Iranian Journal of Fuzzy Systems

Volume 21, Number 3, (2024), pp. 155-175

Original paper

University of Sistan and Baluchestan

Robust semi-supervised fuzzy clustering algorithm based on pairwise constraints

X. Yang [1], L. Jia [2], Y. Ma [3], X. L. Xin [4] and M. M. Zahedi [5]

[1,2,3,4] *School of Science, Xi'an Polytechnic University, Xi'an, China*
[4] *School of Mathematics, Northwest University, Xi'an, China*
[5] *Department of Mathematics, Graduate University of Advanced Technology, Kerman, Iran*

yangxiaofei2002@163.com, j1070819326@163.com, mayingcang@xpu.edu.cn, xlxin@nwu.edu.cn, Zahedi_mm@kgut.ac.ir

Abstract

Semi-supervised clustering, utilizing the supervision information to guide the clustering process, could improve the clustering effect of the models. Most of existing semi-supervised clustering models only consider pairwise constraints or pointwise constraints. In this paper, the semi-supervised method is applied to the fuzzy clustering algorithm, and a robust semi-supervised fuzzy clustering algorithm is proposed. Firstly, fully considering prior knowledge, our models integrate pointwise constraints and pairwise constraints into a unified framework to improve the clustering performance of the fuzzy clustering algorithm. Secondly, in order to alleviate the impact of outliers, the robust performance of the models is considered by introducing an adaptive loss function into the models. Thirdly, our models can capture the global structures and the local manifold structures of data sets. Finally, a simple and efficient algorithm is proposed to solve the models, which ensures that the obtained solution is sparse and satisfies the constraint conditions in our models. Compared with five representative methods, experimental results on public datasets, such as text dataset (dbworld), voice dataset (Isolet), image datasets (YALE, Umist), chemical dataset (wine) and biological datasets (colon, TOX-171), show the effectiveness of the proposed models.

Keywords: Semi-supervised clustering, adaptive loss, pairwise constraints, fuzzy clustering, label information.

1 Introduction

Clustering divides data points into different clusters according to the similarity between samples. And it can be used to process a variety of datasets, such as text datasets, digit datasets, face datasets and speech datasets, etc, which is widely applied to the real world. In the past few decades, many classical clustering algorithms have been proposed for different kinds of datasets in real application, such as fuzzy clustering [33], subspace clustering [36], graph-based clustering [13, 40] and multi-view clustering [14, 41], etc.

Among the existing clustering algorithms, the fuzzy clustering algorithms, which combine fuzzy theory and clustering technology, become a popular clustering method and are widely used in remote sensing image categorization [28], medical diagnosis [26] and insurance fraud detection [16], etc. The fuzzy c-mean clustering algorithm(FCM) is a classical fuzzy clustering algorithm. Different from k-means, the membership of the sample in a cluster belongs to [0,1] in FCM. FCM has the following advantages: low space complexity, stable performance and easy achievement. Thus it has received more and more scholars' attention [1, 32, 35, 37–39, 42]. In order to deal with high-dimensional datasets, Zhang et al. [39] used deep feature extraction and fuzzy clustering simultaneously to generate a more appropriate nonlinear feature map and obtained a better clustering performance. Zhou et al. [42] first time used projection techniques to improve the capability of FCM to deal with high-dimensional data. To improve the comprehensive benefits of the CCHP system, Zhang et al. [38] proposed a bi-level optimal configuration model of the CCHP system based on the improved FCM

Corresponding Author: Y. Ma
Received: September 2023; Revised: June 2024; Accepted: September 2024.
https://doi.org/10.22111/ijfs.2024.46744.8238

clustering algorithm. In order to determine the number of optimal clusters, Wang et al. [32] proposed a clustering validity function fusion model based on D-S evidence theory.

Semi-supervised clustering usually uses part label information by two methods: pairwise constraints [5, 9, 12], such as must-link, cannot-link, and pointwise constraints [23]. It is a hot topic how to effectively exploit prior information in various clustering models. Chen et al. [5] innovatively integrated pairwise constraints into the process of multi-view clustering by setting the learned multi-view representation of must-link samples (cannot-link samples) to be similar (dissimilar), so that the clustering results can be more credible. The method in [9] is the first approach to fuse pairwise constraints and monotonicity constraints in monotonic datasets, and have a better clustering performance. On the other hand, label propagation, as a famous method of pointwise constraints, is considered in the related subspace clustering models [23].

In fuzzy clustering process, some priori information is also considered to obtain the better clustering performance. In [3], they consider must-link and cannot-link pairwise constraints in fuzzy clustering, and propose a robust semi-supervised multi-view fuzzy clustering algorithm, which can provide better interpretability of the clustering results. To deal with insufficient and imprecise membership degree in fuzzy clustering, Wang et al. [30] proposed a new multi-manifold regularization semi-supervised fuzzy clustering method based on pairwise constraints. Tuan et al. [27] considered different fuzzifiers for each data point and for each cluster, respectively in the standard fuzzy c-mean clustering, so that the proposed model can improve clusters quality. Different from traditional supervised information, some labels may be wrong. Gan et al. [8] focused on this issue, and developed a novel safe semi-supervised clustering which can deal with mislabeled instances. Wang et al. [31] extended the traditional pairwise constraint (i.e., must-link or cannot-link) to fuzzy pairwise constraint, which can represent more complicated relationship between the pair of samples. Oskouei et al. [21] exploited an automatic cluster weighting scheme in FCM to reduce the sensitivity to the initialization of primary clusters, the proposed model has a good performance. Mei et al. [17] made theoretical comparison of the above pairwise constrained approaches, and first proposed two types of penalties– consistency-type and distance-type penalties.

It is a challenge to tackle data with noise. Wu et al. [34] presented a robust dynamic semi-supervised symmetric regularized picture fuzzy clustering for image segmentation with high noise. Different from existing fuzzy clustering, Oskouei et al. [22] noticed that some features may be more informative than others, and proposed novel semi-supervised fuzzy c-means approach based on feature-weight, and cluster-weight learning, which is insensitive to the initial center selection and noise.

Most of the above papers have not fully utilized priori information. They only considered pairwise constraint information or pointwise constraints information. In our models, we add pairwise constraint information in similarity matrices and pointwise constraints in the constraint conditions. In particular, the similarity of two samples with different labels is set to be -1 in similarity matrices for our models, which can more accurately reflect the cannot-link information between the two samples.

In addition, FCM uses Euclidean distance to calculate the membership of each sample point. The Euclidean distance is very sensitive to outliers, which makes FCM poor robustness. In order to deal with this issue, we propose a robust semi-supervised clustering algorithm based on pairwise constraints(PC-RSFCM). The main contributions of this paper are as follows:

(1) We replace the Euclidean distance in FCM with an adaptive loss function and get the robust models.

(2) To improve the clustering performance, our models fully consider the label information, and integrate pointwise constraints and pairwise constraints into a unified framework.

(3) Our model can capture the global structures and the local manifold structures of data sets. Thus the membership matrix can faithfully reveal intrinsic structure of data set.

(4) A simple and efficient algorithm is proposed to guarantee that all elements of the membership matrix are greater than or equal to zero, and our algorithm extends the algorithm in [29] to the generalized case.

The rest of the paper is organized as follows. In Section 2, we review the definitions of fuzzy clustering and adaptive loss function, and give the construction of the similarity matrix in this paper. The model building process is given in Section 3. In Section 4, we give the optimization process of model and the proof of convergence of algorithms. In Section 5, the experiments is presented. Futher more, the sensitivity of the model parameters and convergence performance of the model are discussed. Finally, in Section 6, the proposed algorithm is summarized.

2 Related work

In this section, the related work is introduced. For clarity, some notations are given in Table 1. And then, the model of fuzzy c-means clustering is presented. Next, the similarity matrix in our models is constructed by cosine similarity. Finally, some properties of adaptive loss function are introduced.

2.1 The description of the symbols

In this section, we explain the definition of the notations who are used in our paper, the description of the symbols are showed in Tabel 1.

Table 1: The description table of the symbols

Symbol	Description
$X \in \mathbb{R}^{n\times d}$	the set of n sample points
$x_i \in \mathbb{R}^d$	the i-th sample point
d	the dimension of each sample data
c	the number of clusters
m	fuzzy factor
$c_k \in \mathbb{R}^d$	the centers of the k-th cluster
$U \in \mathbb{R}^{n\times c}$	the membership degree matrix
u_{ik}	the membership degree of the i-th sample point in the k-th cluster
$W \in \mathbb{R}^{n\times n}$	the similarity matrix
$N_k(x_i)$	the set of the k-nearset neighbors of x_i
l	the number of labeled data
u	the number of unlabeled data
$X_l = \{\boldsymbol{x}_i\}_{i=1}^{l}$	labeled data in the dataset
$X_u = \{\boldsymbol{x}_i\}_{i=l+1}^{n}$	unlabeled data in the dataset
$Y_l = \{\boldsymbol{y}_i\}_{i=1}^{l}$	the label of the labeled data
L_W	the Laplacian matrix of the similarity matrix W

2.2 Fuzzy c-means clustering

Let $X = [x_1, x_2, ..., x_n] \in \mathbb{R}^{n\times d}$ be the set of n sample points, $x_i \in \mathbb{R}^d$ denote the i-th sample point and d be the dimension of each sample data. Then the objective function of the traditional FCM is defined as:

$$\begin{aligned} &\min_{U,c_k} \sum_{i=1}^{n}\sum_{k=1}^{c} \|x_i - c_k\|_2^2 \, u_{ik}^m, \\ &s.t. \sum_{k=1}^{c} u_{ik} = 1, u_{ik} \geq 0, \end{aligned} \tag{1}$$

where c is the number of clusters, m is the fuzzy factor, $\|\bullet\|_2^2$ is the square of the ℓ_2-norm, $c_k \in \mathbb{R}^d$ is the centers of the k-th cluster, u_{ik} is the membership degree of the i-th sample point in the k-th cluster, and is the (i,k) element of the membership matrix U.

