of four datasets.

Dataset	Units	Pretraining Epochs	Training Epochs
Yale	800	75	10
USPS	256	60	10
ORL	256	2000	20
COIL20	512	30	40

Update W :

$$W^{k+1} = \arg\min_{W} \langle Diag(B\mathbf{1}) - B, W \rangle \quad \text{s.t. } 0 \leq \text{W} \leq \text{I}, \text{Tr(W)=k} \tag{32}$$

The above has a closed-form solution [30]:

$$W^{k+1} = SS^T \tag{33}$$

Here $S \in \mathbb{R}^{n\times k}$ is a matrix composed of the eigenvectors corresponding to the k smallest eigenvalues of matrix $Diag(B\mathbf{1}) - B$.

Setting up the same initialization as the BDR, the specific optimization process is detailed in the following algorithm 1. Firstly, we pre-train the GCN till the loss (23) convergences where C is replaced by initialized affinity matrix and stop training immediately. Then we get initialized Q with pre-trained GCN. With initialized Q and initialized C, the C_subproblem and Q_subproblem of (21) are optimized alternately until the convergence condition is achieved. According to (33), we set the convergence condition as:

$$f(C^t, Q^t) - f(C^t, Q^{t-1}) \leq \epsilon_f \quad and \quad g(C^t, Q^t) - g(C^{t-1}, Q^t) \leq \epsilon_g \tag{34}$$

At the end of optimization, we output the Clustering Assignment Q and Affinity Matrix C.

3.5. Computational complexity analysis

The optimization procedure in algorithm 1 primarily divided into two parts, one for solving the C-subproblem and another for solving the Q-subproblem. As mentioned earlier, n is the number of samples, k is the number of subspaces (or clusters), and h is the number of hidden units, respectively. $|\epsilon|$ is denoted as the number of initialized affinity graph edges. For solving the C-subproblem, the computation complexity of updating C by (28) is $O(n^3)$. For updating B and W, the complexity of (31)(33) are $O(n^2)$ and $O(kn^2 + n^3)$. For solving the Q-subproblem, the computation complexity of GCN is $O(|\epsilon|hkn)$ due to the dense initialized affinity. Thus, the overall computation complexity of the proposed GMSSC is $O(\tau(|\epsilon|hkn + kn^2 + n^3))$.

4. Experiments

To validate the effectiveness of the proposed method, we conducted experiments on four benchmark datasets. These experiments were organized following these four aspects: Clustering Results, Parameter Analysis, Ablation Study, and Convergence Analysis. The results from this series of experiments demonstrate that our method yields superior clustering results on the real-world datasets. Furthermore, the proposed GMSSC shows relative robustness in parameter selection. During the game training process, both clustering accuracy and objective functions converge to a stable state rapidly.

4.1. Datasets and the baseline methods

We conducted experiments on the following four open datasets: ORL, USPS, Yale [48], and COIL20 [49]. These datasets contain real-world samples of objects, faces, and digits. More detailed information regarding these datasets is available in Table 1. In each experiment conducted for all datasets, we randomly selected varying percentages of labels from the true labels to construct a known assignment matrix, encompassing {5%, 10%, 15%, 20%, 25%, 30%}. Details of sample images are illustrated in Fig. 4.

1. Yale comprises facial images of 15 individuals, each captured in 11 different facial expressions or configurations, including variations such as center light, wearing glasses, smiling, left light, without glasses, normal, right light, sad, sleepy, surprised, blinking, and others. This dataset encompasses a total of 165 facial images. For computational efficiency, all samples were downsampled to images sized at 32 × 32.

2. The USPS dataset includes images representing digits ranging from 0 to 9, with 100 samples per digit, totaling 1000 samples. These images are digital data automatically scanned from envelopes by the United States Postal Service. They are standardized, centered, and present diverse font styles. Similarly, we downsampled them to images of size 16 × 16.

3. The ORL dataset consists of 400 facial images belonging to 40 unique individuals, with each person having 10 images. These images were captured under varying conditions of time, lighting, facial expressions (open/closed eyes, smiling/not smiling), and facial details (with/without glasses). All images were taken against a dark, uniform background, and most subjects were facing the camera directly, with some having slight tilts. Similarly, we downsampled them to images of size 32 × 32. It is worth noting that it is challenge to handle this dataset due to its numerous categories and a limited number of samples within each class.

