TSINGHUA SCIENCE AND TECHNOLOGY
ISSN 1007-0214 21/38 pp1661–1677
DOI: 10.26599/TST.2024.9010218
Volume 31, Number 3, June 2026

Research Article

Regularized Semi-Supervised Subspace Clustering with Local-Structure-Preservation-Based Soft-MFA

Xin Lv, Zhenming Su*, and Baifei Chai

Abstract: Semi-supervised subspace clustering (SSC) algorithms utilize partially labeled data to improve clustering performance. However, existing SSC methods often focus on the global similarity structure of data and neglect the local structure to some extent. To address this limitation, we propose a soft marginal Fisher analysis (SMFA) based regularization term to constrain the data after projection, preserving the intraclass similarity and enhancing interclass discrimination in the low-dimensional space. By incorporating the SMFA regularization term, the proposed SSC framework encourages low-rank subspaces to separate data points from different classes while grouping together data points from the same class. Furthermore, by employing high-resolution ordinary differential equations (ODEs), our study explores the convergence dynamics of the alternating direction method of multiplier (ADMM) in addressing low-rank representation (LRR) challenges. Experimental results on five publicly available databases demonstrate that our proposed method outperforms state-of-the-art SSC methods in terms of clustering accuracy and stability. These results demonstrate the effectiveness of our proposed regularization term and its ability to preserve both the global and local similarity structure of data in low-rank subspaces.

Key words: semi-supervised learning; low-rank representation; label propagation; subspace clustering

1 Introduction

Semi-supervised clustering, which aims to leverage both labeled and unlabeled data to improve the clustering performance, has attracted great interest in machine learning and image analysis communities[1–3]. Clustering problems in these fields often involve large number of high-dimensional data. It has been demonstrated that high-dimensional data are often lying near several low-dimensional subspaces[4]. Among various graph-based clustering methods, the self-expressive model is a promising one for capturing the underlying low-dimensional structures of data. It represents each data point as a linear combination of other data points, and the representation matrix can be utilized to express the affinities among all data points.

There are two commonly used self-expressive models for affinity construction: the sparse self-expressive model and the low-rank representation (LRR) model, for which the representation matrix is either sparse or low-rank[4]. LRR has been widely used in subspace clustering due to its efficiency in capturing the global structure of data and its robustness to noise and corruptions[5–7]. LRR is an unsupervised learning method. Existing semi-supervised subspace clustering (SSC) methods typically extend unsupervised LRR to semi-supervised cluster learning through graph-based regularizations, e.g., the local and global consistency (LGC) algorithm[8], Gaussian fields and harmonic function (GFHF)[9], and the propagation learning framework[10], which are known as label propagation (LP) methods in semi-supervised learning (SSL). These regularization terms encourage data points that are not only similar in the graph but also share the same class

• Xin Lv, Zhenming Su, and Baifei Chai are with School of Information Science and Engineering, Lanzhou University, Lanzhou 730000, China. E-mail: lvxin@lzu.edu.cn; suzhm@lzu.edu.cn; chaibaifei1@163.com.
* To whom correspondence should be addressed.
Manuscript received: 2024-07-31; revised: 2024-10-09; accepted: 2024-10-29

label to be grouped into the same cluster. Zhang presents a nonnegative low-rank and sparse (NNLRS) graph for SSC, in which LGC or GFHF is utilized with the learned graph[11]. To reveal the intrinsic and discriminative structure among the data, several manifold learning based graph regularizers have been introduced into the LRR framework. For example, Zhuang et al. incorporate a similarity-preserving regularizer into the nonnegative low-rank and sparse graph[12]. By considering both the data manifold and feature manifold, Yin et al. present a dual graph regularized latent low-rank representation for subspace clustering[13]. Yang et al. present a semi-supervised kernel low-rank representation graph by combining LRR with the kernel trick[14]. Although informative graphs can be obtained via the LRR model with proper regularizations, previous works performed affinity learning and the subsequent clustering separately, which may not optimize the clustering.

To further improve the clustering capability and ensure efficient representation, several unified frameworks jointly considering the self-expressive representation and label propagation have been presented. For example, semi-supervised LRR (S^2LRR) and semi-supervised sparse representation (S^3R) are presented to be incorporated into the unified SSC framework to learn the affinity matrix and infer the unknown labels simultaneously[15]. Fang et al. present a nonnegative LRR (NNLRR) method for semi-supervised subspace clustering by combining the LRR model and the GFHF method in a single optimization problem[5]. Pei et al. present a joint sparse representation and embedding propagation learning (JSREPL) method for semi-supervised clustering, which incorporates sparse representation into a general LP framework, enabling the construction of the affinity matrix and the estimation of the labels of unlabeled data simultaneously[16]. Taking the non-linear geometric structures within data into account, Yin et al. present a nonnegative sparse hyper-Laplacian regularized LRR (NSHLRR) model based framework for data representation and clustering[6]. By considering both the coherence of the affinity between data points from the same subspace and the discrimination of cluster labels for data points from different subspaces, Wang et al. present a discriminative and coherent semi-supervised subspace clustering method[17]. Qi et al. present a semi-supervised subspace clustering approach utilizing a graph convolutional network to enhance clustering accuracy by leveraging the label self-expressiveness[18].

Traditional LRR-based methods effectively capture the global structure of data, but some of them may fail to reveal the local structure, which can be important for clustering tasks. Although aforementioned graph regularized LRR methods can preserve the local geometrical structure to some extent, some fail to fully capture the discriminative information embedded in high-dimensional data or do not sufficiently consider it. Furthermore, high-dimensional data often exhibit strong correlations among features, leading to unstable and unreliable clustering results. To overcome this problem, several methods introduce effective semi-supervised dimensionality reduction methods by combining the global learning structure of LRR with local intra class connections and inter class separation of data, i.e., discriminative low-rank preserving projection (DLRPP)[19], Laplacian regularized nonnegative representation (LapNR)[20], and low-rank projection learning via graph embedding (LRP-GE)[21]. Although these dimension reduction based methods can effectively cope with high-dimensional redundant data, their focus on dimensionality reduction does not take into account the classification of data points. Deep learning based approaches have been presented to address these issues[22−24], but they often require loading the entire dataset into memory, which limits their scalability and the depth of networks that can be employed.

In this paper, we propose a novel semi-supervised subspace clustering joint optimization framework to address the problems mentioned above in existing SSC methods. First, a novel soft-marginal Fisher analysis (SMFA) based regularization term is proposed to improve the clustering performance of the LRR subspace clustering algorithm, which aims to find a low-rank representation of the high-dimensional data in a low-dimensional space. The regularization term encourages the intraclass data to be more compact and the interclass data to be more dispersed in the low-dimensional embedding space. This regularization term, together with the low-rank constraint, enforces the clustering-friendly structure to be preserved in the low-rank subspace. Intuitively, this will enable LRR to find more discriminative graph affinities and LP to efficiently propagate labels, leading to improved clustering performance. Second, by jointly learning the mapping matrix, LRR representation matrix, and label

propagation matrix, the proposed semi-supervised subspace clustering framework can further improve the clustering performance. In brief, we summarize the major contributions of this work as follows:

(1) We propose a regularization term for semi-supervised clustering, which constrains the low-rank representation matrix to preserve intraclass compactness and interclass separability.

(2) We propose a unified framework that jointly considers data projection, low-rank representation, and label propagation to improve affinity representation and clustering performance.

(3) An alternatively iterating algorithm is well designed for solving the optimization of the proposed problem.

(4) We utilize the high-resolution ordinary differential equations (ODEs) to demonstrate the convergence of the proposed algorithm that includes matrix variables.

2 Related Work

2.1 LRR-based subspace clustering

Let $X=[x_1, x_2, \ldots, x_n]\in\mathbb{R}^{m\times n}$ be a matrix, whose columns represent data points drawn from a union of $\varkappa$ subspaces, where $x_i\in\mathbb{R}^{m\times 1}$ is the i-th sample, n represents the number of data points in the dataset, and m is the feature dimension. The self-expressive model represents each data point as a linear combination of other points in the dataset, i.e., $X=XC$, where C is the coefficient matrix and C_{ij} can be explained as the affinity between the i-th data x_i and the j-th data x_j. A low-rank representation model is presented by adding low-rank constraints in self-expression model[25]. It aims to identify the matrix C that represents the data with the lowest possible rank, and this can be formulated as

$$\begin{aligned}&\min_{C,E}\ \|C\|_*+\lambda\|E\|_{2,1},\\&\text{s.t.},\quad X=XC+E\end{aligned}\tag{1}$$

where $\|\cdot\|_*$ is the nuclear norm of a matrix, i.e., the sum of all singular value of the matrix C, which can approximate the rank of the matrix, E represents the noise introduced by the model error and $\|E\|_{2,1}=\sum_{j=1}^{n}\sqrt{\sum_{i=1}^{m}\left(E_{ij}\right)^2}$, and λ is a tradeoff parameter. The low-rank constraint term preserves low-rank structure of C, which can be used to measure the relative affinities between data points that characterize the global structure of data X.

2.2 Semi-supervised subspace clustering

Recently, semi-supervised learning is incorporated into LRR framework to capture both the global structure and the local structure of the whole data for clustering. Label propagation[26] is a representative graph-based SSL algorithm that propagates the label information from labeled data to unlabeled ones through the pairwise affinity between data points. The unknown labels should be consistent with both the given labels and the pairwise affinities. The unified framework for semi-supervised clustering integrates affinity construction and label propagation together.

The given dataset is denoted as $X=[X_\mathrm{l}, X_\mathrm{u}]=[x_1, x_2, \ldots, x_n]\in\mathbb{R}^{m\times n}$, where X_l and X_u represent the labeled and unlabeled datasets, respectively. The label of the given dataset is denoted as $k\in\{1, 2, \ldots, \varkappa\}$. The clustering indicator matrix $H=[H_\mathrm{l}, H_\mathrm{u}]^\mathrm{T}$ is defined to represent the corresponding label probability distribution information, where H_l and H_u are the clustering indicator submatrices for the labeled dataset X_l and unlabeled dataset X_u, respectively. For the labeled data $x_i\in X_\mathrm{l}$, the corresponding vector $H_i=[h_{i1}, h_{i2}, \ldots, h_{i\varkappa}]^\mathrm{T}\in\mathbb{R}^{\varkappa\times 1}$ with $h_{ik}=1$ indicates its class label, and all other elements are equal to 0; for the unlabeled data $x_i\in X_\mathrm{u}$, $h_{ik}\in[0, 1]$ is unknown and needs to be estimated. Let Y_l be the known label matrix, then the unified optimization framework for semi-supervised clustering can be expressed as

$$\begin{aligned}&\min_{W,H}\ \rho(X,W)+\gamma f(W,H),\\&\begin{cases}h_{ik}\geqslant 0,\sum_{k=1}^{\varkappa}h_{ik}=1,\ i=1, 2, \ldots, n;\\H_\mathrm{l}=Y_\mathrm{l}\end{cases}\end{aligned}\tag{2}$$

where γ is a tradeoff parameter and the first term $\rho(X, W)$ is an implicit or explicit function based on the specific approach for constructing the affinity matrix W with data matrix X, e.g., when LRR-model is adopted, the affinity matrix is learnt from the low-rank representation. The second term $f(W,H)$ is a disagreement measure of W with respect to the clustering indicator matrix H using LP technique. Pei et al. construct an extended propagation learning framework by assuming that for the data point x_i, if the affinity W_{ij} is larger, then the label of x_j offers more contributions to predict the label of x_i[16]. Therefore, the estimated clustering indicator vector for x_i can be

expressed as $\hat{H}_i = \sum_{j\neq i} P_{ij}H_j$, where $P_{ij} = W_{ij}/\sum_{j\neq i} W_{ij}$, $P_{ii} = 0$. P_{ij} is called the propagation coefficient and P is the propagation matrix. Then the framework for LP can be formulated as

$$f(P,H) = \sum_{i=1}^{n} \| \sum_{j\neq i} P_{ij}H_j - H_i\|^2 = \|H - PH\|_{\mathrm{F}}^2 \tag{3}$$

where $\|\cdot\|_{\mathrm{F}}$ represents the Frobenius norm. Since the propagation coefficient P_{ij} is a normalized version of the affinity coefficient W_{ij}, in the rest of this paper, we treat the propatation matrix P as the affinity matrix.