By optimizing the objective function of FCM, the equations for updating the membership degree u_{ik} and clustering center c_k are as follows:

$$\begin{aligned} u_{ik} &= \frac{1}{\sum_{j=1}^{c}\left(\frac{\|x_i - c_k\|}{\|x_i - c_j\|}\right)^{\frac{2}{m-1}}}, \\ c_k &= \frac{\sum_{i=1}^{n} u_{ik}^m x_i}{\sum_{i=1}^{n} u_{ik}^m}. \end{aligned} \tag{2}$$

The iteration is stopped if $max\left|u_{ik}^{t+1} - u_{ik}^{t}\right| < \epsilon$, where u_{ik} is computed by Eq.(2), ϵ is a small positive integer and t is the number of iterations [2, 11]. The solution is the local minimum point or saddle point of the objective function.

Finally, each sample is assigned to the corresponding group according to the value of the membership matrix U. When $m = 1$, FCM theoretically converges to the traditional k-means solution [25].

2.3 Similarity matrix

It is a key issue to construct similarity matrix in clustering. In this work, dot product and neighborhood are used to construct similarity matrix. In detail, similarity matrix $S = (s_{ij})_{n\times n}$ is set up as follows:

$$s_{ij} = \begin{cases} \frac{\boldsymbol{x}_i{}^T \boldsymbol{x}_j}{\|\boldsymbol{x}_i\|\|\boldsymbol{x}_j\|}, & x_i \in N_k(x_j) \text{ or } x_j \in N_k(x_i), \\ 0, & \text{otherwise}, \end{cases} \tag{3}$$

where $N_k(x_i)$ is the set of the k-nearset neighbors of x_i.

In order to make full use of the known labels, the label information is considered to the above similarity matrix. If x_i and x_j belong to the same group, x_i and x_j are said to be must-link. On the contrary, if x_i and x_j belong to the different groups, x_i and x_j are called to be cannot-link. The new similarity matrix $W = (w_{ij})_{n\times n}$ is obtained as follows:

$$w_{ij} = \begin{cases} 1, & (x_i, x_j) \in \mathscr{M}, \\ -1, & (x_i, x_j) \in \mathscr{C}, \\ s_{ij}, & \text{otherwise}, \end{cases} \tag{4}$$

where $\mathscr{M}$ and $\mathscr{C}$ denote the set of must-link pairwise constraints and cannot-link pairwise constraints, respectively.

Let $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \boldsymbol{x}_4, \boldsymbol{x}_5, \boldsymbol{x}_6\}$ be a dataset. The samples $\boldsymbol{x}_1, \boldsymbol{x}_2$ belongs to the c_1 cluster, samples $\boldsymbol{x}_3, \boldsymbol{x}_4$ belong to c_2 cluster, and samples $\boldsymbol{x}_5, \boldsymbol{x}_6$ are unlabeled. According to Eq.(4), the similarity matrix W can be presented as

$$W = \begin{pmatrix} 1 & 1 & -1 & -1 & s_{15} & s_{16} \\ 1 & 1 & -1 & -1 & s_{25} & s_{26} \\ -1 & -1 & 1 & 1 & s_{35} & s_{36} \\ -1 & -1 & 1 & 1 & s_{45} & s_{46} \\ s_{51} & s_{52} & s_{53} & s_{54} & 1 & s_{56} \\ s_{61} & s_{62} & s_{63} & s_{64} & s_{65} & 1 \end{pmatrix},$$

where $s_{ij}(i, j = 1, 2, 3, 4, 5, 6)$ is a similarity degree constructed by Eq.(3). In order to avoid clustering different group points into the same group, we set up the similarity between samples in the different group as -1. Hence, the similarity matrix W is different from the most similarity matrix. In this way, negative numbers appear in the similarity matrix such that the model is difficult to solve. In order to solve this problem, we use a simple and effective algorithm which has small calculation cost.

2.4 Adaptive loss function

Given arbitrary vector $\boldsymbol{x}$, ℓ_1-norm and ℓ_2-norm are defined as $\|\boldsymbol{x}\|_1 = \sum_i |x_i|$ and $\|\boldsymbol{x}\|_2 = \sqrt{\sum_i (x_i^2)}$, respectively. Generally, squared ℓ_2-norm loss is sensitive to large outliers and ℓ_1-norm loss is sensitive to small outliers. In order to control robustness to various type outliers, adaptive loss function for vector $\boldsymbol{x}$ is given as [6, 20]:

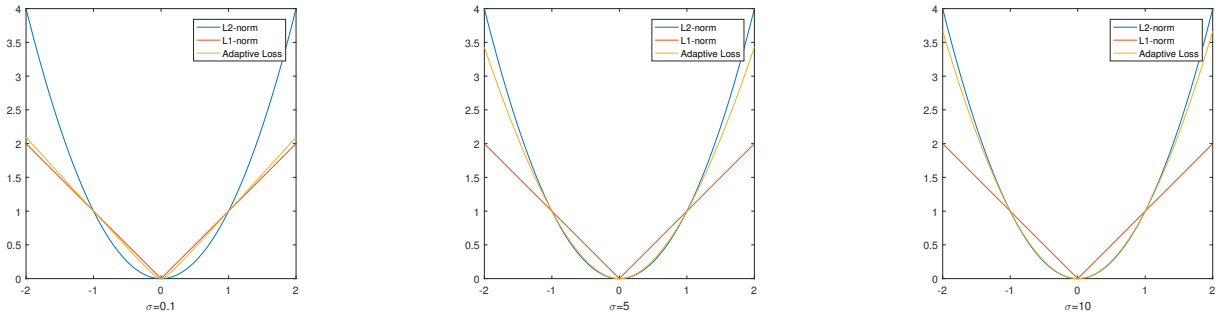
$$\|\boldsymbol{x}\|_\sigma = \sum_i \frac{(1+\sigma)x_i^2}{|x_i| + \sigma}, \tag{5}$$

where σ is a tradeoff parameter that controls robustness to various type outliers, and x_i is the i-th element of the vector $\boldsymbol{x}$. The illustrations of different σ values ($\sigma = 0.1, 5, 10$) for the adaptive loss function are shown in Fig.1 (a)-(c). It can be easily seen that the adaptive loss function in Eq.(5) is intermediate between ℓ_1-norm and ℓ_2-norm. Therefore the adaptive loss function can combine their advantages. In addition, the adaptive loss function in Eq.(5) has the following properties:

1. $\|\boldsymbol{x}\|_\sigma$ is second-order derivable.
2. If $\forall i$, $x_i \ll \sigma$, then $\|\boldsymbol{x}\|_\sigma \to \frac{1+\sigma}{\sigma}\|\boldsymbol{x}\|_2^2$.
3. If $\forall i$, $x_i \gg \sigma$, then $\|\boldsymbol{x}\|_\sigma \to (1+\sigma)\|\boldsymbol{x}\|_1$.

4. If $\sigma \to 0$, then $\|\boldsymbol{x}\|_\sigma \to \|\boldsymbol{x}\|_1$.
5. If $\sigma \to \infty$, then $\|\boldsymbol{x}\|_\sigma \to \|\boldsymbol{x}\|_2^2$.

Finally, the adaptive loss function can be extended from vector to matrix and the detail is described in [20].



(a) σ=0.1 (b) σ=5 (c) σ=10

Figure 1: Illustration of ℓ_1-norm, ℓ_2-norm, and the adaptive loss function with different σ

3 The proposed semi-supervised clustering model

Based on pairwise constraints, we propose a robust semi-supervised fuzzy clustering algorithm(PC-RSFCM). In FCM model, the known label information is added in the membership matrix. Moreover, the information of pairwise constraints is used in the similarity matrix given in Section 2.3. In this way, the manifold structure is preserved. In addition, the adaptive loss function mentioned in Section 2.4 is applied to solve the problem of outliers sensitivity.

In the traditional FCM model (1), when $m = 1$ and $m = 2$, two special models are as follows:

$$\min_{U,c_k} \sum_{i=1}^{n}\sum_{k=1}^{c} \|x_i - c_k\|_2^2 \, u_{ik}, \tag{6}$$

$$\min_{U,c_k} \sum_{i=1}^{n}\sum_{k=1}^{c} \|x_i - c_k\|_2^2 \, u_{ik}^2. \tag{7}$$

Eq.(6) and Eq.(7) have the same constraint: $\sum_{k=1}^{c} u_{ik} = 1, u_{ik} \geq 0$.

Denote $\{X_l, Y_l\} = \{\boldsymbol{x}_i, \boldsymbol{y}_i\}_{i=1}^{l}$ as labeled data in the dataset, and $X_u = \{\boldsymbol{x}_i\}_{i=l+1}^{n}$ as unlabeled data in the dataset, where l and u are the number of labeled and unlabeled data, respectively. In order to improve the clustering performance, the information of known labels is applied to the membership matrix U in Eq.(1). Then we can get the following semi-supervised condition:

$$U_l = Y_l, \tag{8}$$

where U_l is the membership matrix of the first l samples (known label information samples). Using Eq.(8), we obtain the following model:

$$\begin{aligned} &\min_{U,c_k} \sum_{i=1}^{n}\sum_{k=1}^{c} \|x_i - c_k\|_2^2 \, u_{ik}^m, \\ &s.t. \sum_{k=1}^{c} u_{ik} = 1, u_{ik} \geq 0, U_l = Y_l. \end{aligned} \tag{9}$$

For a better clustering result, the label information is also used to construct the similarity matrix W (see Eq.(4)). By the similiarity matrix, the manifold structure between the data is preserved. Thus, the following model is obtained based on Eq.(9):

$$\begin{aligned} \min_{U,c_k} \sum_{i=1}^{n}\sum_{k=1}^{c} \|x_i - c_k\|_2^2 u_{ik}^m + \frac{\alpha}{2} Tr\left(U^T L_W U\right), \\ s.t. \sum_{k=1}^{c} u_{ik} = 1, u_{ik} \geq 0, U_l = Y_l, \end{aligned} \tag{10}$$

where $D_{ii} = \sum_{j=1}^{n} w_{ij}$, $L_W = D - W$ is the Laplacian matrix of the similarity matrix W, and α is the tradeoff parameter of the manifold regular term.