4. The COIL20 dataset comprises images of 20 different objects. Each object is photographed from 0° to 360° at 5° intervals, resulting in a total of 72 images per object. Similarly, we downsampled these images to 32 × 32.

To demonstrate the effectiveness of the proposed approach, we conduct a comparative analysis against several state-of-the-art semi-supervised subspace clustering methods, including STSSL [21], NNLRR [22], DCSSC [23], SSLRR [17], DPLRR [18], and SSC_TLRR [19].

To ensure a fair comparison, we carefully fine-tuned the parameters of all methods to attain optimal results. Each method was run five times on all datasets, and we recorded the average of their clustering accuracy. Clustering accuracy (acc), a commonly used metric to assess clustering performance, is determined by comparing the match between labels generated by the model for each sample and the true labels. It is formally defined as follows [50]:

$$\text{acc} = \frac{\sum_{i=1}^{n} \delta(y_i, \text{map}(q_i))}{n} \times 100\% \tag{35}$$

where q_i represents the labels generated by the algorithm, y_i denotes the true labels provided by the dataset. $\text{map}(\cdot)$ refers to a sorting function that maps q_i to labels equivalent to y_i. $\delta(y_i, \text{map}(q_i))$ is a Dirac function, its value is 1 when $y_i = \text{map}(q_i)$, and is 0 otherwise.

In all our experiments, a two-layer GCN is adopted here and the number of units of the hidden layer is shown in Table 2. We train GCN by Adam optimizer with learning rate of 1e-3 in the pretraining step and 0.5*1e-3 in training step, and stop iterating if the convergence condition are achieved. We initialize all network weights by he_normal weight initializer. The pretraining epochs and training epochs in each iteration are list in Table 2. In addition, we construct the initialized sparse affinity matrix according to the known labels as follows:

$$A_{ij} = \begin{cases} 1, \ if \ q_i = q_j \ (i, j = 1, \ldots, u) \\ 0, \ others \end{cases}$$

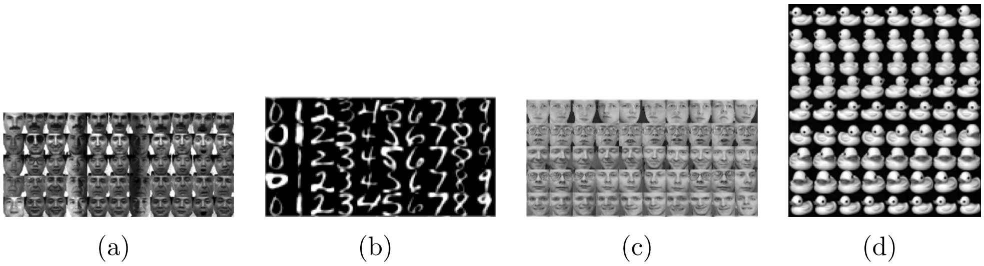


Fig. 4. Sample images of datasets. (a) Yale. (b) USPS. (c) ORL. (d) COIL20.

Table 3
The performance of the proposed method on USPS, ORL, Yale, COIL20 datasets. The highest scores are in bold.