The key principle of Eq. (3) is to use the labeled data points to guide the label estimation for unlabeled data points through the propagation mechanism that leverages the similarity and reconstruction relationships among all data points.

3 Soft-MFA Regularized Semi-Supervised Subspace Clustering

In this section, we first propose a generalized model by constructing an intrinsic graph and a penalty graph to characterize the intraclass compactness and interclass separation of data. This model is generalized to all data, not just the labeled ones. Then, a unified framework that jointly considers data projection, low-rank representation, and label propagation is proposed.

3.1 Soft-MFA-based regularization term

To preserve local geometric structures embedded in a high-dimensional space, several manifold learning based regularizers have been imposed into low-rank representation[6, 12, 13]. Manifold embedding aims at describing each vertex of the graph by a low-dimensional vector while preserves affinity between the vertices. Most existing methods consider only the intraclass affinities, and the interclass discrimination is often overlooked. Therefore, it is natural to consider both types of geometrical information to efficiently discover local structures of data.

Inspired by the graph embedding based marginal Fisher analysis[27], we first define two graphs, the intrinsic graph and the penalty graph, to characterize the similarities of data within classes and the discrimination of data between classes, respectively, as shown in Fig. 1. According to the learned affinity denoted as P, the intrinsic graph gives the intraclass data adjacency relationship and the corresponding affinity matrix is denoted as P^+ that only retains the similarities for intraclass data in P and enforces the interclass similarities to 0; P^- is generated from P by keeping the similarities between data points in different classes and setting the intraclass similarities to 0, note that $P = P^+ + P^-$. To emphasize the interclass separability, we define the penalty graph as $P^{+\!\!\!-} = \tilde{\mathbf{1}} - P^-$ and $\tilde{\mathbf{1}} \in \mathbb{R}^{N\times N}$ denotes a matrix in which the corresponding elements of different classes in P are set to 1 and the rest are zeroed, as shown in Fig. 1. Then we define the intraclass compactness based on the intrinsic graph in the embedding space as

$$\begin{aligned}\tilde{S}_{\mathrm{Ho}} = \sum_{\substack{i \\ x_i,x_j\in \mathrm{C_o}}} \sum_{j} P_{i,j}^{+} \left\|A^{\mathrm{T}}x_i - A^{\mathrm{T}}x_j\right\|_{\mathrm{F}}^2 = \\ 2\,\mathrm{tr}\left[A^{\mathrm{T}}X(D_{P^+} - P^+)X^{\mathrm{T}}A\right] = \\ \mathrm{tr}\left[A^{\mathrm{T}}XL_{P^+}X^{\mathrm{T}}A\right]\end{aligned} \tag{4}$$

where $P_{i,j}^{+}$ represents the $\{i,j\}$-th element of matrix P^+, $x_i, x_j \in \mathrm{C_o}$ denotes that x_i and x_j are in the same class, A is the projection matrix, $D_{P^+} = \mathrm{diag}(P^+)$, and

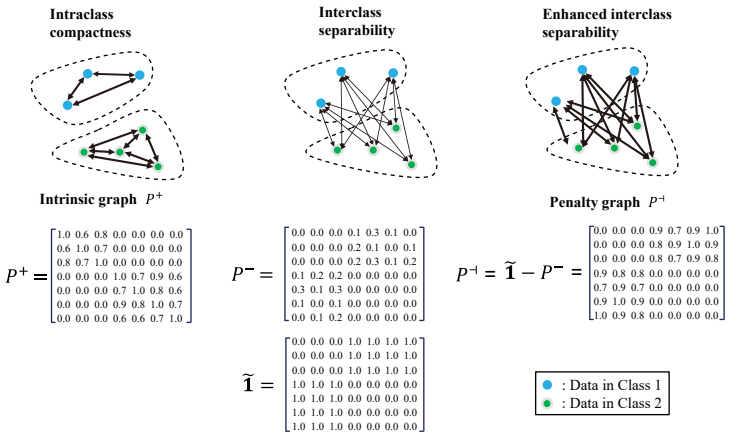


Fig. 1 Adjacency relationship of the proposed method with preserved intra class similarity and enhanced inter class discrimination. Note that the heavier edges denote higher affinities.

$L_{P^+} = D_{P^+} - P^+$. Meanwhile, interclass separability defined on the penalty graph can be expressed as

$$\tilde{S}_{\mathrm{InHo}} = \sum_{\substack{i \\ x_i, x_j \notin \mathrm{C_o}}} \sum_{j} \left(1 - P^-_{i,j}\right) \left\| A^{\mathrm{T}} x_i - A^{\mathrm{T}} x_j \right\|_{\mathrm{F}}^2 = 2\,\mathrm{tr}\left[A^{\mathrm{T}} X (D_{P^-} - P^-) X^{\mathrm{T}} A\right] = \mathrm{tr}\left[A^{\mathrm{T}} X L_{P^-} X^{\mathrm{T}} A\right] \tag{5}$$

where $P^-_{i,j}$ represents the $\{i, j\}$-th element of matrix P^-, $x_i, x_j \notin \mathrm{C_o}$ indicates that x_i and x_j are from different classes, $D_{P^-} = \mathrm{diag}(P^-)$, and $L_{P^-} = D_{P^-} - P^-$.

If the intraclass data are more compact and interclass data are more dispersed in the new space, it will be more beneficial for LRR to find the proper representation. Therefore, we propose a generalized regularization term as

$$\tilde{S}_{\mathrm{Ho}} - \eta \tilde{S}_{\mathrm{InHo}} \tag{6}$$

where η is the tradeoff parameter. We aim for the data in the original space that are in the same class to be closer in the low-dimensional space after projection, which means $\tilde{S}_{\mathrm{Ho}}$ the smaller the better. Conversely, for data points in the original space considered to be in different classes, even if they have high similarity, we want these data points to become more separated in the new space after projection, then $\tilde{S}_{\mathrm{InHo}}$ the larger the better. Therefore, $\tilde{S}_{\mathrm{Ho}} - \eta \tilde{S}_{\mathrm{InHo}}$ is a reasonable regularization term to characterize both intraclass similarities and interclass discrimination.

Note that, compared to the MFA method[27], which employs a hard binary affinity matrix, our proposed regularization term uses soft affinities. This allows for more nuanced similarity measures. Additionally, while the traditional MFA method focuses only on labeled data points, our approach integrates both labeled and unlabeled ones, ensuring that the intrinsic graph and penalty graph encompass all data points.

3.2 Model of SMFA-SSC

Now, we integrate the low-rank representation, label propagation, and the proposed regularization term into a unified optimization framework in the embedding subspace for semi-supervised subspace clustering as

$$\begin{aligned} \min_{P,H,C,E,A} \; & \|H - PH\|_{\mathrm{F}}^2 + \gamma \|C\|_* + \beta \|E\|_{2,1} + \\ & \alpha\left(\tilde{S}_{\mathrm{Ho}} - \eta \tilde{S}_{\mathrm{InHo}}\right), \\ \text{s.t.,} \quad & A^{\mathrm{T}} X = A^{\mathrm{T}} X C + E, A^{\mathrm{T}} A = I, C \geqslant 0, \\ & P = \mathrm{diag}^{-1}\left(C^{\mathrm{T}} \mathbf{1}\right) C^{\mathrm{T}}, \mathrm{diag}(P) = 0, \\ & H_{\mathrm{l}} = Y_{\mathrm{l}}, H \geqslant 0,\ H^{\mathrm{T}} H = I \end{aligned} \tag{7}$$

where α, β, γ, and η are trade off parameters, I is the unit matrix, and $\mathbf{1}$ is the all-one vector. For simplification, we term this formulation as SMFA-SSC. In this objective function, we let the clustering indicator matrix be nonnegative and orthogonal, i.e., $H > 0$ and $H^{\mathrm{T}} H = I$, to ensure that the matrix has exactly one entry in each column greater than 0 and all other entries are zero. The propagation coefficient matrix $P = \mathrm{diag}^{-1}(C^{\mathrm{T}} \mathbf{1}) C^{\mathrm{T}}$ is the normalization of the affinity matrix C, and in the rest of this paper, we will treat P as the affinity matrix.

Note that, rather than learning the representation and clustering separately, we propose to embed the data, learn the data representation, and cluster the data into corresponding subspaces simultaneously to make the learned data representation more suitable for the clustering. Furthermore, the low-rankness criterion in Eq. (7) is capable of capturing the global structure of data X, while the proposed regularization term can capture the local and discriminative structure of the data.