In order to improve the robust performance of the model (10), we use the adaptive loss, which is robust to outliers mentioned in Eq.(5), instead of the squared ℓ_2-norm loss. In this work, we only study the cases of $m = 1$ and $m = 2$ in model (10). Further more, the regularization term ℓ_2-norm is considered to reduce the complexity of model (10). Thereby, two robust semi-supervised fuzzy clustering algorithm models are as follows:

$$\begin{aligned} \min_{U,c_k} \sum_{i=1}^{n}\sum_{k=1}^{c} (\|x_i - c_k\|_\sigma u_{ik} + \lambda u_{ik}^2) + \frac{\alpha}{2} Tr\left(U^T L_W U\right), \\ s.t. \sum_{k=1}^{c} u_{ik} = 1, u_{ik} \geq 0, U_l = Y_l, \end{aligned} \tag{11}$$

$$\begin{aligned} \min_{U,c_k} \sum_{i=1}^{n}\sum_{k=1}^{c} (\|x_i - c_k\|_\sigma u_{ik}^2 + \lambda u_{ik}^2) + \frac{\alpha}{2} Tr\left(U^T L_W U\right), \\ s.t. \sum_{k=1}^{c} u_{ik} = 1, u_{ik} \geq 0, U_l = Y_l, \end{aligned} \tag{12}$$

where σ is the adaptive loss parameter, and λ is the tradeoff parameter of the regularization term.

4 Optimization algorithm

In this section, we firstly introduce the optimization process for solving Eq.(5), and then propose the optimization algorithm of Eq.(11) and Eq.(12).

4.1 Algorithm for the general adaptive loss minimization problem of Eq.(5)

Since Eq.(5) is difficult to be optimized, the literature [20] proposes an efficient algorithm to solve for the optimal model. Firstly, a more general adaptive loss function form of Eq.(5) is defined as

$$\min_x \ f(x) + \sum_i \|g_i(x)\|_\sigma, \tag{13}$$

where $g_i(x)$ is a vector-output function. It can be seen that Eq.(5) is a special case of Eq.(13). According to [20], the minimization problem (13) is solved using an iterative re-weighting algorithm. Taking the derivative of Eq.(13) and setting the derivative to zero, we have

$$f'(x) + 2(1+\sigma) \sum_i \frac{\|g_i(x)\|_2 + 2\sigma}{2(\|g_i(x)\|_2 + \sigma)^2} g_i(x) g_i'(x) = 0. \tag{14}$$

In Eq.(14), let $(1+\sigma)\frac{\|g_i(x)\|_2 + 2\sigma}{2(\|g_i(x)\|_2 + \sigma)^2}$ be equal to p_i, then the following equation is obtained

$$f'(x) + 2\sum_i p_i g_i(x) g_i'(x) = 0. \tag{15}$$

Since the variable p_i depends on x, Eq.(15) is difficult to solve directly. If p_i is fixed, then solving Eq.(13) is equivalent to solving the following problem

$$\min_x f(x) + \sum_i p_i \|g_i(x)\|_2^2, \tag{16}$$

where p_i is the transitional weight. Therefore, the optimal solution to the generalized problem Eq.(13) is proposed in Algorithm 1 as follows. In Algorithm 1, p_i is calculated based on the current $\boldsymbol{x}$, and the solution $\boldsymbol{x}$ is updated by solving Eq.(16). The convergence of Algorithm 1 has been proved in [20].

Algorithm 1 Algorithm to solve Eq.(13).

Input: A vector $\boldsymbol{x}$ and parameter σ.
repeat:
1. Calculate $p_i = (1+\sigma)\frac{\|g_i(x)\|_2+2\sigma}{2(\|g_i(x)\|_2+\sigma)^2}$.
2. Update $\boldsymbol{x}$ by solving Eq.(16).
until converge
Output: $\boldsymbol{x}$.

4.2 Optimization algorithms to solving problem (11) and problem (12)

According to Algorithm 1, the objective functions presented in models (11) and (12) can be rewritten as

$$\begin{aligned} \min_{U,c_k} \sum_{i=1}^{n}\sum_{k=1}^{c}(h_{ik}\left\|x_i-c_k\right\|_2^2 u_{ik}+\lambda u_{ik}^2)+\frac{\alpha}{2}Tr\left(U^T L_W U\right), \\ s.t.\sum_{k=1}^{c} u_{ik}=1, u_{ik}\geq 0, U_l=Y_l, \end{aligned} \tag{17}$$

$$\begin{aligned} \min_{U,c_k} \sum_{i=1}^{n}\sum_{k=1}^{c}(h_{ik}\left\|x_i-c_k\right\|_2^2 u_{ik}^2+\lambda u_{ik}^2)+\frac{\alpha}{2}Tr\left(U^T L_W U\right), \\ s.t.\sum_{k=1}^{c} u_{ik}=1, u_{ik}\geq 0, U_l=Y_l, \end{aligned} \tag{18}$$

where $h_{ik} = (1+\sigma)\frac{\|x_i-c_k\|_2+2\sigma}{2(\|x_i-c_k\|_2+\sigma)^2}$.

Thus, solving Eq.(11) is equivalent to solving Eq.(17), and solving Eq.(12) is equivalent to solving Eq.(18). In the following, we will use the alternating iterative method to solve Eq.(17) and Eq.(18).

4.2.1 Optimization algorithm to solving problem (11)

Update C by Fixing U: Problem (17) can be written as

$$\min_{c_k} \sum_{i=1}^{n}\sum_{k=1}^{c} h_{ik}\left\|x_i-c_k\right\|_2^2 u_{ik}. \tag{19}$$

By taking the derivative of Eq.(19) w.r.t. c_k and setting the derivative to zero, we have

$$c_k = \frac{\sum_{i=1}^{n} h_{ik}u_{ik}x_i}{\sum_{i=1}^{n} h_{ik}u_{ik}}. \tag{20}$$

Update U by Fixing C: Problem (17) becomes

$$\begin{aligned} \min_{U} \sum_{i=1}^{n}\sum_{k=1}^{c} e_{ik}u_{ik}+\lambda u_{ik}^2+\frac{\alpha}{2}Tr\left(U^T L_W U\right), \\ s.t.\sum_{k=1}^{c} u_{ik}=1, u_{ik}\geq 0, U_l=Y_l, \end{aligned} \tag{21}$$

where $e_{ik} = h_{ik}\left\|x_i-c_k\right\|_2^2$.

The solving process for Eq.(21) is as follows.

To ensure that U always satisfies the condition of $u_{ik}\geq 0$, inspired by the literature [29], the process of solving U is carried out as follows:

Given any sample point $\boldsymbol{x} \in \mathbb{R}^d$. Define the vectors $\boldsymbol{w} = (w_j)$ and $\boldsymbol{e} = (e_k)$, where w_j are the affinity between $\boldsymbol{x}$ and $\boldsymbol{x}_j$, $j = 1, \ldots, n$, and $e_k = h_k \|\boldsymbol{x} - c_k\|_2^2$, $k = 1, \ldots, c$, respectively. After removing the constant term, the optimization problem (21) is reduced to n quadratic programs where each quadratic program has c variables as follows:

$$\begin{aligned} \min_{\boldsymbol{u}} \frac{1}{2} \|\boldsymbol{u} - (\widetilde{\boldsymbol{u}} - \gamma \boldsymbol{e})\|^2, \\ s.t. \boldsymbol{u}^T \mathbf{1}_c = 1, \boldsymbol{u} \geq 0, \end{aligned} \tag{22}$$

where $\gamma = \frac{1}{2\lambda + \alpha \sum_{j=1}^n w_j}$, $\widetilde{\boldsymbol{u}} = \frac{\alpha \sum_{j=1}^n w_j \boldsymbol{u}_j}{2\lambda + \alpha \sum_{j'=1}^n w_{j'}}$.

Let $\boldsymbol{u}$ equal to $\boldsymbol{x}$, and $\widetilde{\boldsymbol{u}} - \gamma \boldsymbol{e}$ equal to $\boldsymbol{y}$ in Eq.(22), respectively. Thus, solving problem (22) changes to solve the minimization problem of the following equation:

$$\frac{1}{2} \|\boldsymbol{x} - \boldsymbol{y}\|^2 \quad s.t.\ \boldsymbol{x}^T \mathbf{1} = 1, \boldsymbol{x} \geq 0. \tag{23}$$

The solution of Eq.(23) can be obtained through Algorithm 2. To sum up, $\boldsymbol{u}$ in Eq.(22) can be solved by Algorithm 2 [29].

Algorithm 2 Algorithm to solve $\min_{\boldsymbol{x}} \frac{1}{2} \|\boldsymbol{x} - \boldsymbol{y}\|^2$ $s.t.\ \boldsymbol{x}^T \mathbf{1} = 1, \boldsymbol{x} \geq 0$.

Input: $\boldsymbol{y} \in \mathbb{R}^c$.
1. Sort $\boldsymbol{y}$ into $\boldsymbol{t} : t_1 \geq t_2 \geq \ldots \geq t_c$.
2. Find $\rho = \max \left\{ 1 \leq j \leq c : u_j + \frac{1}{j}(1 - \sum_{i=1}^j u_i) > 0 \right\}$.
3. Define $\lambda = \frac{1}{\rho}(1 - \sum_{i=1}^\rho u_i)$.
Output: $\boldsymbol{x}$ $s.t.$ $x_i = \max\{y_i + \lambda, 0\}$, $i = 1, \ldots, c$.

In summary, when $m = 1$, the algorithm for optimizing problem (11) is summarized in Algorithm 3.