Yale						
Per.	5%	10%	15%	20%	25%	30%
STSSL	38.91 ± 2.22	39.40 ± 1.99	48.73 ± 2.78	66.42 ± 1.74	71.39 ± 2.53	77.58 ± 1.58
NNLRR	36.61 ± 2.91	39.03 ± 2.67	56.36 ± 3.04	65.21 ± 2.05	71.52 ± 1.71	75.88 ± 2.47
DCSSC	37.58 ± 2.63	39.88 ± 2.34	56.12 ± 1.65	57.09 ± 0.45	63.03 ± 0.77	71.39 ± 0.59
SSLRR	**44.97 ± 3.54**	48.64 ± 2.58	50.67 ± 2.09	52.24 ± 3.10	53.45 ± 2.28	54.79 ± 2.19
DPLRR	38.06 ± 3.82	43.15 ± 2.25	52.24 ± 2.01	52.73 ± 2.89	64.12 ± 2.44	70.30 ± 1.99
SSC_TLRR	42.18 ± 2.67	45.09 ± 2.47	50.67 ± 2.12	53.70 ± 2.09	59.15 ± 1.86	66.30 ± 2.09
GMSSC	39.40 ± 0.01	**50.79 ± 0.45**	**61.52 ± 0.01**	**71.94 ± 0.82**	**75.15 ± 0.01**	**82.61 ± 0.82**
USPS						
Per.	5%	10%	15%	20%	25%	30%
STSSL	64.54 ± 6.56	79.50 ± 4.59	81.88 ± 2.86	84.48 ± 2.33	88.44 ± 1.58	89.16 ± 1.22
NNLRR	63.24 ± 5.43	78.14 ± 2.35	80.56 ± 1.42	82.54 ± 1.26	85.36 ± 0.98	88.02 ± 0.88
DCSSC	**80.04 ± 4.05**	86.00 ± 1.96	89.22 ± 1.45	92.54 ± 0.86	92.86 ± 0.67	93.82 ± 0.53
SSLRR	62.34 ± 3.56	72.96 ± 2.22	75.82 ± 1.06	78.48 ± 0.98	79.86 ± 0.84	81.32 ± 0.86
DPLRR	66.32 ± 4.58	76.98 ± 2.36	82.50 ± 2.43	84.68 ± 2.02	88.74 ± 1.48	90.56 ± 1.00
SSC_TLRR	68.83 ± 2.34	**89.45 ± 1.02**	91.88 ± 0.95	91.90 ± 0.78	93.03 ± 1.32	94.03 ± 0.88
GMSSC	72.60 ± 0.01	85.40 ± 0.01	**92.20 ± 0.01**	**92.90 ± 0.01**	**94.50 ± 0.01**	**96.00 ± 0.01**
ORL						
Per.	5%	10%	15%	20%	25%	30%
STSSL	40.40 ± 3.27	56.70 ± 1.30	67.30 ± 1.16	75.60 ± 1.25	80.15 ± 1.56	84.45 ± 1.49
NNLRR	34.10 ± 4.61	53.25 ± 2.33	64.85 ± 3.07	73.80 ± 2.25	78.60 ± 1.40	82.40 ± 1.95
DCSSC	37.45 ± 3.06	55.20 ± 2.08	69.10 ± 1.71	76.35 ± 1.68	81.50 ± 1.87	86.35 ± 1.83
SSLRR	72.60 ± 3.16	73.75 ± 2.59	74.85 ± 1.78	75.15 ± 1.79	76.45 ± 1.35	76.75 ± 1.34
DPLRR	35.05 ± 3.40	57.25 ± 1.47	75.70 ± 1.54	77.55 ± 1.97	83.15 ± 1.46	84.30 ± 1.96
SSC_TLRR	**75.15 ± 4.91**	75.40 ± 3.42	**83.80 ± 2.52**	84.30 ± 1.35	89.05 ± 0.91	89.70 ± 0.94
GMSSC	66.60 ± 0.46	**78.93 ± 0.42**	83.23 ± 0.24	**86.55 ± 0.19**	**89.95 ± 0.42**	**90.53 ± 0.41**
COIL20						
Per.	5%	10%	15%	20%	25%	30%
STSSL	69.43 ± 4.88	82.03 ± 2.01	83.40 ± 2.87	90.29 ± 1.16	93.36 ± 1.32	94.20 ± 0.58
NNLRR	66.40 ± 3.21	81.10 ± 1.67	86.22 ± 0.92	89.92 ± 1.25	92.54 ± 1.19	94.96 ± 0.51
DCSSC	69.43 ± 3.39	85.03 ± 2.50	89.36 ± 1.91	91.18 ± 1.58	92.34 ± 0.50	95.81 ± 0.53
SSLRR	66.16 ± 3.57	68.95 ± 3.05	69.34 ± 2.77	69.65 ± 2.13	69.75 ± 2.09	71.22 ± 1.10
DPLRR	73.04 ± 3.11	79.89 ± 1.75	84.34 ± 0.54	86.05 ± 0.56	88.21 ± 0.50	90.15 ± 0.54
SSC_TLRR	81.21 ± 2.35	**91.73 ± 1.11**	93.85 ± 1.27	95.40 ± 1.17	95.99 ± 0.45	97.45 ± 0.34
GMSSC	**88.89 ± 0.09**	91.27 ± 0.03	**94.79 ± 0.04**	**96.13 ± 0.03**	**97.57 ± 0.06**	**98.54 ± 0.03**