The objective function in Eq. (7) is nonlinear and the optimization problem is constrained. Hence there generally does not exist a closed-form solution. By considering it as a multiobjective optimization problem, we divide Eq. (7) into three subproblems as follows:

(1) Subproblem of C and E (robust LRR model learning in embedding subspace): Given the other variables, we aim to learn the low-rank representation models C and E on the data points, which are in a low-dimensional space with less redundant features and noises. The reduced problem can be formulated as

$$\begin{aligned} \min_{C,E} \quad & \gamma \|C\|_* + \beta \|E\|_{2,1}, \\ \text{s.t.,} \quad & P = \mathrm{diag}^{-1}\left(C^{\mathrm{T}} \mathbf{1}\right) C^{\mathrm{T}}, \mathrm{diag}(P) = 0, \\ & A^{\mathrm{T}} X = A^{\mathrm{T}} X C + E, C \geqslant 0 \end{aligned} \tag{8}$$

(2) Subproblem of P and H (adaptive label propagation): As the LRR models C and E and the projection matrix A are known, we can focus on optimizing the affinity matrix P and label indicator matrix H simultaneously. By substituting $\tilde{S}_{\mathrm{Ho}}$ and $\tilde{S}_{\mathrm{InHo}}$ defined in Eqs. (4) and (5) into Eq. (7), we obtain the following adaptive problem to propagate the label information from labeled to unlabeled data

$$\min_{P,H} \quad \|H - PH\|_{\mathrm{F}}^2 + \alpha\left(\mathrm{tr}\left[A^{\mathrm{T}} X L_{P^+} X^{\mathrm{T}} A\right] -\right.$$

$$\eta \operatorname{tr}\left[A^{\mathrm{T}} X L_{P^{+}} X^{\mathrm{T}} A\right]\Big),$$
$$\text{s.t.,}\quad P=\operatorname{diag}^{-1}\left(C^{\mathrm{T}}1\right)C^{\mathrm{T}}, \operatorname{diag}(P)=0,$$
$$P=P^{+}+P^{-}, P^{+}\geqslant 0, P^{-}\geqslant 0,$$
$$H\geqslant 0, H^{\mathrm{T}}H=I, H_1=Y_1 \tag{9}$$

(3) Subproblem of A and E (data embedding and self-expressiveness error learning): In this step, given the affinity matrix P and the label indicator matrix H, we aim to learn a robust projection matrix A and the self-expressiveness model. To make the problem computationally tractable, we assume that the self-representation coefficient matrix C is fixed. The problem then reduces to learning the projection matrix A and the expression error E simultaneously as

$$\min_{A,E}\ \beta\|E\|_{2,1}+\alpha\left(\operatorname{tr}\left[A^{\mathrm{T}} X L_{P^{+}} X^{\mathrm{T}} A\right]-\eta \operatorname{tr}\left[A^{\mathrm{T}} X L_{P^{+}} X^{\mathrm{T}} A\right]\right),$$
$$\text{s.t.,}\quad A^{\mathrm{T}}A=I, A^{\mathrm{T}}X=A^{\mathrm{T}}XC+E \tag{10}$$

3.3 Iterative procedure for optimization

The subproblems can be optimized sequentially and iteratively until convergence is achieved. The procedures for these subproblems are detailed in this section.

3.3.1 Subproblem of *C* and *E*

To make Eq. (8) separable for optimization, we introduce some extra variables, $Z\triangleq C$, $d\triangleq C^{\mathrm{T}}\mathbf{1}$, and $D\triangleq \operatorname{diag}(d)$, and apply the alternating direction method of multiplier (ADMM)[28] to optimize the problem. By alternating between optimizing different variables and updating dual variables, ADMM ensures that the constraints are respected while steadily improving the solution at each step. Thus, Eq. (8) becomes

$$\min_{C\geqslant 0,E,Z}\ \gamma\|Z\|_{*}+\beta\|E\|_{2,1},$$
$$\text{s.t.,}\quad P=D^{-1}C^{\mathrm{T}},$$
$$A^{\mathrm{T}}X=A^{\mathrm{T}}XC+E,$$
$$d=C^{\mathrm{T}}\mathbf{1}, C=Z \tag{11}$$

The augmented Lagrange function of Eq. (11) is

$$\min_{\substack{C\geqslant 0,E,Z,d,\\ \mu_1,\mu_2,\mu_3,\mu_4}}\ \gamma\|Z\|_{*}+\beta\|E\|_{2,1}+\left\langle\mu_1, P-D^{-1}C^{\mathrm{T}}\right\rangle+\left\langle\mu_2, A^{\mathrm{T}}X-A^{\mathrm{T}}XC-E\right\rangle+\left\langle\mu_3, d-C^{\mathrm{T}}\mathbf{1}\right\rangle+\langle\mu_4, C-Z\rangle+\frac{\rho_1}{2}\left(\left\|P-D^{-1}C^{\mathrm{T}}\right\|_{\mathrm{F}}^{2}+\left\|d-C^{\mathrm{T}}\mathbf{1}\right\|_{2}^{2}+\|C-Z\|_{\mathrm{F}}^{2}+\left\|A^{\mathrm{T}}X-A^{\mathrm{T}}XC-E\right\|_{\mathrm{F}}^{2}\right) \tag{12}$$

where μ_1, μ_2, μ_3, and μ_4 are the matrices of Lagrange multipliers, ρ_1 is the condition parameter. According to the ADMM algorithm, the variables can be updated iteratively by fixing each other.

• **Update *Z*:** Given the other variables as constants, Z is updated by solving the following reduced minimization problem

$$\min_{Z}\ \frac{\gamma}{\rho_1}\|Z\|_{*}+\frac{1}{2}\|Z-(C+\wedge_4)\|_{\mathrm{F}}^{2} \tag{13}$$

where $\wedge_4=\mu_4/\rho_1$. This is a nuclear parametric minimization problem which has a closed-form solution[29]

$$Z=\mathcal{S}_{\gamma/\rho_1}(C-\wedge_4) \tag{14}$$

where $\mathcal{S}_{\gamma/\rho_1}(\cdot)$ is the singular value thresholding (SVT) operator.

• **Update *C*:** By fixing other variables, the optimization for C can be transformed to

$$\min_{C\geqslant 0}\ \frac{1}{2}\left\|P-D^{-1}C^{\mathrm{T}}-\wedge_1\right\|_{\mathrm{F}}^{2}+\frac{1}{2}\left\|d-C^{\mathrm{T}}\mathbf{1}-\wedge_3\right\|_{2}^{2}+\frac{1}{2}\left\|A^{\mathrm{T}}X-A^{\mathrm{T}}XC-E-\wedge_2\right\|_{\mathrm{F}}^{2}+\frac{1}{2}\|C-Z-\wedge_4\|_{\mathrm{F}}^{2} \tag{15}$$

where $\wedge_1=\mu_1/\rho_1$, $\wedge_2=\mu_2/\rho_1$, and $\wedge_3=\mu_3/\rho_1$. By setting the derivative of Eq. (15) with respect to C to zero, we obtain

$$(CD^{-1}-P^{\mathrm{T}}+\wedge_1^{\mathrm{T}})D^{-1}+\mathbf{1}(\mathbf{1}^{\mathrm{T}}C-d^{\mathrm{T}}+\wedge_3^{\mathrm{T}})+X^{\mathrm{T}}A(A^{\mathrm{T}}XC-A^{\mathrm{T}}X+E+\wedge_2)+C-Z-\wedge_4=0 \tag{16}$$

With some algebra, Eq. (16) can be written as follows:

$$(X^{\mathrm{T}}AA^{\mathrm{T}}X+\mathbf{1}\cdot\mathbf{1}^{\mathrm{T}}+I)C+C(D^{-2})=(P^{\mathrm{T}}-\wedge_1^{\mathrm{T}})D^{-1}+X^{\mathrm{T}}A(A^{\mathrm{T}}X-E-\wedge_2)+\mathbf{1}(d^{\mathrm{T}}-\wedge_3{}^{\mathrm{T}})+(Z+\wedge_4) \tag{17}$$

Note that Eq. (17) is a Sylvester equation with the form $AX+XB=C$, we can use the Sylvester function in MATLAB to solve this problem. For solving $C(D^{-2})$ and $(P^{\mathrm{T}}+\wedge_1{}^{\mathrm{T}})D^{-1}$, since D is an $N\times N$ matrix and N is generally large in clustering problem, we use the operator "\" in MATLAB to calculate matrix inversion.

• **Update *d*:** Given other variables fixed, d can be obtained by minimizing the following form of Eq. (12)

$$\arg\min_{d\geqslant 0}\ \frac{1}{2}\left\|P-D^{-1}C^{\mathrm{T}}-\wedge_1\right\|_{\mathrm{F}}^{2}+\frac{1}{2}\left\|d-C^{\mathrm{T}}\mathbf{1}-\wedge_3\right\|_{2}^{2} \tag{18}$$

For convenience of the calculation, we use the constant

equation $d = C^{\mathrm{T}}$ as the solution to Eq. (18).

• **Update E:** By fixing other variables, E is achieved by solving the following minimization problem

$$\min_{E} \frac{\beta}{\rho_1}\|E\|_{2,1} + \frac{1}{2}\left\|E - (A^{\mathrm{T}}X - A^{\mathrm{T}}XC - \wedge_2)\right\|_{\mathrm{F}}^2 \tag{19}$$

This equation is an $L_{2,1}$ parametric minimization problem, which can be solved as follows[25]:

$$E_{\cdot i} = \begin{cases} \dfrac{\|U_{\cdot i}\|_2 - \frac{\beta}{\rho_1}}{\|U_{\cdot i}\|_2} U_{\cdot i}, & \dfrac{\beta}{\rho_1} < \|U_{\cdot i}\|_2; \\ 0, & \dfrac{\beta}{\rho_1} \geqslant \|U_{\cdot i}\|_2 \end{cases} \tag{20}$$

where $U = A^{\mathrm{T}}X - A^{\mathrm{T}}XC - \wedge_2$, and $E_{\cdot i}$ and $U_{\cdot i}$ are the i-th column vectors of variables E and U, respectively.

• **Update μ_1, μ_2, μ_3, and μ_4:** The Lagrange multipliers at each step are updated as follows:

$$\begin{aligned} &\mu_1 \leftarrow \mu_1 + \rho_1(P - D^{-1}C^{\mathrm{T}}), \\ &\mu_2 \leftarrow \mu_2 + \rho_1(A^{\mathrm{T}}X - A^{\mathrm{T}}XC - E), \\ &\mu_3 \leftarrow \mu_3 + \rho_1(d - C^{\mathrm{T}}\mathbf{1}), \\ &\mu_4 \leftarrow \mu_4 + \rho_1(C - Z) \end{aligned} \tag{21}$$

3.3.2 Subproblem of P and H

Firstly, we introduce the variable $\theta \in \mathbb{R}^{N\times N}$ to facilitate the derivation, the entries of which satisfy

$$\theta_{ij} = \left\|A^{\mathrm{T}}x_i - A^{\mathrm{T}}x_j\right\|_{\mathrm{F}}^2 \tag{22}$$

and $\theta = \theta^+ + \theta^-$, where $\theta^+ = \left[\theta_{ij}^+\right]_{i,j=1}^N$ and the entries $\theta^+{}_{ij} = \theta_{ij}$ if x_i and x_j belong to the same class, otherwise $\theta^+{}_{ij} = 0$, meanwhile $\theta^- = \left[\theta_{ij}^-\right]_{i,j=1}^N$ and $\theta^-{}_{ij} = \theta_{ij}$ if x_i and x_j are in different classes, otherwise $\theta^-{}_{ij} = 0$. The class information can be obtained from the label indicator matrix H_t obtained at the t-th iteration, therefore, θ^+ and θ^- can be expressed as

$$\begin{aligned} &\theta^+ = \theta \odot (H_t H_t^{\mathrm{T}}), \\ &\theta^- = \theta \odot (\mathbf{1}\cdot\mathbf{1}^{\mathrm{T}} - H_t H_t^{\mathrm{T}}) \end{aligned} \tag{23}$$

where $\odot$ denotes dot product. Then we rewrite the second and third terms in Eq. (9) in matrix forms as

$$\begin{aligned} &\mathrm{tr}\left[A^{\mathrm{T}}XL_{P^+}X^{\mathrm{T}}A\right] = \\ &\sum_{\substack{i \\ x_i,x_j\in \mathrm{C_o}}}\sum_{j} P_{i,j}^+ \left\|A^{\mathrm{T}}x_i - A^{\mathrm{T}}x_j\right\|_{\mathrm{F}}^2 = \mathrm{tr}\left[(\boldsymbol{\theta}^+)^{\mathrm{T}}P^+\right] \end{aligned} \tag{24}$$

and

$$\mathrm{tr}\left[A^{\mathrm{T}}XL_{P^-}X^{\mathrm{T}}A\right] = \sum_{\substack{i \\ x_i,x_j\notin \mathrm{C_o}}}\sum_{j} \left(1 - P_{i,j}^-\right)\left\|A^{\mathrm{T}}x_i - A^{\mathrm{T}}x_j\right\|_{\mathrm{F}}^2 =$$

$$\sum_{\substack{i \\ x_i,x_j\notin \mathrm{C_o}}}\sum_{j} \theta_{i,j}^- - \sum_{\substack{i \\ x_i,x_j\notin \mathrm{C_o}}}\sum_{j} P_{i,j}^-\theta_{i,j}^- = -\mathrm{tr}\left[(\theta^-)^{\mathrm{T}}P^-\right] \tag{25}$$