Algorithm 3 Algorithm to solve Eq.(11).

Input: Data matrix $X \in \mathbb{R}^{n \times d}$, the number of clusters c, similarity matrix $W \in \mathbb{R}^{n \times n}$, parameters α, λ, and σ.
repeat:
1. Calculate $\boldsymbol{c}_k, (k = 1, 2, \ldots, c)$ by Eq.(20) to update C.
2. Calculate $\boldsymbol{u}_i, (i = l + 1, l + 2, \ldots, n)$ by Eq.(22) to update U.
until converge or exceed the maximum iterations
Output: $U \in \mathbb{R}^{n \times c}$.

4.2.2 Optimization algorithm to solving problem (12)

According to Algorithm 1, solving Eq.(12) is equivalent to solving Eq.(18).
Update C by Fixing U: Problem (18) can be written as

$$\min_{c_k} \sum_{i=1}^n \sum_{k=1}^c h_{ik} \|x_i - c_k\|_2^2 u_{ik}^2. \tag{24}$$

By taking the derivative of Eq.(24) w.r.t. c_k and setting the derivative to zero, we have

$$c_k = \frac{\sum_{i=1}^n h_{ik} u_{ik}^2 x_i}{\sum_{i=1}^n h_{ik} u_{ik}^2}. \tag{25}$$

Update U by Fixing C: Problem (18) becomes

$$\begin{aligned} \min_U \sum_{i=1}^n \sum_{k=1}^c e_{ik} u_{ik}^2 + \lambda u_{ik}^2 + \frac{\alpha}{2} Tr\left(U^T L_W U\right), \\ s.t. \sum_{k=1}^c u_{ik} = 1, u_{ik} \geq 0, U_l = Y_l, \end{aligned} \tag{26}$$

where $e_{ik} = h_{ik} \|x_i - c_k\|_2^2$ in Eq.(26).

To ensure that U always satisfies the condition of $u_{ik} \geq 0$, similarly to solve Eq.(21), the process of solving U is carried out as follows:

For a sample point $\boldsymbol{x}_i \in \mathbb{R}^d$. The affinity vectors are $\boldsymbol{w}_i = (w_{ij})$ and $\boldsymbol{e}_i = (e_{ik})$, where w_{ij} are the affinity between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, $j = 1, \ \ldots, \ n$, and $e_{ik} = h_{ik} \|\boldsymbol{x}_i - \boldsymbol{c}_k\|_2^2$, $k = 1, \ldots, c$, respectively. After dropping the constant terms, the optimization problem (18) is reduced to n quadratic programs where each quadratic program has c variables as follows:

$$\begin{aligned} \min_{\boldsymbol{u}_i} \ & \frac{1}{2} \|\boldsymbol{p}_i \odot \boldsymbol{u}_i - \boldsymbol{q}_i\|^2, \\ s.t. & \boldsymbol{u}_i^T \mathbf{1}_c = 1, \boldsymbol{u}_i \geq 0, i = 1, \ \ldots, \ n, \end{aligned} \tag{27}$$

where $\boldsymbol{p}_i = \sqrt{\boldsymbol{e}_i + \lambda + \frac{\alpha}{2} \sum_{j \neq i}^{n} w_{ij}}$, $\boldsymbol{q}_i = \frac{\alpha \sum_{j \neq i}^{n} w_{ij} \boldsymbol{u}_j}{2\boldsymbol{p}_i}$, and $\odot$ denotes the dot product operation between vectors. Thus, the $\boldsymbol{u}_i$ can be solved by Algorithm 4. The proof and convergence analysis of Algorithm 4 will be given in the Appendix.

Algorithm 4 Algorithm to solve Eq.(27).

Input: $\boldsymbol{p}_i \in \mathbb{R}^{1 \times c}, \ \boldsymbol{q}_i \in \mathbb{R}^{1 \times c}$.

1. Sort $\boldsymbol{p}_i \odot \boldsymbol{q}_i$ into $\boldsymbol{t} : t_1 \geq t_2 \geq \ldots \geq t_c$.
2. Find $\rho = \max \left\{ 1 \leq j \leq c : t_j + \frac{1 - \sum_{l=1}^{j} \frac{t_l}{w_l^2}}{\sum_{l'=1}^{j} \frac{1}{w_{l'}^2}} > 0 \right\}$.
3. Define $\alpha = \frac{1 - \sum_{l=1}^{\rho} \frac{t_l}{w_l}}{\sum_{l'=1}^{\rho} \frac{1}{w_{l'}^2}}$.

Output: $\boldsymbol{u}_i \ s.t. \ \boldsymbol{u}_i = \max\{\boldsymbol{p}_i \odot \boldsymbol{q}_i + \alpha, 0\}, \ i = 1, \ldots, c.$

In summary, the algorithm for optimizing problem (12) is summarized in Algorithm 5.

Algorithm 5 Algorithm to solve Eq.(12).

Input: Data matrix $X \in \mathbb{R}^{n \times d}$, the number of clusters c, similarity matrix $W \in \mathbb{R}^{n \times n}$, parameters α, λ, and σ.

repeat:

1. Calculate $\boldsymbol{c}_k, (k = 1, \ 2, \ \ldots, \ c)$ by Eq.(25) to update C.
2. Calculate $\boldsymbol{u}_i, (i = l + 1, \ l + 2, \ \ldots, \ n)$ by Eq.(27) to update U.

until converge or exceed the maximum iterations

Output: $U \in \mathbb{R}^{n \times c}$.

4.3 Computation complexity analysis

The time complexity of the proposed algorithm is mainly divided into two parts: calculating the center matrix C and the membership matrix U, respectively. The time complexity of update C is $O(ndc)$, and the time complexity of update U is $O((n-l)clogc)$, where n is the number of samples, d is the dimensions of the samples, c is the number of clusters, and l is the number of labels. Since $logc << d$, the final time complexity is $O(tndc)$, where t is the number of iterations. The time complexity of the FCM and PCFCM algorithms [17] is $O(tndc)$.

5 Experiments

In this section, we evaluate the performance of the proposed algorithm on seven public datasets and compare with four state-of-the-art related semi-supervised clustering algorithms to verify the effectiveness of the proposed algorithm. In addition, we discuss the sensitivity of each parameter of the model and the convergence of the algorithm.

5.1 Datasets

In this paper, experiments are conducted on the following eight datasets, namely YALE dataset, wine dataset, TOX-171 dataset, Isolet dataset, dbworld dataset, colon dataset, Umist dataset and Cdata04 dataset. The detailed information of the above datasets is shown in Table 2.

Table 2: Information of the datasets

Datasets	Size(n)	Dimensions(d)	Class(c)
YALE	165	1024	15
Tox-171	171	5748	4
dbworld	64	4702	2
colon	62	2000	2
wine	178	13	3
Isolet	1560	617	2
Umist	575	10304	20
Cdata04	100	2	4

5.2 Comparison algorithm

In order to evaluate the effectiveness of the proposed algorithm, the proposed algorithm is compared with SFS [15], SEE [20], CSFS [4], ℓ_1-graph [19], NCM [10], SSWL [7] and FCM[11], which are briefly described, except for FCM, as follows:

(1) SFS: it is a semi-supervised feature selection method, which emploies the generalized uncorrelated constraint and the manifold structure into ridge regression. The manifold structure and closed-form solution can maintain data's topology information.

(2) SEE: it is a semi-supervised feature selection method, which uses σ-loss function to relax the linear embedding model. The new adaptive loss has the advantages from both ℓ_1 and ℓ_2. And thus SEE is robust to outlier under Laplacian distribution.

(3) CSFS: it is a convex semi-supervised multi-label feature selection algorithm, which uses $\ell_{2,1}$-norm to conduct sparse feature selection. Due to low computational burden, CSFS can handle the large-scale datasets.

(4) ℓ_1-graph: it is a semi-supervised graph model, which uses ℓ_1-norm of spectral embedding to learn manifold information. By minimizing the ℓ_1-norm, the algorithm gets a sparse solution, and presents a more clear cluster structure.

(5) NCM: it is an extension of FCM, which introduces ambiguity rejection and distance rejection to objective function. By introducing these two inhibitors, the model can handle the uncertainty caused by imprecise and incomplete clustering data.

(6) SSWL: it is a semi-supervised weak-label learning method, which considers both instance similarity and label similarity in the model. By employing multiple models integration method, the problem of multi-label learning with insufficient label information is solved, and the robustness of the model is improved.

5.3 Evaluation metrics

The performance of the algorithm is evaluated by the following two clustering metrics: Clustering Accuracy(ACC) and Normalized Mutual Information(NMI) [39] . They are defined as follows.

Clustering Accuracy(ACC) reflects the correspondence between data and classes, which is defined as

$$ACC = \frac{1}{n}\sum_{i=1}^{n}\delta\left(map\left(c_i\right), y_i\right),$$

where c_i is the class to which the i-th sample point belongs, that is, the prediction label. y_i is the true label of the i-th sample point, and $map\left(\bullet\right)$ is the optimal map function that maps a data to a class. $\delta\left(x, y\right)$ is the delta function. When $x = y$, $\delta\left(x, y\right) = 1$, otherwise, $\delta\left(x, y\right) = 0$. The result of ACC is in [0,1], and the higher the value indicates, the better the algorithm performs.

Normalized Mutual Information(NMI) is used to determine the quality of the clusters. NMI is defined as follows

$$NMI = \frac{\sum_{i=1}^{k}\sum_{j=1}^{k} n_{ij} log\frac{n_{ij}}{n_i \hat{n}_j}}{\sqrt{\left(\sum_{i=1}^{k} n_i log\frac{n_i}{n}\right)\left(\sum_{j=1}^{k} \hat{n}_j log\frac{\hat{n}_j}{n}\right)}},$$

where n_{ij} is the number of sample points in the cluster C_i intersects with class Y_j, and n_i is the number of sample points in the cluster C_i. The higher the value of the NMI means the better the performance of the algorithm.