4.2. Clustering results

Table 3 illustrates the mean of clustering accuracies (%) and the standard deviations of all methods across the four datasets. Bold values within the table denote the highest clustering accuracy, while underlined values signify the second-best performance. It is evident that our GMSSC exhibits relatively superior experimental results in terms of accuracy when compared to all other methods, except for those following individual cases. Notably, it ranks second in specific cases such as {15% of ORL, 10% of COIL20, 5% of USPS}, and third in {5% of ORL, 5% of COIL20, 10% of USPS}. Overall, our method consistently ranks within the top three performers. Particularly, for Yale, it is the significant improvement achieved by GMSSC compared to the second-best method, with enhancements of {2.15%, 5.16%, 5.52%, 3.63%, 5.03%} for {10%, 15%, 20%, 25%, 30%}, respectively. This notable performance advantage validates the effectiveness of the proposed game model, highlighting the beneficial impact of dynamically updating the graph for achieving superior spectral clustering results.

4.3. Ablation study

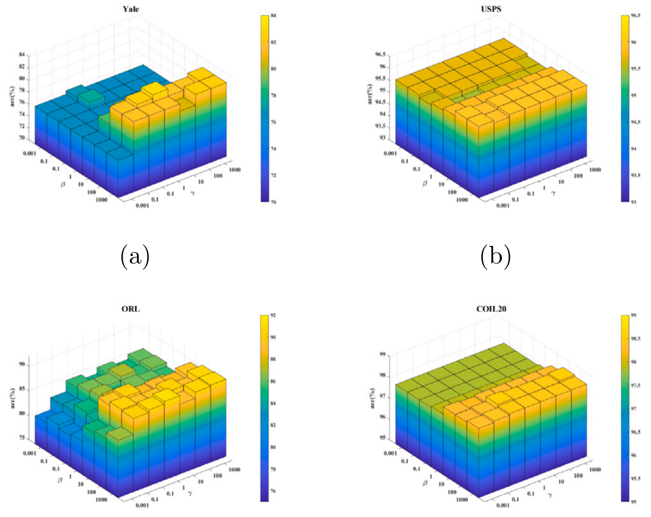
The label self-expressive term and MSR are important elements within our game model, each considering the global geometric structure of the assignment matrix and the other focusing on the local relationship of the assignment matrix and the affinity matrix. To verify the necessity of these two components in GMSSC, we conducted an ablation study on four datasets, as detailed in Table 4. Here, we set the percentage of known labels to 30% and fine-tuned all parameters, subsequently recording the average of the best results of five times. The experimental results on four datasets indicate that the absence of either term leads to a decline in clustering accuracy.

4.4. Parameter analysis

The label self-expressive term stands as a crucial component within the architecture of the proposed model. To thoroughly investigate its significance in GMSSC, we conducted an analysis on the parameters

Table 4
Ablation study of the proposed method on four datasets.

Per.30%	Yale	ORL	USPS	COIL20
No MSR	75.15	85.15	93.20	96.32
No LS	73.33	84.78	86.90	95.56
GMSSC	**82.61**	**90.53**	**96.00**	**98.54**



(a) (b)

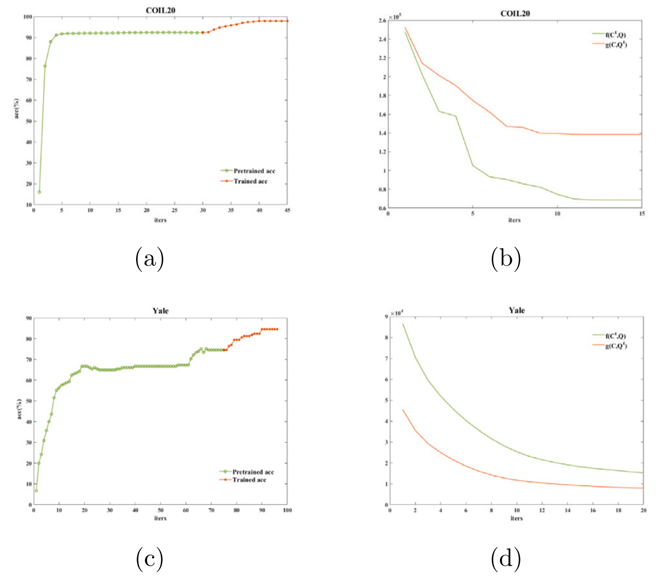
(c) (d)