Now, by introducing auxiliary variable $\mathcal{F}$ and letting $P = \mathrm{diag}^{-1}(C^{\mathrm{T}}\mathbf{1})C^{\mathrm{T}} \triangleq \mathcal{F}$, we can reformulate the objective function in Eq. (9) as follows:

$$\begin{aligned} \min_{P^+,P^-,H} \quad & \|H - PH\|_{\mathrm{F}}^2 + \alpha\left(\mathrm{tr}\left[\theta^{+\mathrm{T}}P^+\right] + \eta\,\mathrm{tr}\left[\theta^{-\mathrm{T}}P^-\right]\right), \\ \text{s.t.,} \quad & P = \mathcal{F}, P\mathbf{1} = \mathbf{1}, \\ & P = P^+ + P^-, P^+ \geqslant 0, P^- \geqslant 0, \\ & H \geqslant 0, H^{\mathrm{T}}H = I, H_1 = Y_1 \end{aligned} \tag{26}$$

Then, the corresponding augmented Lagrange function of Eq. (26) can be represented as

$$\begin{aligned} \min_{\substack{P^+,P^-,H, \\ \mu_5,\mu_6}} \quad & \|H - P^+H - P^-H\|_{\mathrm{F}}^2 + \\ & \alpha\left(\mathrm{tr}\left[\theta^{+\mathrm{T}}P^+\right] + \eta\,\mathrm{tr}\left[\theta^{-\mathrm{T}}P^-\right]\right) + \\ & \langle\mu_5, P^+ + P^- - \mathcal{F}\rangle + \\ & \langle\mu_6, P^+\mathbf{1} + P^-\mathbf{1} - \mathbf{1}\rangle + \\ & \frac{\rho_2}{2}\left\|P^+ + P^- - \mathcal{F}\right\|_{\mathrm{F}}^2 + \\ & \frac{\rho_2}{2}\left\|P^+\mathbf{1} + P^-\mathbf{1} - \mathbf{1}\right\|_2^2 \\ \text{s.t.} \quad & P^+ \geqslant 0, P^- \geqslant 0, \\ & H \geqslant 0, H^{\mathrm{T}}H = I, H_1 = Y_1 \end{aligned} \tag{27}$$

where ρ_2 is the tradeoff paramete, and μ_5 and μ_6 are Lagrange multiplier matrices. According to the ADMM algorithm, Eq. (27) can be split into multiple subproblems and solved iteratively until convergence.

• **Update P^+:** When other variables are fixed, the objective function in Eq. (27) with respect to the matrix P^+ can be rewritten as

$$\begin{aligned} &\min_{P^+\geqslant 0}\left\|H - P^+H - P^-H\right\|_{\mathrm{F}}^2 + \alpha\,\mathrm{tr}\left[\theta^{+\mathrm{T}}P^+\right] + \\ &\quad \frac{\rho_2}{2}\left\|P^+ + P^- - \mathcal{F} - \wedge_5\right\|_{\mathrm{F}}^2 + \frac{\rho_2}{2}\left\|P^+\mathbf{1} + P^-\mathbf{1} - \mathbf{1} - \wedge_6\right\|_2^2 \end{aligned} \tag{28}$$

where $\wedge_5 = \mu_5/\rho_2$ and $\wedge_6 = \mu_6/\rho_2$. Taking derivative of Eq. (28) with respect to P^+ and letting it be 0, we can update P^+ as

$$\begin{aligned} &P^+ \leftarrow (2(H - P^-H)H^{\mathrm{T}} - \alpha\theta^+ + \\ &\rho_2(\mathcal{F} - P^- + \wedge_5) + \rho_2(\mathbf{1} - P^-\mathbf{1} + \wedge_6)\mathbf{1}^{\mathrm{T}}) \cdot \\ &(2HH^{\mathrm{T}} + \rho_2 I + \rho_2\mathbf{1}\cdot\mathbf{1}^{\mathrm{T}})^{-1} \end{aligned} \tag{29}$$

To ensure that the constraint $P^+ \geqslant 0$ is satisfied, we set the elements in P^+ that are less than 0 to zero. As a result, we update P^+, such that $P^+ \leftarrow (P^+)_+$.

• **Update P^-:** For the optimization of P^-, Eq. (27)

can be expressed as

$$\min_{P^-\geqslant 0}\left\|H-P^+H-P^-H\right\|_{\mathrm{F}}^2+\alpha\eta\,\mathrm{tr}\left[\theta^{-\mathrm{T}}P^-\right]+\frac{\rho_2}{2}\left\|P^++P^--\mathcal{F}-\wedge_5\right\|_{\mathrm{F}}^2+\frac{\rho_2}{2}\left\|P^+\mathbf{1}+P^-\mathbf{1}-\mathbf{1}-\wedge_6\right\|_2^2 \tag{30}$$

Similarly, by setting the derivative of Eq. (30) with respect to P^- to zero, we can update P^- as

$$P^-\leftarrow(2(H-P^+H)H^{\mathrm{T}}-\alpha\eta\theta^-+\rho_2(\mathcal{F}-P^++\wedge_5)+\rho_2(\mathbf{1}-P^+\mathbf{1}+\wedge_6)\mathbf{1}^{\mathrm{T}})\times(2HH^{\mathrm{T}}+\rho_2 I+\rho_2\mathbf{1}\cdot\mathbf{1}^{\mathrm{T}})^{-1} \tag{31}$$

We enforce the numbers less than 0 in P^- to be 0 to meet the constraint that $P^-\geqslant 0$ as $P^-\leftarrow(P^-)_+$.

After P^+ and P^- are updated, we can compute P as

$$P=P^++P^- \tag{32}$$

Then we normalize P to satisfy $P\mathbf{1}=\mathbf{1}$.

• **Update H:** When P is obtained, we update H. Since $\|H-PH\|_{\mathrm{F}}^2=\mathrm{tr}(H^{\mathrm{T}}(I-P)^{\mathrm{T}}(I-P)H)$, the objective function of Eq. (27) can be rewritten as

$$\min_H\ \mathrm{tr}(H^{\mathrm{T}}(I-P)^{\mathrm{T}}(I-P)H),\quad \text{s.t.},\ H\geqslant 0, H^{\mathrm{T}}H=I, H_1=Y_1 \tag{33}$$

This is a typical trace minimization problem[30], and the optimal H is formed by eigenvectors of $(I-P)^{\mathrm{T}}(I-P)$ corresponding to the first $\mathcal{N}$ smallest eigenvalues.

• **Update μ_5 and μ_6:** After P^+, P^-, and H are computed, we can update $\wedge_5$ and $\wedge_6$ as

$$\mu_5\leftarrow\mu_5+\rho_2(P^++P^--F),\quad \mu_6\leftarrow\mu_6+\rho_2(P^+\mathbf{1}+P^-\mathbf{1}-\mathbf{1}) \tag{34}$$

3.3.3 Subproblem of A and E

Write Eq. (10) in augmented Lagrangian function form as follows:

$$\min_{\substack{A,E\\ A^{\mathrm{T}}A=I,\mu_7}}\alpha\left(\mathrm{tr}\left[A^{\mathrm{T}}XL_{P^+}X^{\mathrm{T}}A\right]-\eta\,\mathrm{tr}\left[A^{\mathrm{T}}XL_{P^-}X^{\mathrm{T}}A\right]\right)+<\mu_7,A^{\mathrm{T}}X-A^{\mathrm{T}}XC-E>+\frac{\rho_3}{2}\left\|A^{\mathrm{T}}X-A^{\mathrm{T}}XC-E\right\|_{\mathrm{F}}^2+\beta\|E\|_{2,1} \tag{35}$$

where ρ_3 is the tradeoff parameter, and μ_7 is the Lagrange multiplier matrix. The optimization strategy is the same as that employed in the optimization of P and E, Eq. (35) is split into several subproblems which are solved iteratively until convergence.

• **Update E:** By fixing other variables, E is updated by solving the following objective function

$$\min_E\frac{\beta}{\rho_3}\|E\|_{2,1}+\frac{1}{2}\left\|E-(A^{\mathrm{T}}X-A^{\mathrm{T}}XC-\wedge_7)\right\|_{\mathrm{F}}^2 \tag{36}$$

where $\wedge_7=\eta/\rho_3$. This is the $L_{2,1}$ parametric minimization problem, which can be solved in the same way as solving Eq. (19).

• **Update A:** With other variables fixed, A is then updated by solving the following problem

$$\min_{A,A^{\mathrm{T}}A=I}\alpha\left(\mathrm{tr}\left[A^{\mathrm{T}}XL_{P^+}X^{\mathrm{T}}A\right]-\eta\,\mathrm{tr}\left[A^{\mathrm{T}}XL_{P^-}X^{\mathrm{T}}A\right]\right)+\frac{\rho_3}{2}\left\|A^{\mathrm{T}}X-A^{\mathrm{T}}XC-E-\wedge_7\right\|_{\mathrm{F}}^2 \tag{37}$$

Since $A^{\mathrm{T}}A=I$, the second term in Eq. (37) can be rewritten as

$$\begin{aligned}&\left\|A^{\mathrm{T}}X-A^{\mathrm{T}}XC-E-\wedge_7\right\|_{\mathrm{F}}^2=\\&\left\|A^{\mathrm{T}}X-A^{\mathrm{T}}XC-A^{\mathrm{T}}AE-A^{\mathrm{T}}A\wedge_7\right\|_{\mathrm{F}}^2=\\&\left\|A^{\mathrm{T}}(X-XC-AE-A\wedge_7)\right\|_{\mathrm{F}}^2=\\&\mathrm{tr}[A^{\mathrm{T}}(X-XC-AE-A\wedge_7)\times\\&(X-XC-AE-A\wedge_7)^{\mathrm{T}}A]\end{aligned} \tag{38}$$

Let $G\triangleq X-XC-AE-A\wedge_7$, Eq. (37) can be written as

$$\min_{A,A^{\mathrm{T}}A=I}\alpha(\mathrm{tr}[A^{\mathrm{T}}XL_{P^+}X^{\mathrm{T}}A]-\eta\,\mathrm{tr}[A^{\mathrm{T}}XL_{P^-}X^{\mathrm{T}}A])+\frac{\rho_3}{2}\mathrm{tr}[A^{\mathrm{T}}GG^{\mathrm{T}}A] \tag{39}$$

and Eq. (39) equals to[31]

$$\min_{A^{\mathrm{T}}A=I}\frac{\mathrm{tr}\left[A^{\mathrm{T}}\left(\alpha XL_{P^+}X^{\mathrm{T}}+\frac{\rho_3}{2}GG^{\mathrm{T}}\right)A\right]}{\alpha\eta\,\mathrm{tr}\left[A^{\mathrm{T}}XL_{P^-}X^{\mathrm{T}}A\right]} \tag{40}$$

The solution to Eq. (40) can be written as[32]

$$(XL_{P^-}X^{\mathrm{T}})^{-1}\left(\alpha XL_{P^+}X^{\mathrm{T}}+\frac{\rho_3}{2}GG^{\mathrm{T}}\right)A=\lambda_A\cdot A \tag{41}$$

where λ_A represents the eigenvalue of matrix $(XL_{P^-}X^{\mathrm{T}})^{-1}\left(\alpha XL_{P^+}X^{\mathrm{T}}+\frac{\rho_3}{2}GG^{\mathrm{T}}\right)$.