5.4 Parameter discussion

During the experiments, refer to the parameter settings given in the corresponding literature of each algorithm, and set the parameters of each algorithm in Table 3. In our method(PC-RSFCM), all parameters are set in the range $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2, 10^3\}$ to find the optimal value of parameters. According to the experimental result, we set $\alpha = 0.01, \lambda = 0.1, \sigma = 0.1$ for datasets used in this paper.

Table 3: The parameter settings of comparison algorithms and our models

Algorithm	Parameter setting
SFS	$\beta = 0.1, \lambda = 0.001$
SEE	$\gamma = 1, \sigma = 0.1$
CSFS	$\mu = 1$
ℓ_1-graph	no parameter
NCM	$\omega_1 = 0.1, \omega_2 = 0.8, \omega_3 = 0.9, \delta = 0.8$
SSWL	$\alpha = 1.6, \beta = 0.06, \zeta = 0.06$
our models	$\alpha = 0.01, \lambda = 0.1, \sigma = 0.1$

5.5 Experimental on public datasets

For each dataset, we random select the ratios of labeled samples from data matrix with 10%, 20%, 30%, 40%, 50%, 60%, respectively, and the rest are unlabeled samples. Experiment is conducted for all the comparison algorithms according to the parameter settings given in Section 5.4, and the experiment is repeated 10 times to take the average value and standard deviation. Finally, the ACC and NMI results of our algorithm and the comparison algorithms on the five datasets are shown in Fig.2 (a)-(e) and Fig.3 (a)-(e). Fig.2 shows the average clustering accuracy and standard deviation. The red line and blue line indicate the clustering accuracy results of our two algorithms PCRSFCM($m = 1$) and PC-RSFCM($m = 2$), respectively. Fig.3 shows the results of the normalized mutual information and standard deviation, and the red and blue lines in the graph indicate the normalized mutual information results for our two algorithms PC-RSFCM($m = 1$) and PC-RSFCM($m = 2$), respectively. From the two figures, the lines representing the two algorithms PC-RSFCM($m = 1$) and PC-RSFCM($m = 2$) are basically located above the other colored lines and show an overall upward trend, indicating that the PC-RSFCM algorithm basically outperforms the rest of the comparison algorithms.

Meanwhile, we give the value of ACC when exist 10%, 30% and 50% labeled samples in Tables 4, 5, and 6. The all results are repeated the experiment 10 times. We also calculated the ACC results of the traditional FCM algorithm and presented them in these tables to demonstrate the effectiveness of the proposed algorithms. In these tables, the bold ones represent the optimal results. From those tables, it can be seen that our proposed models perform best when there are 30% and 50% labeled samples. When the number of labeled samples is 10%, our proposed models exhibit the best results on the dbworld, colon, and wine datasets. And PC-RSFCM($m = 1$) shows suboptimal results on the YALE and Umist datasets. From Tables 3-5, we have some interesting analysis as follows:

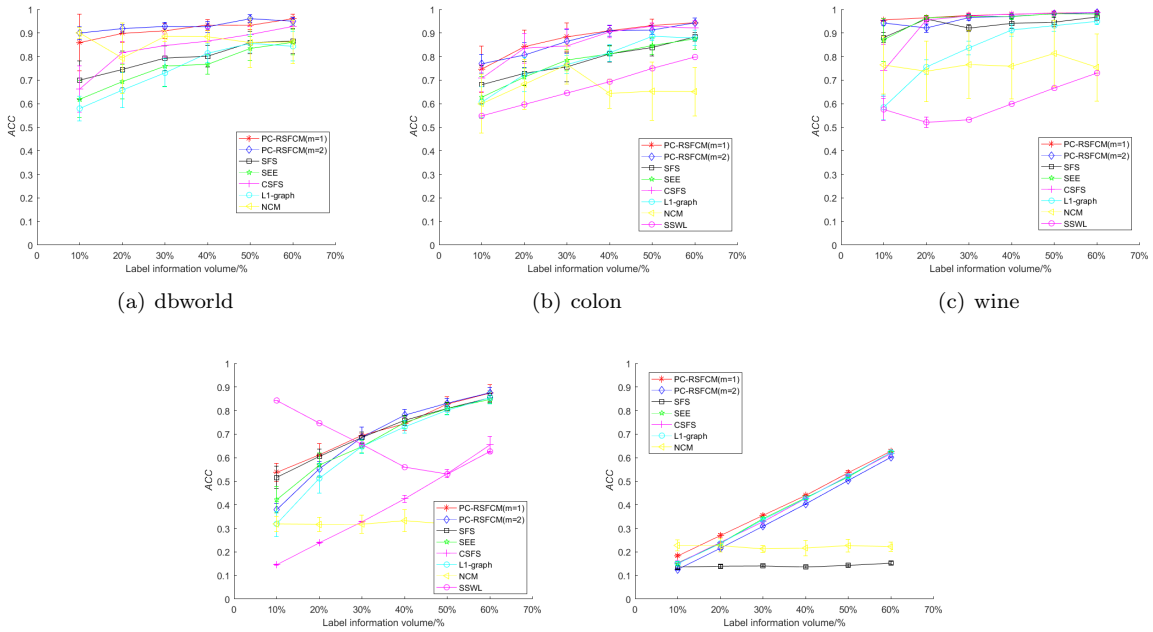
(1) With the increase of label information, our two algorithms perform better than other comparison algorithms. Since our models fully take advantage of priori information in the objective functions and constraint conditions.

(2) Our methods work better in the robustness against noises and outliers that are inevitable in real datasets. Since our models use σ-norm to alleviate the impact of noises and outliers.

(3) In image datasets, such as YALE and Umist in Tabel 3, our models obtain some suboptimal results. The reason may be high data dimensionality and low data volume.

Table 4: The value of ACC when exciting 10% labeled samples (MEANS±STD)

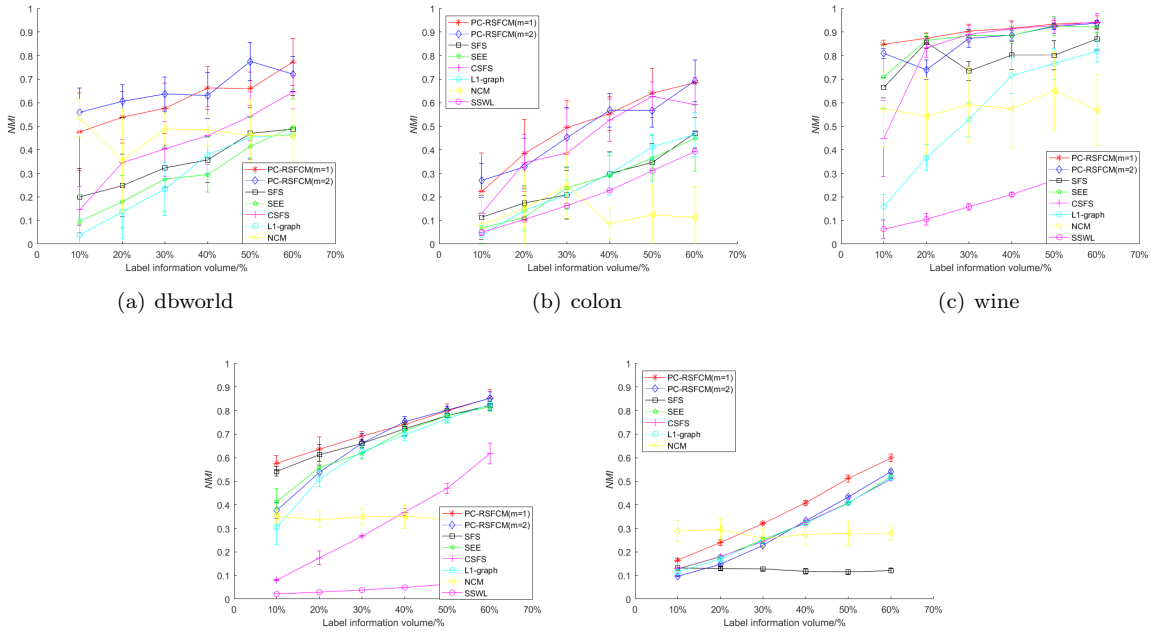
Dataset	SFS	SEE	CSFS	ℓ_1-graph	NCM	SSWL	FCM	our($m = 1$)	our($m = 2$)
YALE	0.5164±0.0482	0.4218±0.0560	0.1455±0.0027	0.3194±0.0531	0.3188±0.0320	**0.8428 ± 0.0000**	0.1576±0.0180	0.5376±0.0376	0.3794±0.0269
dbworld	0.7000±0.0811	0.6188±0.0776	0.6625±0.0984	0.5797±0.0534	**0.9000 ± 0.0234**	–	0.8844±0.0187	0.8594±0.1206	**0.9000 ± 0.0272**
colon	0.6806±0.0822	0.6274±0.0891	0.7081±0.0579	0.6065±0.0658	0.6000±0.1255	0.5484±0.0000	0.6048±0.0654	0.7468±0.0989	**0.7694 ± 0.0402**
wine	0.8809±0.0230	0.8713±0.0911	0.7410±0.1109	0.5848±0.0521	0.7640±0.1233	0.5758±0.0474	0.9494±0.0000	**0.9556 ± 0.0064**	0.9427±0.0093
Umist	0.1355±0.0078	0.1537±0.0028	0.1520±0.0111	0.1503±0.0100	**0.2273 ± 0.0237**	–	0.1130±0.0000	0.1830±0.0040	0.1259±0.0028



(a) dbworld (b) colon (c) wine

(d) YALE (e) Umist

Figure 2: ACC results of all methods at different number of labeled instances



(a) dbworld (b) colon (c) wine

(d) YALE (e) Umist

Figure 3: NMI results of all methods at different number of labeled instances

Table 5: The value of ACC when exciting 30% labeled samples (MEANS±STD)