Fig. 5. Sensitivity analysis of parameters on Yale (a), USPS (b), ORL (c) and COIL20 (d).

of this term. Besides, as described in Eq. (1), traditional subspace clustering methods include various regularization terms of C. The selection of k-block diagonal regularization is crucial in our approach. Therefore, we further examined the parameters of this term across the four datasets with 30% labels known, as visually depicted in Fig. 5. We evaluated the effects by setting the parameter selection range as {0.001, 0.01, 0.1, 1.0, 10.0, 100.0, 1000.0} while other parameters are fixed. Notably, the analysis from these graphs illustrates that our method achieves nearly optimal experimental results across a relatively wide range, i.e., $\beta \in [10.0 - 1000.0]$, $\gamma \in [0.1 - 1000.0]$. This implies that our method is robust to these parameters. It is noteworthy that higher values of β are necessary to attain improved experimental outcomes, aligning with the penalization formulation of GMSSC for handling the label self-expressive term, which necessitates a larger penalty factor. Furthermore, overall, larger γ values result in better experimental outcomes, signifying the vital importance of the k-block diagonal term in our model. Without emphasizing otherwise, we set the parameters α and λ to be 100.0, 100.0 empirically.

4.5. Convergence analysis

Figs. 6 illustrate the convergence curves of our method on COIL20 and Yale datasets, demonstrating both the clustering accuracy and the two objective functions involved to the game model. In Figs. 6(a) and 6(c), the convergence process of clustering accuracy for GMSSC is presented. The green line indicates the status of acc (%) during the pre-training stage, while the yellow line tracks the changes in acc (%) during the game process. Notably, it can be observed that acc (%) gradually increases and stabilizes during the game process. Figs. 6(b) and 6(d) depict the variations in the two objective functions involved in the game model. The green line signifies the change of f in Eq. (21), while the yellow line denotes the variation of g. Both objective functions exhibit rapid descent and eventual convergence.



(a) (b)

(c) (d)

Fig. 6. The acc curve (a) (c) and the objective function (b) (d) of the proposed method over iterations on COIL20 and Yale.

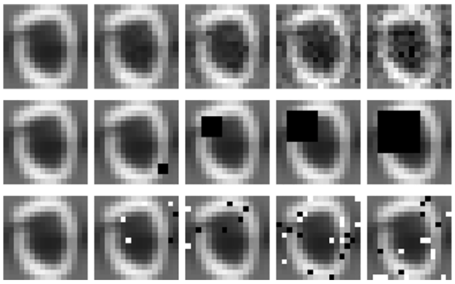


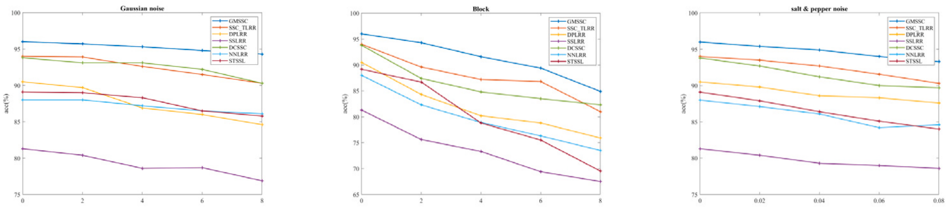
Fig. 7. An example image in USPS corrupted by different types of noises. Top: Gaussian noise. Middle: Block. Bottom: salt & pepper noise.

4.6. Robustness

To assess the robustness of GMSSC against disturbances such as noise and outliers, we conduct experiments on the perturbed USPS dataset in this section. Providing 30% of all labels as known labels, there are three types of perturbations: Gaussian noise-corrupted, Block-masked, and salt & pepper noise-corrupted. Fig. 7 shows an example image under different types and different degrees of disturbances. Specifically, for Gaussian noise-corrupted data, we add different levels of Gaussian noise to the clean data with noise level $\sigma \in \{0.0, 0.02, 0.04, 0.06, 0.08\}$. For Block-masked data, we apply b $*$ b-block blocking positioned randomly to each clean image, with $b \in \{0, 2, 4, 6, 8\}$. Regarding salt & pepper noise-corrupted data, $s\%$ of all pixels in each image are noised, where $s \in \{0, 2, 4, 6, 8\}$. The larger σ, s, b is, the greater the corruption is. Fig. 8 presents the clustering accuracy curves of different methods under three disturbances. The observation reveals that the proposed method adeptly handles various noise types, achieving superior clustering accuracy compared to alternative methods.