Then we select b eigenvectors $\{A_1, A_2, \ldots, A_b\}$ corresponding to the b smallest eigenvalues to form the projection matrix A as the solution of Eq. (41), where b needs to be adjusted according to specific problems.

• **Update μ_7:** After A and E are computed, the Lagrange multiplier μ_7 can be solved easily as

$$\mu_7\leftarrow\mu_7+\rho_3(A^{\mathrm{T}}X-A^{\mathrm{T}}XC-E) \tag{42}$$

So far, the algorithm variables in Eq. (7) are solved. We summarize the proposed SMFA-SSC method in Algorithm 1. Algorithm 1 is a nested loop. We use variable $O=\{A,E,C,P,H\}$ to represent the set of all

Algorithm 1 SMFA-SSC

Input: X, Y_L, α, β, γ, and η.
Initialize: $i = 0$.
1: while not converged do
2: Initialize: $t = 0$.
3: while not converged do
4: Update Z, C, and E by solving Eqs. (14), (17), and (20);
5: Check convergence conditions:
6: $\max \|C_t - Z_t\|_\infty < 10^{-8}$ and
7: $\max \|AX - AXC_t - E_t\|_\infty < 10^{-8}$
8: Update t: $t \leftarrow t + 1$
9: end while
10: Initialize: $t = 0$.
11: while not converged do
12: Update P and H by solving Eqs. (32) and (33);
13: Check convergence conditions:
14: $\max \|P_t - P_{t-1}\|_F / \|P_{t-1}\|_F < 10^{-8}$ and
15: $\max \|H_t - H_{t-1}\|_F / \|H_{t-1}\|_F < 10^{-8}$
16: Update t: $t \leftarrow t + 1$
17: end while
18: Initialize: $t = 0$.
19: while not converged do
20: Update A and E by solving Eqs. (36) and (41);
21: Check convergence conditions:
22: $\max \|A_t - A_{t-1}\|_F / \|A_{t-1}\|_F < 10^{-8}$ and
23: $\max \|E_t - E_{t-1}\|_F / \|E_{t-1}\|_F < 10^{-8}$
24: Update t: $t \leftarrow t + 1$
25: end while
26: Check convergence conditions:
27: $\|O_i - O_{i-1}\|_F / \|O_{i-1}\|_F < 10^{-6}, O = \{A, E, C, P, H\}$
28: Update i: $i \leftarrow i + 1$
29: end while
Output: clustering results

optimization variables A, E, C, P, and H. The variable i is the loop variable of the outer loop, and O_i represents the variable at the i-th loop iteration. The variable t is the loop variable of the inner loop. The inner loop consists of three independent loops, which respectively correspond to the three sub-optimization problems metioned above: optimizing $\{C, E\}$, $\{P, H\}$, and $\{A, E\}$. Here, $(\cdot)t$ represents the variable at the t-th loop iteration.

4 Convergence Analysis

The optimization problem in Eq. (7) is notably complex, and for the purpose of analyzing its convergence, we use the first subproblem as an illustrative example. The subproblem of C and E mainly involves optimizing low-rank representation in the low-dimensional space. The key is the optimization formula in Eq. (1), and here we demonstrate the convergence of Eq. (1). The augmented Lagrangian of Eq. (1) can be expressed as

$$L_s(C, E, \Lambda) = f(C) + g(E) + \langle \Lambda, XC + E - X \rangle + \frac{1}{2s}\|XC + E - X\|_F^2 \tag{43}$$

where s is the step size, $f(C) = \|C\|_*$, $g(E) = \|E\|_{2,1}$, and Λ is the multiplier matrix. The ADMM iteration consists of updating C, E, and the multiplier Λ according to

$$\begin{cases} C_{k+1} = \arg\min\limits_{C \in \mathbb{R}^{N \times N}} \left\{ f(C) + \dfrac{1}{2s}\|XC + E_k - X + s\Lambda_k\|_F^2 \right\}, \\ E_{k+1} = \arg\min\limits_{E \in \mathbb{R}^{m \times N}} \left\{ g(E) + \dfrac{1}{2s}\|XC_{k+1} + E - X + s\Lambda_k\|_F^2 \right\}, \\ \Lambda_{k+1} = \Lambda_k + \dfrac{1}{s}(XC_{k+1} + E_{k+1} - X). \end{cases}$$

where C_k, E_k, and Λ_k are the k-th iteration sequences of C, E, and Λ. Here, the high-resolution ordinary differential equations are employed to characterize the continuous behavior of the ADMM algorithm[33, 34], where the discrete iterative process is reformulated as a continuous dynamical system

$$\begin{cases} X^T \dot{E} = X^T \Lambda + \nabla f(C), \\ 0 = \Lambda + \nabla g(E), \\ s^2 \dot{\Lambda} = XC + E - X \end{cases} \tag{44}$$

where $\nabla f(C)$ is the subgradient of $f(C)$ and $\nabla g(E)$ represents the subgradient of $g(E)$. Both $f(C)$ and $g(E)$ are convex but non-smooth, however, their subgradients exist. To analyze the convergence of ADMM, we construct the discrete Lyapunov function as[35]

$$\epsilon(k) = \frac{1}{2s}\|E_k - E\|_F^2 + \frac{s}{2}\|\Lambda_k - \Lambda\|_F^2 \tag{45}$$

Lemma 1 Let both $f(C)$ and $g(E)$ be convex functions. Then, the discrete Lyapunov function in Eq. (45) satisfies

$$\begin{aligned} &\epsilon(k+1) - \epsilon(k) \leqslant \\ &f(C) - f(C_{k+1}) + g(E) - g(E_{k+1}) + \\ &\left\langle \Lambda_{k+1} - \frac{(E_{k+1} - E_k)}{s}, X(C - C^*) + (E - E^*) \right\rangle - \\ &\langle \Lambda, X(C_{k+1} - C^*) + (E_{k+1} - E^*) \rangle \\ &\underbrace{-\frac{1}{2s}\|(E_{k+1} - E_k)\|_F^2 - \frac{s}{2}\|\Lambda_{k+1} - \Lambda_k\|_F^2}_{\text{NE}} \end{aligned} \tag{46}$$

where (C^*, E^*) is an equilibrium, and NE represents the numerical error resulting from the optimization process.

Proof We can write the iterative difference through Eq. (45) as

$$\begin{aligned}&\epsilon(k+1)-\epsilon(k)=\Gamma-\mathrm{NE},\\&\Gamma=\frac{1}{s}\langle(E_{k+1}-E_k),(E_{k+1}-E)\rangle+\\&\quad s\langle\Lambda_{k+1}-\Lambda_k,\Lambda_{k+1}-\Lambda\rangle,\\&\mathrm{NE}=\frac{1}{2s}\|(E_{k+1}-E_k)\|_{\mathrm{F}}^2-\frac{s}{2}\|\Lambda_{k+1}-\Lambda_k\|_{\mathrm{F}}^2,\end{aligned}$$

where Γ denotes the quantity derived from the continuous analysis approach, and NE signifies the computational error that emanates from the implicit discretization process. By employing the third iteration step of the ADMM algorithm as outlined in Eq. (44), we can recast Γ as follows:

$$\begin{aligned}&\Gamma=\frac{1}{s}\langle(E_{k+1}-E_k),(E_{k+1}-E)\rangle+\\&\langle X(C_{k+1}-C^*)+(E_{k+1}-E^*),\Lambda_{k+1}-\Lambda\rangle\end{aligned}\tag{47}$$

Nuclear norm and $L_{2,1}$ norm are both convex functions, and their subgradients exist. Hence, we can represent the first two optimization equations in Eq. (44) as

$$\begin{cases}s\partial f(C_{k+1})+X^{\mathrm{T}}(XC_{k+1}+E_k-X+s\Lambda_k)\ni 0,\\ s\partial g(E_{k+1})+(XC_{k+1}+E_{k+1}-X+s\Lambda_k)\ni 0\end{cases}\tag{48}$$

By utilizing the third iteration of Eq. (44), we have

$$\begin{cases}\partial f(C_{k+1})-X^{\mathrm{T}}\cdot\dfrac{E_{k+1}-E_k}{s}+X^{\mathrm{T}}\Lambda_{k+1}\ni 0,\\ \partial g(E_{k+1})+\Lambda_{k+1}\ni 0\end{cases}\tag{49}$$

Based on the properties of convex functions, we can derive that

$$f(C)-f(C_{k+1})\geqslant\left\langle X^{\mathrm{T}}\cdot\frac{E_{k+1}-E_k}{s}-X^{\mathrm{T}}\Lambda_{k+1},C-C_{k+1}\right\rangle\tag{50}$$

and

$$g(E)-g(E_{k+1})\geqslant\langle-\Lambda_{k+1},E-E_{k+1}\rangle\tag{51}$$

By adding Eqs. (50) and (51) into Eq. (47), we obtain

$$\begin{aligned}&\Gamma\leqslant f(C)-f(C_{k+1})+g(E)-g(E_{k+1})+\\&\left\langle\Lambda_{k+1}-\frac{(E_{k+1}-E_k)}{s},X(C-C^*)+(E-E^*)\right\rangle-\\&\langle\Lambda,X(C_{k+1}-C^*)+(E_{k+1}-E^*)\rangle+\\&\langle E_{k+1}-E_k,\Lambda_{k+1}-\Lambda_k\rangle\end{aligned}\tag{52}$$

where the last term $\langle E_{k+1}-E_k,\Lambda_{k+1}-\Lambda_k\rangle\leqslant 0$ always holds due to the convex property. Therefore, the proof is complete. ■

In optimization problem in Eq. (11), linear constraints are incorporated compared to Eq. (1), thus enabling a convergence analysis similar to previous derivations. Due to space constraints, we omit this derivation.

5 Experiment

In this section, a series of experiments are performed to demonstrate the efficacy of our proposed approach SMFA-SSC.

5.1 Dataset and experimental setting

(1) Dataset: For comparison, five different datasets are adopted in the experiments, i.e., four benchmark face recognition datasets Yale B[36], ORL[37], AR[38], and PIE[39], and one palmprint image dataset PALM[40]. A summary information of the dataset is shown in Table 1.