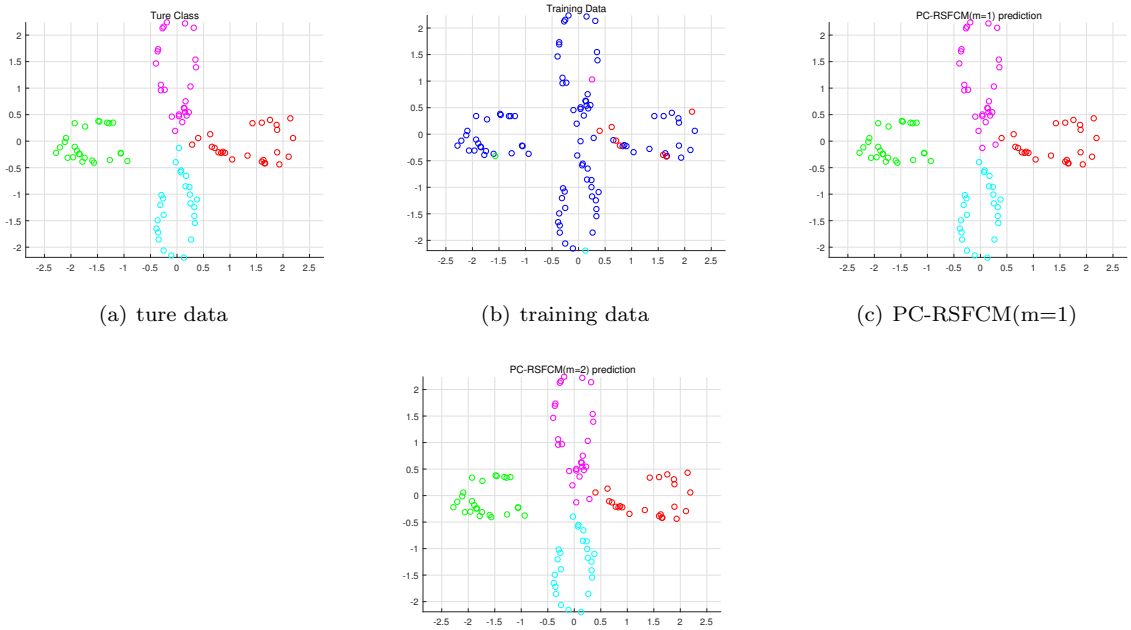
Dataset	SFS	SEE	CSFS	ℓ_1-graph	NCM	SSWL	FCM	our($m=1$)	our($m=2$)
YALE	0.6861±0.0236	0.6473±0.0273	0.3279±0.0033	0.6479±0.0299	0.3176±0.0393	0.6561±0.0003	0.1576±0.0180	**0.6952 ± 0.0349**	0.6873±0.0424
dbworld	0.7937±0.0536	0.7594±0.0848	0.8469±0.0483	0.7312±0.0588	0.8859±0.0232	–	0.8844±0.0187	0.9094±0.0319	**0.9281 ± 0.0174**
colon	0.7565±0.0631	0.7855±0.0456	0.8452±0.0740	0.7661±0.0390	0.7661±0.0836	0.6452±0.0000	0.6048±0.0654	**0.8839 ± 0.0598**	0.8645±0.0506
wine	0.9208±0.0143	0.9708±0.0083	0.9725±0.0105	0.8371±0.0296	0.7663±0.1444	0.5318±0.0000	0.9494±0.0000	**0.9747 ± 0.0098**	0.9657±0.0124
Umist	0.1404±0.0049	0.3412±0.0084	0.3280±0.0047	0.3367±0.0065	0.2130±0.0151	–	0.1130±0.0000	**0.3551 ± 0.0060**	0.3089±0.0051

Table 6: The value of ACC when exciting 50% labeled samples (MEANS±STD)

Dataset	SFS	SEE	CSFS	ℓ_1-graph	NCM	SSWL	FCM	our($m=1$)	our($m=2$)
YALE	0.8091±0.0255	0.8085±0.0250	0.5326±0.0178	0.8018±0.0192	0.3188±0.0230	0.5305±0.0000	0.1576±0.0180	0.8279±0.0296	**0.8315 ± 0.0199**
dbworld	0.8594±0.0474	0.8344±0.0500	0.8938±0.0348	0.8562±0.0375	0.8594±0.1048	–	0.8844±0.0187	0.9328±0.0186	**0.9609 ± 0.0160**
colon	0.8387±0.0353	0.8468±0.0362	0.9290±0.0129	0.8861±0.0192	0.6532±0.1231	0.7500±0.0000	0.6048±0.0654	**0.9323 ± 0.0258**	0.9129±0.0194
wine	0.9461±0.0197	0.9820±0.0106	0.9837±0.0081	0.9315±0.0230	0.8135±0.1407	0.6667±0.0000	0.9494±0.0000	**0.9848 ± 0.0044**	0.9815±0.0036
Umist	0.1431±0.0060	0.5183±0.0029	0.5235±0.0044	0.5217±0.0082	0.2271±0.0265	–	0.1130±0.0000	**0.5363 ± 0.0084**	0.5037±0.0021

5.6 Experimental on artificial datasets

Meanwhile, we conduct experiments for the proposed algorithm on the artificial dataset Cdata04, and the results of the experiments are shown in Fig.4. Fig.4(a) represents the distribution of true sample data. The blue dots represent the unlabeled data, and the other colored dots represent the labeled data in each class, respectively, in Fig.4(b). Fig.4(c) shows the clustering results of PC-RSFCM($m=1$), and Fig.4(d) shows the clustering results of PC-RSFCM($m=2$). It can be seen that the Fig.4(c) and Fig.4(d) are basically consistent with the Fig.4(a), and there are only two sample points with clustering errors in both algorithms. This proves that the proposed model has a good clustering performance.



(a) ture data (b) training data (c) PC-RSFCM(m=1)

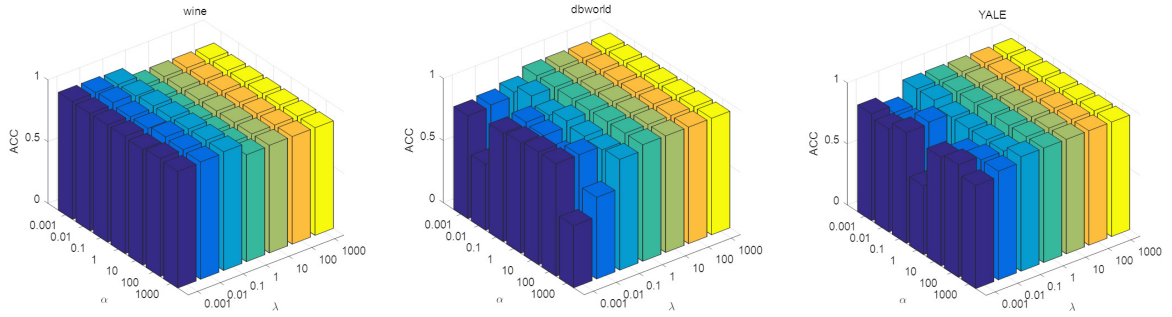
(d) PC-RSFCM(m=2)

Figure 4: Experimental results on Cdata04 dataset

5.7 Sensitivity analysis of parameters

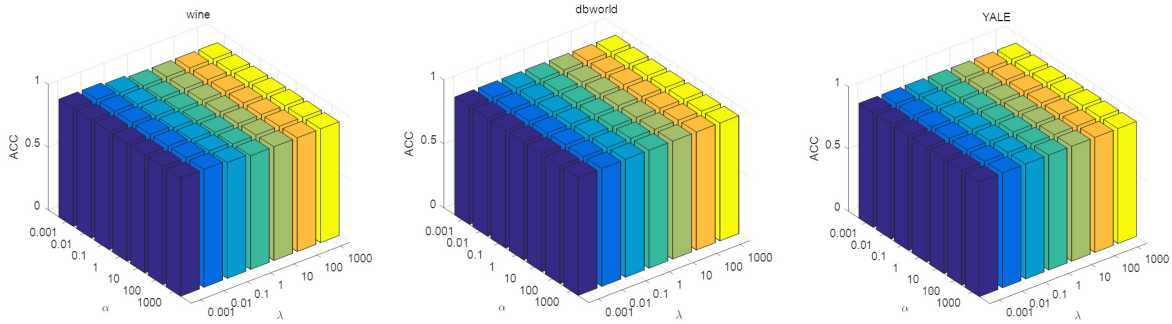
Our model contains two tradeoff parameters α and λ. In the presence of 10% of the labeled information, we analyze the influence of tradeoff parameters to the performance of PC-RSFCM algorithm when parameters α and λ value are set empirically in the range $\{10^{-3}, 10^{-2}, 10^{-1}, 1, 10, 10^2, 10^3\}$. In addition, set the adaptive parameter σ to 1 and the number of neighbors k to 3. Fig.5 (a)-(c) and Fig.6 (a)-(c) show the effects of parameter α and λ on the wine dataset, dbworld dataset and YALE dataset respectively. It can be observed from Fig.5 that the model PC-RSFCM($m=1$) is a

little sensitive to the parameters. For the wine dataset, the parameters α has no effect on the model, parameter λ has a smaller effect on the model, and the model works best when λ takes values of $\{0.1, 0.01, 0.001\}$. For the dbworld dataset and YALE dataset, the parameters α and λ both have a small effect on the model in Fig.5(b) and Fig.5(c). Especially, α is sensitive when λ is small. When $m = 2$, as can be observed from Fig.6, the ACC histogram has almost no ups and downs for any α and λ, which indicates that the algorithm PC-RSFCM($m = 2$) is insensitive to the parameters.