5. Conclusion

This paper introduces a dynamic game model designed for simultaneous label learning and affinity matrix learning. The method comprises two submodules, capitalizing on both the global geometric structure and the local relationship between the assignment matrix and



(a) (b) (c)

Fig. 8. Clustering results under different types of outliers on the USPS dataset. (a) Gaussian noise. (b) Block. (b) salt & pepper noise.

the affinity matrix. One submodule implements GCN to achieve end-to-end learning of the assignment matrix, while the other leverages the assignment matrix to facilitate affinity matrix learning. A self-expressive model is formulated by utilizing the assignment matrix, eliminating modeling of the raw data directly and consequently reducing the influence of data noise. Meanwhile, the assignment matrix and affinity matrix assist each other mutually in the game process, thereby enhancing the quality of both matrices. Experimental results on four real-world datasets illustrate the superior performance of our approach compared to existing semi-supervised subspace clustering methods. These results show robustness across various parameters and exhibit rapid convergence.

Although the proposed method achieves elegant results in clustering through end-to-end learning from raw data to the assignment matrix, avoiding the complex process of spectral clustering. There are still three aspects that need further investigation. Firstly, operations like inverse operation and eigenvalue decomposition are involved in solving the C-subproblem, which increases the computational complexity of the algorithm. Consequently, this limitation prevents the method from being extended to large-scale datasets. Hence, reducing the computational complexity of GMSSC while maintaining high-quality assignment matrix and affinity matrix remains a challenge. Secondly, the proposed method mainly deals with high-dimensional graph-based data which is currently appeared in Image Processing, Biomedical Research, etc. In the real-world applications, it is essential to check the data used to avoid the bias. Continuous monitoring and feedback mechanisms should be in place to adapt the proposed model to changing circumstances and mitigate unforeseen consequences. Any data that can be represented as a graph can be clustered by GMSSC. How to apply the idea presented in GMSSC to handle different types of data structures and modalities will be one of our key focuses in the future. Finally, it is still a crucial problem that the interpretability of the proposed model is difficult to resolve due to the GCN, which we will explore furthermore.

CRediT authorship contribution statement

Tingting Qi: Writing – review & editing, Writing – original draft, Software, Methodology, Conceptualization. **Xiangchu Feng:** Writing – review & editing, Methodology, Conceptualization. **Weiwei Wang:** Methodology, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

The authors would like to express their gratitude to the editors and anonymous reviewers for their constructive comments and suggestions. This work was supported by the National Natural Science Foundation of China (Grant No. s 61972264, 61772389, 62072312 and 62372359).