(2) Compared method: We compare with several state-of-the-art two-stage SSC approaches, namely LGC[8], GFHF[9], flexible manifold embedding (FME)[41], semi-supervised elastic embedding (SEE)[42], adaptive semi-supervised learning (ASL)[43], and correntropy based semi-supervised NMF (CSNMF)[44], as well as five unified frameworks, i.e., SSC-novel graph construction (SSC-NGC)[3], non-negative LRR (NNLRR)[5], S^2LRR[15], S^3R[15], and JSREPL[16].

(3) Evaluation criterion: The semi-supervised subspace clustering problem is to correctly assign unlabeled data to their corresponding clusters. To intuitively evaluate the performance of model, we use clustering accuracy (v_{ACC}) as the evaluation metric[5], which is defined as

$$v_{\mathrm{ACC}}=\frac{N_{\mathrm{c}}}{N}\tag{53}$$

where N is the number of total data points, and N_{c} is the number of accurately clustered data points.

In comparison, the LGC, GFHF, FME, SEE, ASL, S^2LRR, and S^3R algorithms all use the code given by

Table 1 Experimental dataset.

Dataset	Type	Dimension	Sample	Class
Yale B[36]	Face image	1024	1792	28
ORL[37]	Face image	1024	400	40
AR[38]	Face image	1024	1680	120
PIE[39]	Face image	1024	1700	10
PALM[40]	Palmprint image	256	2000	100

the authors of the paper, and we adjust the parameters of these algorithms to get their best results. The experimental results of NNLRR, CSNMF, SSC-NGC, and JSREPL algorithms are directly cited from the corresponding works.

5.2 Comparison with baseline method

For each experiment, we conduct experiments under different labeled ratios and record the average clustering accuracy (ACC) and standard deviation (STD). As the number of labeled data points increases, the semi-supervised methods generally show better performances. Therefore, we implement experiments under different labeled ratios. We take $\{10\%, 20\%, 30\%\}$ data of each subject in the Yale B, AR, ORL, and PIE datasets as the training set and the rest is used as the test set. For PALM dataset, we use 1 labeled, 2 labeled, and 3 labeled to represent the selection of 1, 2, and 3 sample points respectively as training data in the dataset. Experimental results are reported in Table 2–5 in the form of ACC%$\pm$STD%, and the clustering results with the best performance under each dataset are marked in bold. Then, we can draw conclusions as follows:

(1) The semi-supervised subspace clustering algorithms using the unified optimization framework, i.e., S^2LRR, S^3R, JSREPL, SSC-NGC, and the proposed method SMFA-SSC, show significant advantages over the two-stage methods in most cases, especially when there are fewer labeled data.

(2) In general, incorporating labeled data into a model as priors enhances the clustering process, thereby increasing the accuracy rate as more labeled samples are added. Conversely, the clustering task becomes increasingly challenging when the number of labeled data decreases. Notably, compared to all the other algorithms, SMFA-SSC method achieves the highest v_{ACC} on all databases (except for ORL

Table 2 Clustering performance (ACC% $\pm$ STD%) with different methods under 10% labeled ratio.

Method	Yale B	ORL	AR	PIE
LGC[8]	58.16±2.14	69.44±2.14	61.72±0.84	78.01±2.28
GFHF[9]	52.26±2.20	65.14±2.31	61.36±1.65	77.94±2.13
FME[41]	74.46±1.46	81.03±2.22	65.35±1.97	96.33±0.53
SEE[42]	70.09±1.23	68.78±2.16	83.35±0.61	93.52±0.35
ASL[43]	83.23±2.13	64.64±2.97	78.50±1.22	97.02±0.90
CSNMF[44]	45.42±0.00	67.73±0.00	–	–
SSC-NGC[3]	86.40±0.61	**90.95 ± 3.43**	94.65±3.26	93.62±3.53
NNLRR[5]	94.44±0.59	–	–	87.78±2.91
S^2LRR[15]	74.37±2.67	72.28±3.55	90.35±2.36	92.83±1.82
S^3R[15]	90.96±0.96	84.50±2.82	94.11±1.06	95.64±0.88
JSREPL[16]	86.35±0.00	74.00±0.00	–	83.64±0.00
SMFA-SSC	**94.48 ± 0.57**	85.00 ± 1.62	**95.14 ± 1.45**	**97.11 ± 0.91**

Table 3 Clustering performance (ACC% $\pm$ STD%) with different methods under 20% labeled ratio.

Method	Yale B	ORL	AR	PIE
LGC[8]	63.53±1.22	79.94±2.39	70.30±2.03	86.34±1.12
GFHF[9]	60.63±0.99	72.97±1.69	69.71±1.55	87.51±1.52
FME[41]	85.43±0.97	89.53±2.03	72.27±1.16	96.85±0.64
SEE[42]	77.65±1.04	79.00±2.79	88.27±0.83	96.29±0.79
ASL[43]	90.93±1.83	81.37±1.41	85.37±1.75	**97.81 ± 0.37**
CSNMF[44]	–	67.73±0.00	–	–
SSC-NGC[3]	–	–	–	–
NNLRR[5]	94.69±0.87	–	–	89.37±2.42
S^2LRR[15]	81.89±1.83	81.96±2.02	92.86±1.75	94.34±1.65
S^3R[15]	93.59±0.69	**92.16 ± 1.82**	97.35±0.74	97.57±0.35
JSREPL[16]	87.07±0.00	82.33±0.00	–	90.19±0.00
SMFA-SSC	**94.75 ± 0.38**	91.87±1.05	**97.81 ± 0.66**	97.23±0.65

Table 5 Clustering performance (ACC% ± STD%) on PALM dataset.

Method	1 labeled	2 labeled	3 labeled
LGC[8]	92.74±0.66	96.34±0.79	98.74±0.53
GFHF[9]	91.61±0.96	95.11±0.80	97.38±0.95
FME[41]	99.06±0.37	99.68±0.18	99.69±0.29
SEE[42]	96.25±0.67	98.24±0.59	99.04±0.42
ASL[43]	98.28±0.74	99.37±0.62	99.38±0.42
S^2LRR[15]	95.14±1.82	97.88±1.65	98.31±1.79
S^3R[15]	79.95±1.95	89.64±1.01	95.10±0.54
SMFA-SSC	100.00±0.00	100.00±0.00	100.00±0.00

Table 4 Clustering performance (ACC% ± STD%) with different methods under 30% labeled ratio.

Method	Yale B	ORL	AR	PIE
LGC[8]	66.66±1.62	84.21±3.12	75.33±1.46	90.10±1.35
GFHF[9]	65.50±1.63	82.00±3.16	73.63±1.15	91.84±1.17
FME[41]	88.72±1.53	91.54±1.54	78.01±1.36	97.59±0.85
SEE[42]	83.26±1.05	81.71±3.18	91.57±0.85	97.60±0.41
ASL[43]	93.34±0.62	88.00±1.58	89.47±1.33	98.08±0.34
CSNMF[44]	–	–	–	–
SSC-NGC[3]	–	–	–	–
NNLRR[5]	95.71±1.17	–	–	90.18±1.69
S^2LRR[15]	87.62±1.35	84.32±1.68	93.40±1.19	94.67±1.79
S^3R[15]	95.71±0.45	**95.39±1.29**	98.10±0.66	**98.09±0.34**
JSREPL[16]	93.30±0.00	84.66±0.00	–	89.62±0.00
SMFA-SSC	**95.74±0.42**	94.76±0.72	**98.50±0.59**	97.62±0.29

database) when 10% of the data points are labeled. In the Yale B dataset, our proposed algorithm achieves the highest score, with improvements of 8.08, 3.52, 8.13, and 20.11 over SSC-NGC, S^3R, JSREPL, and S^2LRR, respectively, which also adopt the unified optimization framework. This suggests that the proposed data representation is more informative and thus more suitable for semi-supervised clustering.

(3) When the percentage of labeled data exceeds 10%, the proposed method outperforms other algorithms on the Yale B and AR datasets, and achieves the second-best performance on the ORL dataset, with results close to the best. For the PIE dataset, traditional semi-supervised clustering methods, ASL, SEE, and FME, perform very well.

(4) The clustering accuracy results for the PALM dataset are presented in Table 5, demonstrating that the proposed method consistently achieves a perfect score of 100.00% clustering accuracy.

From the performance on five public available datasets, it is evident that the proposed method shows good clustering performance especially when there are fewer labeled data points. Due to the introduction of regularization, compared with the unified frameworks of JSREPL, S^2LRR, S^3R, and SSC-NGC, the proposed algorithm demonstrates superior performance which illustrates that the projection of the original high-dimensional data into the embedding subspace is beneficial for clustering and fully extracting the intraclass similarity and interclass discrimination to guide the affinity construction and label propagation is advantageous for semi-supervised subspace clustering.

5.3 Parameter selection and sensitivity

The proposed SMFA-SSC model contains 4 parameters, i.e., α, β, γ, and η for regularization term. Different datasets may have different optimal parameters for clustering task, and as far as we know, there is still an open problem to adaptively select the optimal parameters. In the following, we study the influence of parameters α, β, γ, and η on v_{ACC}. These four parameters are chosen from an empirical and reasonable candidate set $[10^{-8}, 10^{-5}, 10^{-2}, 10^{-1}, 0.5, 1.0, 10.0]$. During the experiment, we vary a parameter

while keeping others fixed and conduct the experiment in the case of 10% labeled data on two datasets: Yale B and PIE. Corresponding clustering results with different parameter settings are given in Fig. 2. From the results, we can see that the proposed model performs well and is relatively stable when β increases from 0.1 to 1.0, and with increasing β from 1.0 to above, v_{ACC} decreases slightly from 90% to 80%. With decreasing β from 0.1 to 10^{-8}, v_{ACC} drops from 94% to 10%. We can find that the reconstructed error term parameter β and the low-rank constraint term parameter γ have a large impact on the proposed method, since the low-rank model has direct effect on the performance of the clustering results by seeking the lowest-rank matrix to capture the irrelevant subspace. As the regularizer parameter α increases from 10^{-5} to 10^{-2} and the parameter η increases from 10^{-8} to 10^{-2}, the clustering accuracy of the model gradually increases, which shows that the constraints have a positive impact on the performance of the algorithm, and the performance of SMFA-SSC is somewhat sensitive to the value setting of parameter η.

5.4 Visualization of representation

Figure 3 presents the visual representation of affinity matrices for four different methods: LRR, S^2LRR, S^3R, and our proposed SMFA-SSC, respectively, when the percentage of labeled data is 10%. The block-diagonal structure in these matrices is indicative of effective clustering, where high intra-cluster similarity and low inter-cluster similarity are expected. As can be seen, the affinity matrix generated by SMFA-SSC shows clearer and more distinct block structures compared to other methods, reflecting better separation between clusters and tighter grouping within clusters. This demonstrates the superior ability of SMFA-SSC to capture both global and local structures in the data, which is crucial for accurate clustering.