(a) wine (b) dbworld (c) YALE

Figure 5: when $m = 1$, ACC results on wine, dbworld and YALE at different parameters α and λ



(a) wine (b) dbworld (c) YALE

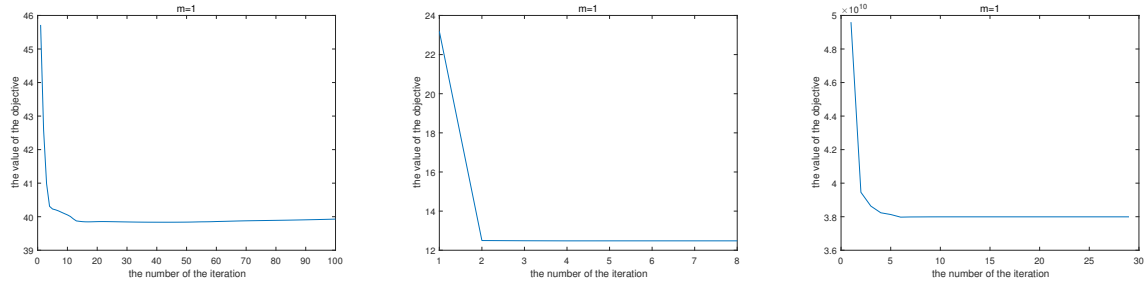
Figure 6: when $m = 2$, ACC results on wine, dbworld and YALE at different parameters α and λ

5.8 Convergence analysis

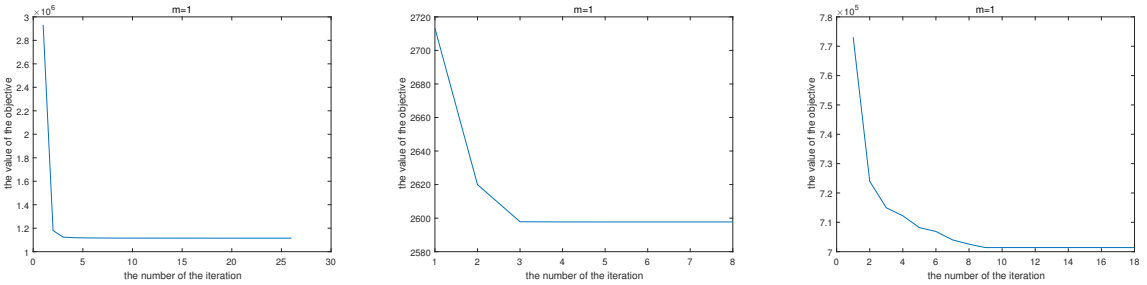
In this paper, the convergence of the proposed model is analyzed on six datasets, namely YALE, wine, TOX-171, Isolet, dbworld and colon. Their objective function values are shown in Fig.7 (a)-(f) and Fig.8 (a)-(f). It can be seen that all the curves in Fig.7 and Fig.8 rapidly decrease and then stabilize, which indicates that the algorithm PC-RSFCM has a good convergence performance.

6 Conclusion

This paper presents a new robust semi-supervised fuzzy clustering algorithm, to obtain a more accurate clustering result. This algorithm utilizes the pairwise constraints as supervisory information to guide the clustering process, which can improve the result in traditional FCM clustering algorithm. Firstly, in order to alleviate the influence of the noise, robustness is considered in our models by replacing the F-norm loss function with adaptive loss function. Secondly, to capture global structures in datasets, graph Laplacian regularization is added in our models. Thirdly, a simple and efficient optimization algorithm is proposed to ensure that the elements of the membership matrix must be nonnegative. Concretely, the optimization algorithm is generalized to the solution of the model with different coefficients of independent variables, and the convergence of the optimization algorithm is proved. Finally, extensive experiments

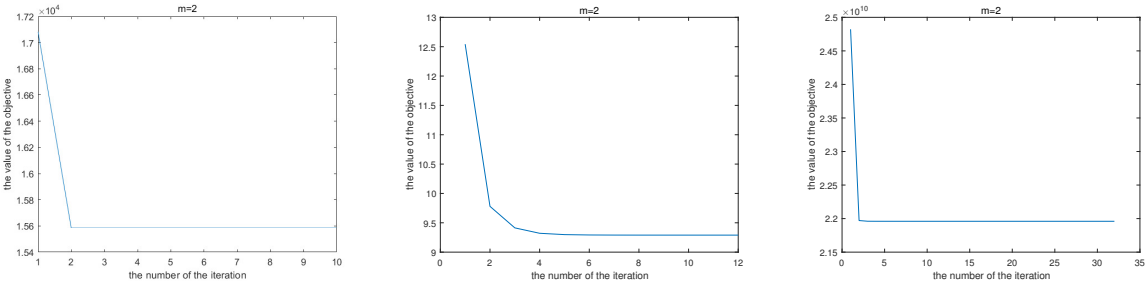


(a) YALE (b) wine (c) TOX-171

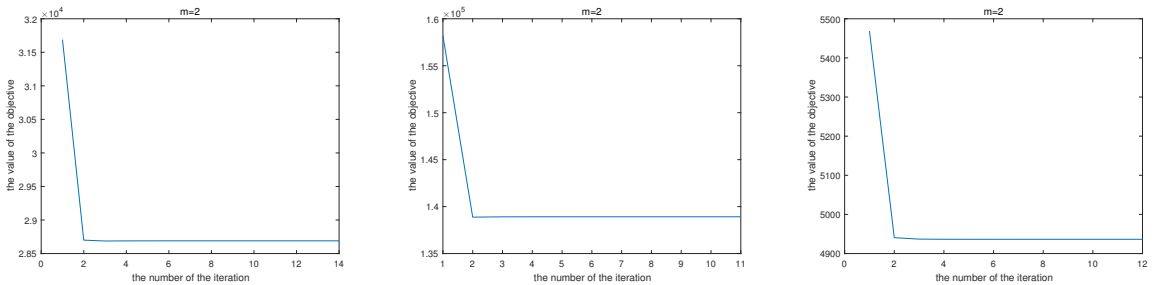


(d) Isolet (e) dbworld (f) colon

Figure 7: when $m = 1$, the iterative convergence graph on six datasets



(a) YALE (b) wine (c) TOX-171



(d) Isolet (e) dbworld (f) colon

Figure 8: when $m = 2$, the iterative convergence graph on six datasets

are conducted on artificial datasets and public datasets, such as text datasets, voice datasets, image datasets, chemical datasets and biological datasets. The experiment results show that the proposed algorithm is generally superior to the comparison algorithms. In the future, we will consider self-paced method in our models, which gradually involves easy samples to difficult samples in clustering process, so that we can obtain a strong fuzzy clustering model.

A Proof of Algorithm 4

Theorem A.1. *The optimal solution to problem (27) takes the form of*

$$u_i = \max\{p_i \odot q_i + \alpha, 0\},\ i = 1, \ldots, c.$$

Proof. Using the standard KKT condition for Eq.(27), the Lagrange function for problem (27) is

$$L(\boldsymbol{u}, \alpha, \boldsymbol{\beta}) = \frac{1}{2}\|\boldsymbol{p}_i \odot \boldsymbol{u}_i - \boldsymbol{q}_i\|^2 - \alpha\left(\boldsymbol{u}^T\mathbf{1} - 1\right) - \boldsymbol{\beta}^T\boldsymbol{u},$$

where α and $\boldsymbol{\beta} = [\beta_1, \beta_2, \ldots, \beta_c]^T$ are the Lagrange multipliers for the equation and inequality constraints, respectively. For the optimal solution $\boldsymbol{u}$, the following KKT conditions are satisfied

$$p_l^2 u_l - p_l q_l - \alpha - \beta_l = 0,\ \ l = 1, \ldots, c \tag{28}$$

$$\sum_{l=1}^{c} u_l = 1, \tag{29}$$

$$u_l \geq 0,\ \ l = 1, \ldots, c \tag{30}$$

$$\beta_l \geq 0,\ \ l = 1, \ldots, c \tag{31}$$

$$u_l \beta_l = 0.\ \ l = 1, \ldots, c \tag{32}$$

When $u_l > 0$, according to Eq.(32), we have $\beta_l = 0$. Substituting $\beta_l = 0$ into Eq.(28) yields

$$u_l = \frac{p_l q_l + \alpha}{p_l^2} = \frac{q_l}{p_l} + \frac{\alpha}{p_l^2} > 0.$$

When $u_l = 0$, according to Eq.(28), we can obtain $p_l q_l + \alpha = -\beta_l \leq 0$. Both sides of the $\frac{q_l}{p_l} + \frac{\alpha}{p_l^2} > 0$ are multiplied by p_l^2, and we can get $p_l q_l + \alpha > 0$, then $\frac{q_l}{p_l} + \frac{\alpha}{p_l^2} > 0$ is equal to $p_l q_l + \alpha > 0$. Thus, we obtain the optimal solution of u_l

$$u_l = (p_l q_l + \alpha)_+,$$

where, $(\bullet)_+ = \max(\bullet, 0)$. Obviously, the component of the optimal solution $\boldsymbol{u}$ is 0 corresponding to the smaller component of $\boldsymbol{q}$. Without loss of generality, Sort the components of $\boldsymbol{p} \odot \boldsymbol{q}$ and use the same order for $\boldsymbol{u}$, i.e.

$$p_1 q_1 \geq \ldots \geq p_\rho q_\rho \geq p_{\rho+1} q_{\rho+1} \geq \ldots \geq p_c q_c,$$

$$u_1 \geq \ldots \geq u_\rho > u_{\rho+1} = \ldots = u_c,$$

and that $u_1 \geq \ldots \geq u_\rho > 0, u_{\rho+1} = \ldots = u_c = 0$. In other words, ρ is the number of positive components in the solution $\boldsymbol{u}$. From Eq.(29) we have

$$1 = \sum_{l=1}^{c} u_l = \sum_{l=1}^{\rho} u_l = \sum_{l=1}^{\rho} (p_l q_l + \alpha),$$

which gives $\alpha = \frac{1 - \sum_{l=1}^{j} \frac{p_l q_l}{p_l^2}}{\sum_{l'=1}^{j} \frac{1}{p_{l'}^2}}$. Therefore, ρ is the key to the solution, once we know ρ which has only c possible values, α can be calculated by adding α to each component of q and thresholding as at the end of algorithm 4. (It is easy to verify that this solution does indeed satisfy all KKT conditions). In the algorithm, we execute the tests for $j = 1, \ldots, c$ if $t_j = p_j q_j + \frac{1 - \sum_{l=1}^{j} \frac{p_l q_l}{p_l^2}}{\sum_{l'=1}^{j} \frac{1}{p_{l'}^2}} > 0$. Theorem A.2 proves that the number of times this test result is positive is exactly ρ. The following theorem is essentially Lemma 3 of literature [24]. □