References

[1] M.F.A. Hady, F. Schwenker, Semi-supervised learning, in: M. Bianchini, M. Maggini, L.C. Jain (Eds.), Handbook on Neural Information Processing, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 215–239.
[2] O. Chapelle, B. Scholkopf, A. Zien, Semi-supervised learning (chapelle, o. et al. eds.; 2006) [book reviews], IEEE Trans. Neural Netw. 20 (3) (2009) 542.
[3] Y. Grandvalet, Y. Bengio, Semi-supervised learning by entropy minimization, in: L. Saul, Y. Weiss, L. Bottou (Eds.), in: Advances in Neural Information Processing Systems, vol. 17, MIT Press, 2004.
[4] A. Blum, J. Lafferty, M.R. Rwebangira, R. Reddy, Semi-supervised learning using randomized mincuts, in: Proceedings of the Twenty-First International Conference on Machine Learning, ICML '04, Association for Computing Machinery, New York, NY, USA, 2004, p. 13.
[5] K. Chen, S. Wang, Semi-supervised learning via regularized boosting working on multiple semi-supervised assumptions, IEEE Trans. Pattern Anal. Mach. Intell. 33 (1) (2010) 129–143.
[6] Z. Yang, W. Cohen, R. Salakhudinov, Revisiting semi-supervised learning with graph embeddings, in: International Conference on Machine Learning, PMLR, 2016, pp. 40–48.
[7] D.-H. Lee, et al., Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks, in: Workshop on Challenges in Representation Learning, Vol. 3, No. 2, ICML, Atlanta, 2013, p. 896.
[8] Z.-H. Zhou, M. Li, Semi-supervised learning by disagreement, Knowl. Inf. Syst. 24 (2010) 415–439.
[9] T. Qi, X. Feng, B. Gao, K. Wang, An end-to-end Graph Convolutional Network for Semi-supervised Subspace Clustering via label self-expressiveness, Knowl.-Based Syst. 286 (2024) 111393.
[10] W. Zhan, M.-L. Zhang, Inductive semi-supervised multi-label learning with co-training, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2017, pp. 1305–1314.
[11] F. Nie, G. Cai, X. Li, Multi-view clustering and semi-supervised classification with adaptive neighbours, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 31, No. 1, 2017.
[12] X. Li, M. Chen, F. Nie, Q. Wang, Locality adaptive discriminant analysis, in: IJCAI, Vol. 2201, No. 2207, 2017.
[13] Z. Chen, H. Cao, K.C.-C. Chang, GraphEBM: Energy-based graph construction for semi-supervised learning, in: 2020 IEEE International Conference on Data Mining, ICDM, IEEE, 2020, pp. 62–71.
[14] Z. Zhu, Q. Gao, Semi-supervised clustering via cannot link relationship for multiview data, IEEE Trans. Circuits Syst. Video Technol. 32 (12) (2022) 8744–8755.
[15] Y. Cai, Z. Zhang, P. Ghamisi, Z. Cai, X. Liu, Y. Ding, Fully linear graph convolutional networks for semi-supervised and unsupervised classification, ACM Trans. Intell. Syst. Technol. 14 (3) (2023) 1–23.
[16] Z. Wang, L. Zhang, R. Wang, F. Nie, X. Li, Semi-supervised learning via bipartite graph construction with adaptive neighbors, IEEE Trans. Knowl. Data Eng. 35 (5) (2022) 5257–5268.
[17] L. Zhuang, Z. Zhou, S. Gao, J. Yin, Z. Lin, Y. Ma, Label information guided graph construction for semi-supervised learning, IEEE Trans. Image Process. 26 (9) (2017) 4182–4192.
[18] H. Liu, Y. Jia, J. Hou, Q. Zhang, Learning low-rank graph with enhanced supervision, IEEE Trans. Circuits Syst. Video Technol. 32 (4) (2022) 2501–2506.
[19] Y. Jia, G. Lu, H. Liu, J. Hou, Semi-supervised subspace clustering via tensor low-rank representation, IEEE Trans. Circuits Syst. Video Technol. 33 (7) (2023) 3455–3461.

[20] S. Basu, A. Banerjee, R.J. Mooney, Active semi-supervision for pairwise constrained clustering, in: Proceedings of the 2004 SIAM International Conference on Data Mining, SDM, pp. 333–344.

[21] C.-G. Li, Z. Lin, H. Zhang, J. Guo, Learning semi-supervised representation towards a unified optimization framework for semi-supervised learning, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 2767–2775.

[22] X. Fang, Y. Xu, X. Li, Z. Lai, W.K. Wong, Robust semi-supervised subspace clustering via non-negative low-rank representation, IEEE Trans. Cybern. 46 (8) (2015) 1828–1838.

[23] W. Wang, C. Yang, H. Chen, X. Feng, Unified discriminative and coherent semi-supervised subspace clustering, IEEE Trans. Image Process. 27 (5) (2018) 2461–2470.

[24] K. Tang, R. Liu, Z. Su, J. Zhang, Structure-constrained low-rank representation, IEEE Trans. Neural Netw. Learn. Syst. 25 (12) (2014) 2167–2179.

[25] F. Wang, C. Zhang, Label propagation through linear neighborhoods, in: Proceedings of the 23rd International Conference on Machine Learning, 2006, pp. 985–992.

[26] S. Yan, H. Wang, Semi-supervised learning by sparse representation, in: Proceedings of the 2009 SIAM International Conference on Data Mining, SIAM, 2009, pp. 792–801.

[27] X. Zhu, Z. Ghahramani, J.D. Lafferty, Semi-supervised learning using gaussian fields and harmonic functions, in: Proceedings of the 20th International Conference on Machine Learning, ICML-03, 2003, pp. 912–919.