5.5 Robustness to noise

To investigate the robustness of various methods to noises, the images are corrupted by adding Gaussian noise with zero mean and variance σ^2. We choose five semi-supervised learning methods, i.e., GFHF, LGC,

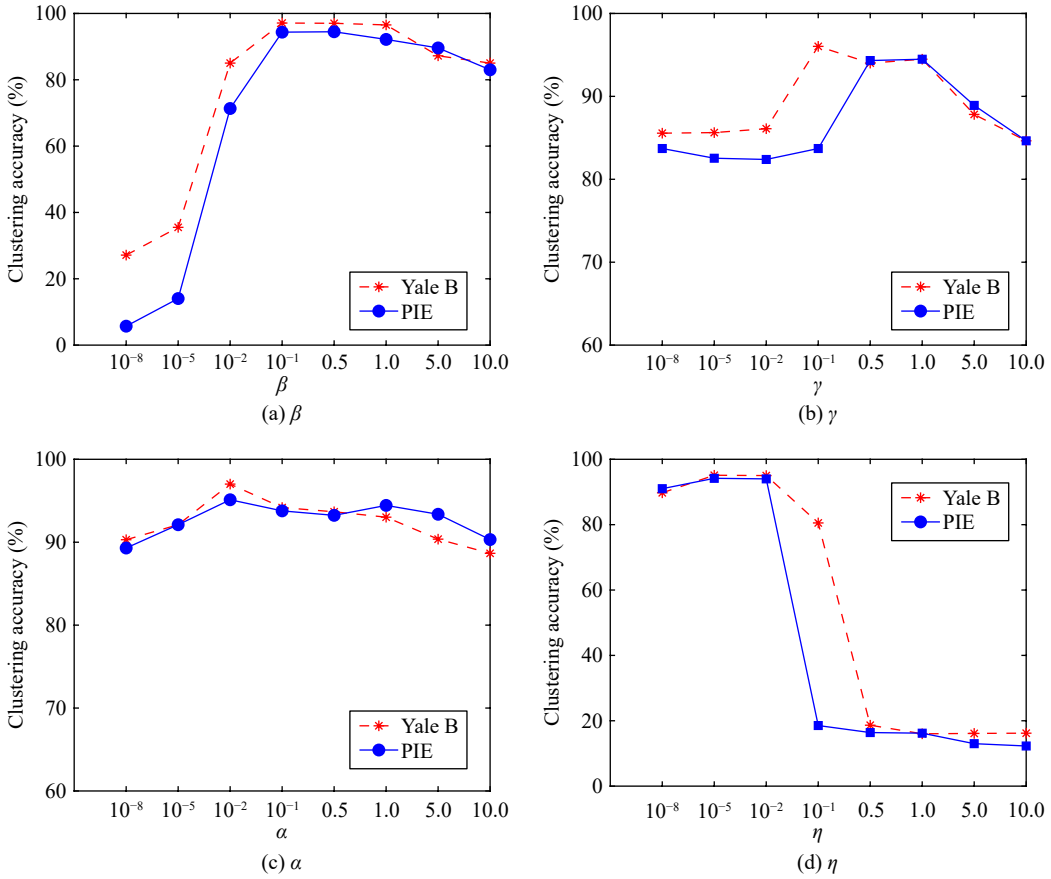


Fig. 2 Influence of SMFA-SSC parameters. These results are collected from the Yale B and PIE datasets, and the percentage of labeled data is 10%.

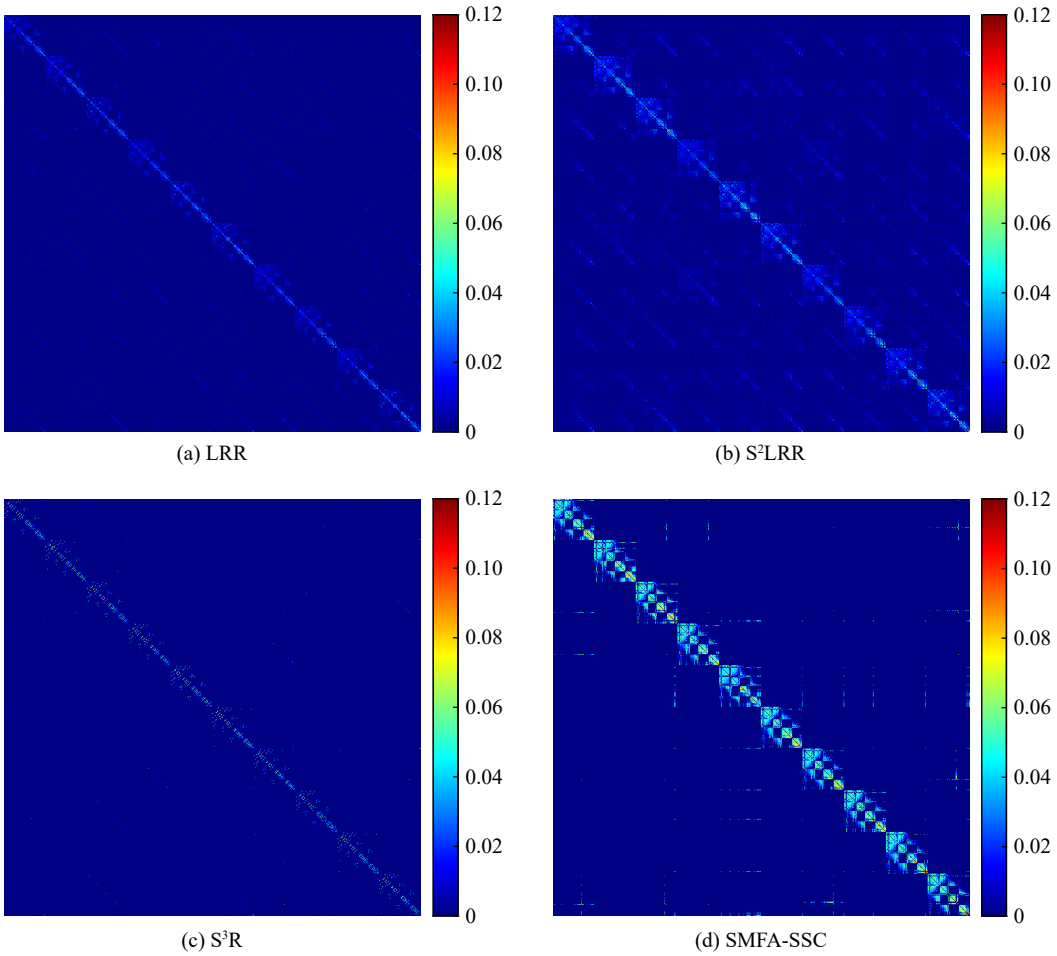


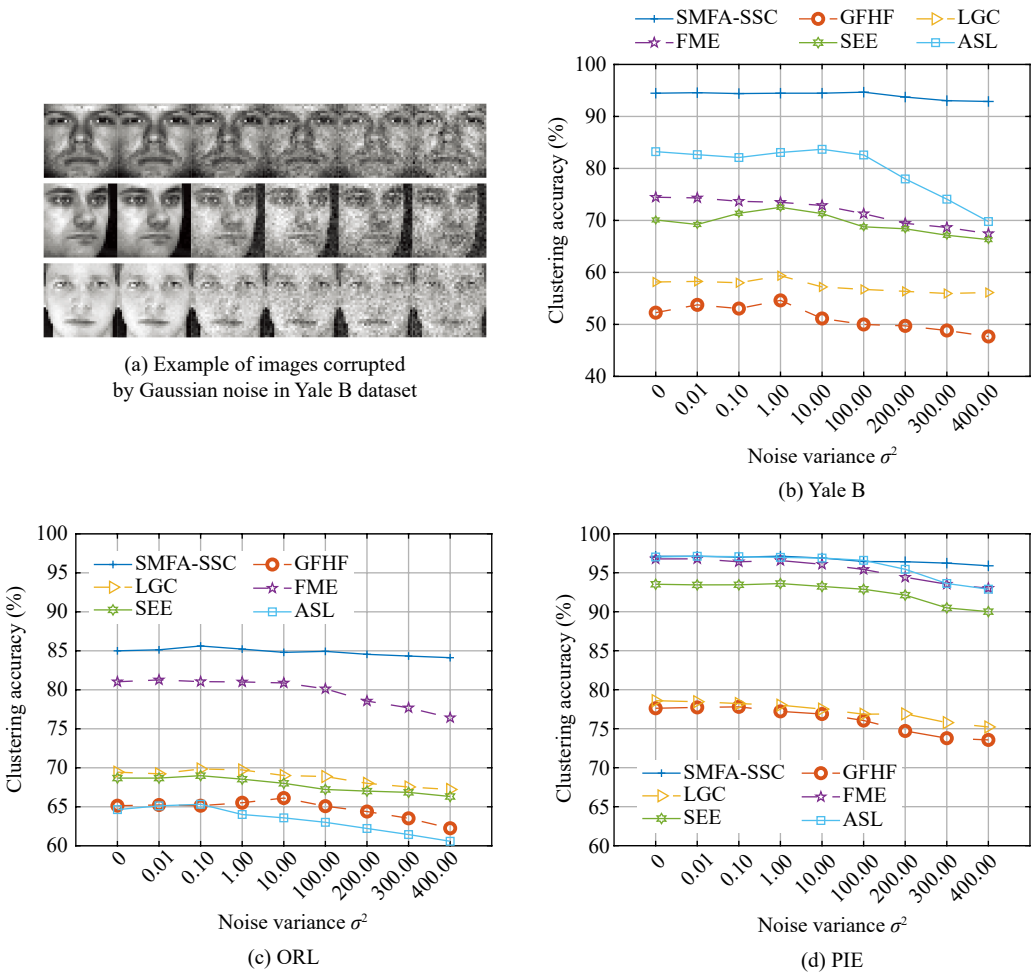
Fig. 3 Visualization of different representation matrices on Yale B dataset. The percentage of labeled data is 10%.

FME, SEE, and ASL for comparison and conduct the experiment in the case of 10% labeled data on three datasets: Yale B, PIE, and ORL.

In order to test the robustness of various methods, we add Gaussian white noise with variance of σ^2 = [0.01, 0.10, 1.00, 10.00, 100.00, 200.00, 300.00, 400.00] in the dataset, respectively and the corresponding clustering accuracy v_{ACC} is evaluated. The examples of the images in the Yale B dataset corrupted with Gaussian noise of different variances are shown in Fig. 4. We can see from Fig. 4 that with the increase of noise level, the final performance of each method decreases to varying extent. Obviously, the proposed SMFA-SSC outperforms other competitors, which shows the robustness to noises. This is mainly because SMFA-SSC model combines low-rank model and graph embedding technology together. The low-rank model can filter out the noise that will increase the matrix rank in the data by finding the lowest-rank matrix, and the graph embedding technology can filter out the irrelevant information in the data by reducing the dimension of the data, so the proposed algorithm has best performance robustness to noise.