[28] E. Elhamifar, R. Vidal, Sparse subspace clustering: Algorithm, theory, and applications, IEEE Trans. Pattern Anal. Mach. Intell. 35 (11) (2013) 2765–2781.

[29] G. Liu, Z. Lin, S. Yan, J. Sun, Y. Yu, Y. Ma, Robust recovery of subspace structures by low-rank representation, IEEE Trans. Pattern Anal. Mach. Intell. 35 (1) (2013) 171–184.

[30] C. Lu, J. Feng, Z. Lin, T. Mei, S. Yan, Subspace clustering by block diagonal representation, IEEE Trans. Pattern Anal. Mach. Intell. 41 (2) (2019) 487–501.

[31] Y. Yuan, X. Li, Q. Wang, F. Nie, A semi-supervised learning algorithm via adaptive Laplacian graph, Neurocomputing 426 (2021) 162–173.

[32] F. Nie, H. Zhang, R. Wang, X. Li, Semi-supervised clustering via pairwise constrained optimal graph, in: Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence, 2021, pp. 3160–3166.

[33] H. Ling, C. Bao, X. Liang, Z. Shi, Semi-supervised clustering via dynamic graph structure learning, 2022, arXiv preprint arXiv:2209.02513.

[34] Z. Song, Y. Zhang, I. King, Optimal block-wise asymmetric graph construction for graph-based semi-supervised learning, in: Thirty-Seventh Conference on Neural Information Processing Systems, 2023.

[35] A.K. Ziemann, D.W. Messinger, P.S. Wenger, An adaptive k-nearest neighbor graph building technique with applications to hyperspectral imagery, in: 2014 IEEE Western New York Image and Signal Processing Workshop, WNYISPW, IEEE, 2014, pp. 32–36.

[36] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, 2016, arXiv preprint arXiv:1609.02907.

[37] T. Yang, S. Zhou, Z. Zhang, The k-sparse LSR for subspace clustering via 0-1 integer programming, Signal Process. 199 (2022) 108622.

[38] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 888–905.

[39] G. Chen, S. Atev, G. Lerman, Kernel spectral curvature clustering (KSCC), in: 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops, IEEE, 2009, pp. 765–772.

[40] V.M. Patel, R. Vidal, Kernel sparse subspace clustering, in: 2014 IEEE International Conference on Image Processing, ICIP, 2014, pp. 2849–2853.

[41] M. Liu, Y. Wang, J. Sun, Z. Ji, Adaptive low-rank kernel block diagonal representation subspace clustering, Appl. Intell. 52 (2022) 2301–2316.

[42] T. Qi, X. Feng, W. Wang, X. Li, Game theory based Bi-domanial deep subspace clustering, Inform. Sci. 617 (2022) 150–164.

[43] P. Ji, T. Zhang, H. Li, M. Salzmann, I. Reid, Deep subspace clustering networks, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS '17, Curran Associates Inc., Red Hook, NY, USA, 2017, pp. 23–32.

[44] J. Zhang, C.-G. Li, C. You, X. Qi, H. Zhang, J. Guo, Z. Lin, Self-supervised convolutional subspace clustering network, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 5473–5482.

[45] F. Nie, Z. Zeng, I.W. Tsang, D. Xu, C. Zhang, Spectral embedded clustering: A framework for in-sample and out-of-sample spectral clustering, IEEE Trans. Neural Netw. 22 (11) (2011) 1796–1808.

[46] J. Ye, Least squares linear discriminant analysis, in: Proceedings of the 24th International Conference on Machine Learning, 2007, pp. 1087–1093.

[47] A. Chakraborty, J.S. Duncan, Game-theoretic integration for image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 21 (1) (1999) 12–30.

[48] F. Nie, S. Xiang, Y. Song, C. Zhang, Orthogonal locality minimizing globality maximizing projections for feature extraction, Opt. Eng. 48 (1) (2009) 017202.

[49] C.-Y. Lu, H. Min, Z.-Q. Zhao, L. Zhu, D.-S. Huang, S. Yan, Robust and efficient subspace segmentation via least squares regression, in: Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part VII 12, Springer, 2012, pp. 347–360.

[50] Y. Xu, S. Chen, J. Li, L. Luo, J. Yang, Learnable low-rank latent dictionary for subspace clustering, Pattern Recognit. 120 (2021) 108142.