6 Conclusion

In this paper, by projecting the high-dimensional data to a latent subspace, in which the intraclass compactness and interclass separation of data are required to be maximized, we proposed a generalized regularization to capture both the intrinsic manifold and discriminative structure of data. We incorporated the proposed regularization term into LRR to capture both the global and local structures of the whole data. Based on this, we proposed a unified framework for semi-supervised subspace clustering, SMFA-SSC, to simultaneously project the data, find the low-rank representation coefficient, and estimate the labels for unlabeled data points. Furthermore, we developed an efficient optimization algorithm for this framework and utilized ordinary differential equations to analyze the



(a) Example of images corrupted by Gaussian noise in Yale B dataset

(b) Yale B

(c) ORL

(d) PIE

Fig. 4 Illustration of images corrupted by Gaussian noise and clustering accuracies of different SSL methods under different noise variances σ^2 = [1, 10, 100, 200, 300, 400].

convergence of the proposed ADMM algorithm for low-rank representation in the projected subspace. Experiments on the widely used image datasets demonstrate that the proposed SMFA-SSC can achieve competitive clustering results than other state-of-the-art related methods, especially when there are less labeled data.

References

[1] J. He, B. Gong, J. Yang, H. Wang, P. Xu, and T. Xing, ASCFL: accurate and speedy semi-supervised clustering federated learning, *Tsinghua Science and Technology*, vol. 28, no. 5, pp. 823–837, 2023.

[2] J. Chen, J. Han, X. Meng, Y. Li, and H. Li, Graph convolutional network combined with semantic feature guidance for deep clustering, *Tsinghua Science and Technology*, vol. 27, no. 5, pp. 855–868, 2022.

[3] Z. Yu, F. Ye, K. Yang, W. Cao, C. L. P. Chen, L. Cheng, J. You, and H. S. Wong, Semisupervised classification with novel graph construction for high-dimensional data, *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 1, pp. 75–88, 2022.

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

[5] X. Fang, Y. Xu, X. Li, Z. Lai, and W. K. Wong, Robust semi-supervised subspace clustering via non-negative low-rank representation, *IEEE Trans. Cybern.*, vol. 46, no. 8, pp. 1828–1838, 2016.

[6] M. Yin, J. Gao, and Z. Lin, Laplacian regularized low-rank representation and its applications, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 3, pp. 504–517, 2016.

[7] W. K. Wong, Z. Lai, J. Wen, X. Fang, and Y. Lu, Low-rank embedding for robust image feature extraction, *IEEE Trans. Image Process.*, vol. 26, no. 6, pp. 2905–2917, 2017.

[8] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, Learning with local and global consistency, in *Proc. 16th Int. Conf. Neural Inf. Process. Syst.*, Cambridge, MA, USA, 2003, pp. 321–328.

[9] X. Zhu, Z. Ghahramani, and J. Lafferty, Semisupervised learning using Gaussian fields and harmonic functions, in

Proc. Int. Conf. Mach. Learn., Fort Lauderdale, FL, USA, 2003, pp. 912–919.

[10] B. Ni, S. Yan, and A. Kassim, Learning a propagable graph for semisupervised learning: classification and regression, *IEEE Trans. Knowl. Data Eng.*, vol. 24, no. 1, pp. 114–126, 2012.

[11] L. Zhuang, H. Gao, Z. Lin, Y. Ma, X. Zhang, and N. Yu, Non-negative low rank and sparse graph for semi-supervised learning, in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Providence, RI, USA, 2012, pp. 2338–2335.

[12] L. Zhuang, S. Gao, J. Tang, J. Wang, Z. Lin, Y. Ma, and N. Yu, Constructing a nonnegative low-rank and sparse graph with data-adaptive features, *IEEE Trans. Image Process.*, vol. 24, no. 11, pp. 3717–3728, 2015.

[13] M. Yin, J. Gao, Z. Lin, Q. Shi, and Y. Guo, Dual graph regularized latent low-rank representation for subspace clustering, *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 4918–4933, 2015.

[14] S. Yang, Z. Feng, Y. Ren, H. Liu, and L. Jiao, Semi-supervised classification via kernel low-rank representation graph, *Knowl. Based Syst.*, vol. 69, pp. 150–158, 2014.

[15] C.-G. Li, Z. Lin, H. Zhang, and J. Guo, Learning semi-supervised representation towards a unified optimization framework for semi-supervised learning, in *Proc. IEEE Int. Conf. Computer Vision* (*ICCV*), Santiago, Chile, 2015, pp. 2767–2775.

[16] X. Pei, C. Chen, and Y. Guan, Joint sparse representation and embedding propagation learning: A framework for graph-based semisupervised learning, *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 12, pp. 2949–2960, 2017.

[17] W. Wang, C. Yang, H. Chen, and X. Feng, Unified discriminative and coherent semi-supervised subspace clustering, *IEEE Trans. Image Process.*, vol. 27, no. 5, pp. 2461–2470, 2018.

[18] T. Qi, X. Feng, B. Gao, and K. Wang, An end-to-end graph convolutional network for semi-supervised subspace clustering via label self-expressiveness, *Knowl. Based Syst.*, vol. 286, p. 111393, 2024

[19] Z. Liu, J. Wang, G. Liu, and L. Zhang, Discriminative low-rank preserving projection for dimensionality reduction, *Appl. Soft Comput.*, vol. 85, p. 105768, 2019.

[20] Y.-P. Zhao, L. Chen, and C. L. P. Chen, Laplacian regularized nonnegative representation for clustering and dimensionality reduction, *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 1, pp. 1–14, 2021.

[21] Y. Liang, L. You, X. Lu, Z. He, and H. Wang, Low-rank projection learning via graph embedding, *Neurocomputing*, vol. 348, pp. 97–106, 2019.

[22] Y. Li, S. Wang, C. Li, Y. Yuan, and G. Wang, Towards very deep representation learning for subspace clustering, *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 7, pp. 3568–3579, 2024.

[23] Y. Wang, J. Zou, K. Wang, C. Liu, and X. Yuan, Semi-supervised deep embedded clustering with pairwise constraints and subset allocation, *Neural Netw.*, vol. 164, pp. 310–322, 2023.

[24] J. Li, G. Li, and Y. Yu, Adaptive betweenness clustering for semi-supervised domain adaptation, *IEEE Trans. Image Process.*, vol. 32, pp. 5580–5594, 2023.

[25] C. Wang, M. Yan, Y. Rahimi, and Y. Lou, Accelerated schemes for the L_1/L_2 minimization, *IEEE Trans. Signal Process.*, vol. 68, pp. 2660–2669, 2020.

[26] A. Iscen, G. Tolias, Y. Avrithis, and O. Chum, Label propagation for deep semi-supervised learning, in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, Long Beach, CA, USA, 2019, pp. 5070–5079.

[27] S. Yan, D. Xu, B. Zhang, H.-J. Zhang, Q. Yang, and S. Lin, Graph embedding and extensions: a general framework for dimensionality reduction, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 1, pp. 40–51, 2007.

[28] Z. Lin, H. Li, and C. Fang, *Alternating Direction Method of Multipliers for Machine Learning*. Singapore City, Singapore: Springer Nature Singapore, 2022.

[29] T.-H. Oh, Y. Matsushita, Y.-W. Tai, and I. S. Kweon, Fast randomized singular value thresholding for low-rank optimization, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 2, pp. 376–391, 2018.

[30] U. von Luxburg, A tutorial on spectral clustering, *Stat. Comput.*, vol. 17, no. 4, pp. 395–416, 2007.

[31] O. Chapelle, B. Schlkopf, and A. Zien, Semisupervised learning, *IEEE Trans. Neural Netw.*, vol. 20, no. 3, p. 542, 2009.

[32] F. Nie, D. Wu, R. Wang, and X. Li, Truncated robust principle component analysis with a general optimization framework, *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 2, pp. 1081–1097, 2022.

[33] B. Li and B. Shi, Understanding the ADMM algorithm via high-resolution differential equations, arXiv preprint arXiv: 2401.07096, 2024.

[34] Y. Zhao, X. He, M. Zhou, and T. Huang, Accelerated primal-dual projection neurodynamic approach with time scaling for linear and set constrained convex optimization problems, *IEEE/CAA J. Autom. Sinica*, vol. 11, no. 6, pp. 1485–1498, 2024.

[35] H. Ren, H. Ma, H. Li, and Z. Wang, Adaptive fixed-time control of nonlinear MASs with actuator faults, *IEEE/CAA J. Autom. Sinica*, vol. 10, no. 5, pp. 1252–1262, 2023.

[36] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, From few to many: Illumination cone models for face recognition under variable lighting and pose, *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 23, no. 6, pp. 643–660, 2001.

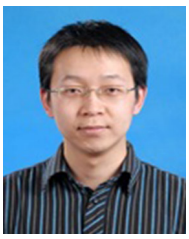
[37] F. S. Samaria and A. C. Harter, Parameterisation of a stochastic model for human face identification, in *Proc. IEEE Workshop on Applications of Computer Vision*, Sarasota, FL, USA, 1994, pp. 138–142.

[38] G.-S. J. Hsu, H.-Y. Wu, and M. H. Yap, A comprehensive study on loss functions for cross-factor face recognition, in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition Workshops*, Seattle, WA, USA, 2020, pp. 826–827.

[39] T. Sim, S. Baker, and M. Bsat, The CMU pose, illumination, and expression (PIE) database, in *Proc. Fifth IEEE Int. Conf. Automatic Face Gesture Recognition*,

Washington, DC, USA, 2002, pp. 53–58.
[40] F. Nie, J. Yuan, and H. Huang, Optimal mean robust principal component analysis, in *Proc. 31st Int. Conf. Mach. Learn*., Beijing, China, 2014, pp. 2755–2763, 2014.
[41] F. Nie, D. Xu, I. W.-H. Tsang, and C. Zhang, Flexible manifold embedding: A framework for semi-supervised and unsupervised dimension reduction, *IEEE Trans. Image Process.*, vol. 19, no. 7, pp. 1921–1932, 2010.
[42] F. Nie, H. Wang, H. Huang, and D. Chris, Adaptive loss minimization for semi-supervised elastic embedding, in *Proc. 23nd Int. Joint Conf. Artif. Intell*., Bejing, China, 2013, pp. 1565–1571.
[43] D. Wang, F. Nie, and H. Huang, Large-scale adaptive semi-supervised learning via unified inductive and transductive model, in *Proc. 20th ACM SIGKDD Int. Conf. Knowledge discovery and data mining*, New York, NY, USA, 2014, pp, 482–491.
[44] S. Peng, W. Ser, B. Chen, and Z. Lin, Robust semi-supervised nonnegative matrix factorization for image clustering, *Pattern Recognit.*, vol. 111, p. 107683, 2021.

Xin Lv received the BS degree in electronic information science and technology from Lanzhou University, Lanzhou, China, in 2003, and the MS degree in information and communication engineering and PhD degree in radio physics from Lanzhou University, China in 2006 and 2015, respectively. She is currently a lecturer at School of Information Science and Engineering, Lanzhou University, China. Her research interests include machine learning, neural networks, and optimization.



Zhenming Su received the BS and MS degrees in computer science and technology from Lanzhou University, Lanzhou, China, in 2004 and 2007, respectively, and the PhD degree in radio physics from Lanzhou University, China, in 2017. He is currently a lecturer at School of Information Science and Engineering, Lanzhou University, China. His research interests include neural networks and optimization.

Baifei Chai received the MS degree in information and communication engineering from Lanzhou University, Lanzhou, China, in 2019. His research interests include machine learning, neural networks, and optimization.