Theorem A.2. *ρ is the number of positive components in the solution $\boldsymbol{u}$, then*

$$\rho = \max\left\{1 \le j \le c : p_j q_j + \frac{1 - \sum_{l=1}^{j} \frac{p_l q_l}{p_l^2}}{\sum_{l'=1}^{j} \frac{1}{p_{l'}^2}} > 0\right\}.$$

Proof. According to the KKT conditions (28)-(32) that $\alpha_\rho = \frac{1-\sum_{l=1}^{\rho} \frac{p_l q_l}{p_l^2}}{\sum_{l'=1}^{\rho} \frac{1}{p_{l'}^2}}$, $p_i q_i + \alpha > 0 \; for \; i = 1, \ldots, \rho$ and $p_i q_i + \alpha \le 0 \; for \; i = \rho + 1, \ldots, c$. It will be shown below that for $j = 1, \ldots, c$, $p_i q_i + \alpha$ will remain positive until $j = \rho$ and then remains non-positive, i.e., $p_j q_j + \frac{1-\sum_{l=1}^{j} \frac{p_l q_l}{p_l^2}}{\sum_{l'=1}^{j} \frac{1}{p_{l'}^2}} > 0$ for $j \le \rho$, and $p_j q_j + \frac{1-\sum_{l=1}^{j} \frac{p_l q_l}{p_l^2}}{\sum_{l'=1}^{j} \frac{1}{p_{l'}^2}} \le 0$ for $j > \rho$.

(1) For $j = \rho$, we have

$$p_\rho q_\rho + \frac{1 - \sum_{l=1}^{\rho} \frac{p_l q_l}{p_l^2}}{\sum_{l'=1}^{\rho} \frac{1}{p_{l'}^2}} = p_\rho q_\rho + \alpha = x_\rho > 0.$$

(2) For $j < \rho$, we have

$$\begin{aligned}
p_j q_j + \frac{1 - \sum_{l=1}^{j} \frac{p_l q_l}{p_l^2}}{\sum_{l'=1}^{j} \frac{1}{p_{l'}^2}} &= \frac{1}{\sum_{l=1}^{j} \frac{1}{p_l^2}} \left(\sum_{l=1}^{j} \frac{p_j q_j}{p_l^2} + 1 - \sum_{l=1}^{j} \frac{p_l q_l}{p_l^2} \right) \\
&= \frac{1}{\sum_{l=1}^{j} \frac{1}{p_l^2}} \left(\sum_{l=1}^{j} \frac{p_j q_j}{p_l^2} + \sum_{i=j+1}^{\rho} \frac{p_i q_i}{p_i^2} + 1 - \sum_{l=1}^{\rho} \frac{p_l q_l}{p_l^2} \right) \\
&= \frac{1}{\sum_{l=1}^{j} \frac{1}{p_l^2}} \left(\sum_{l=1}^{j} \frac{p_j q_j}{p_l^2} + \sum_{i=j+1}^{\rho} \frac{p_i q_i}{p_i^2} + \sum_{l=1}^{\rho} \frac{\alpha}{p_l^2} \right) \\
&= \frac{1}{\sum_{l=1}^{j} \frac{1}{p_l^2}} \left(\sum_{l=1}^{j} \frac{1}{p_l^2} (p_j q_j + \alpha) + \sum_{i=j+1}^{\rho} \left(\frac{p_i q_i}{p_i^2} + \frac{\alpha}{p_i^2} \right) \right) \\
&= \frac{1}{\sum_{l=1}^{j} \frac{1}{p_l^2}} \left(\sum_{l=1}^{j} \frac{1}{p_l^2} (p_j q_j + \alpha) + \sum_{i=j+1}^{\rho} \frac{1}{p_i^2} (p_i q_i + \alpha) \right).
\end{aligned}$$

Since $p_i q_i + \alpha > 0$ for $i = j, \ldots, \rho$, we have $p_j q_j + \frac{1-\sum_{l=1}^{j} \frac{p_l q_l}{p_l^2}}{\sum_{l'=1}^{j} \frac{1}{p_{l'}^2}} > 0$.

(3) For $j > \rho$, we have

$$\begin{aligned}
p_j q_j + \frac{1 - \sum_{l=1}^{j} \frac{p_l q_l}{p_l^2}}{\sum_{l'=1}^{j} \frac{1}{p_{l'}^2}} &= \frac{1}{\sum_{l=1}^{j} \frac{1}{p_l^2}} \left(\sum_{l=1}^{j} \frac{p_j q_j}{p_l^2} + 1 - \sum_{l=1}^{j} \frac{p_l q_l}{p_l^2} \right) \\
&= \frac{1}{\sum_{l=1}^{j} \frac{1}{p_l^2}} \left(\sum_{l=1}^{j} \frac{p_j q_j}{p_l^2} + 1 - \sum_{l=1}^{\rho} \frac{p_l q_l}{p_l^2} - \sum_{i=\rho+1}^{j} \frac{p_i q_i}{p_i^2} \right) \\
&= \frac{1}{\sum_{l=1}^{j} \frac{1}{p_l^2}} \left(\sum_{l=1}^{j} \frac{p_j q_j}{p_l^2} + \sum_{l=1}^{\rho} \frac{\alpha}{p_l^2} - \sum_{i=\rho+1}^{j} \frac{p_i q_i}{p_i^2} \right) \\
&= \frac{1}{\sum_{l=1}^{j} \frac{1}{p_l^2}} \left(\sum_{l=1}^{\rho} \frac{p_\rho q_\rho}{p_l^2} + \sum_{i=\rho+1}^{j} \frac{p_j q_j}{p_i^2} + \sum_{l=1}^{\rho} \frac{\alpha}{p_l^2} - \sum_{i=\rho+1}^{j} \frac{p_i q_i}{p_i^2} \right) \\
&= \frac{1}{\sum_{l=1}^{j} \frac{1}{p_l^2}} \left(\sum_{l=1}^{\rho} \frac{1}{p_l^2} (p_\rho q_\rho + \alpha) + \sum_{i=\rho+1}^{j} \frac{1}{p_i^2} (p_j q_j - p_i q_i) \right).
\end{aligned}$$

Notice $p_\rho q_\rho + \alpha \leq 0$ for $j > \rho$, and $p_j q_j \leq p_i q_i$ for $j \geq i$ since pq is sorted, therefore $p_j q_j + \frac{1-\sum_{l=1}^{j} \frac{p_l q_l}{p_l^2}}{\sum_{l'=1}^{j} \frac{1}{p_{l'}^2}} < 0$. □

B Convergence analysis of Algorithm 4

The algorithm finds a locally optimal solution, and in order to prove its convergence, the lemma introduced by literature [18] need to be utilized.

Lemma B.1. *For any positive real number u and v, the following inequality holds:*

$$\sqrt{u} - \frac{u}{2\sqrt{v}} \leq \sqrt{v} - \frac{v}{2\sqrt{v}}.$$

Theorem B.2. *In Algorithm 4, the updated U will cause the objective value of problem (27) to decrease until convergence.*

Proof. Suppose that after each iteration, the U is $\widetilde{U}$, then it is easy to obtain that:

$$\frac{\|p_i \odot \widetilde{u}_i - \widetilde{q}_i\|^2}{2\,|p_i \odot u_i - q_i|} \leq \frac{\|p_i \odot u_i - q_i\|^2}{2\,|p_i \odot u_i - q_i|}. \tag{33}$$

According to Lemma B.1, we can obtain

$$|p_i \odot \widetilde{u}_i - \widetilde{q}_i| - \frac{\|p_i \odot \widetilde{u}_i - \widetilde{q}_i\|^2}{2\,|p_i \odot u_i - q_i|} \leq |p_i \odot u_i - q_i| - \frac{\|p_i \odot u_i - q_i\|^2}{2\,|p_i \odot u_i - q_i|}. \tag{34}$$

Adding the two sides of Eq.(33) and Eq.(34), we arrive at:

$$|p_i \odot \widetilde{u}_i - \widetilde{q}_i| \leq |p_i \odot u_i - q_i|.$$

Therefore, the following inequality also holds

$$\|p_i \odot \widetilde{u}_i - \widetilde{q}_i\|^2 \leq \|p_i \odot u_i - q_i\|^2.$$

This completes the proof of convergence of Algorithm 4.

□

Acknowledgement

This research is supported by Foreign Expert Program of China (Grant No.DL20230410021), Natural Science Basic Research Program of Shaanxi (Program No.2024JC-YBMS-015), Key Scientific Research Program (Information Security Special Project) of Shaanxi Provincial Education Department (Program No.21JX008) and Yulin City Industry University Research Project (No.CXY-2022-59).

References

[1] J. C. Bezdek, *A convergence theorem for the fuzzy ISODATA clustering algorithms*, IEEE Transactions on Pattern Analysis and Machine Intelligence, **2**(1) (1980), 1-8. https://doi.org/10.1109/TPAMI.1980.4766964

[2] J. C. Bezdek, *Pattern recognition with fuzzy objective function algorithms*, Springer Science and Business Media, 1981. https://doi.org/10.1007/978-1-4757-0450-1

[3] D. P. Branco, F. D. A. de Carvalho, *Medoid based semi-supervised fuzzy clustering algorithms for multi-view relational data*, Fuzzy Sets and Systems, **469**(15) (2023), 108630. https://doi.org/10.1016/j.fss.2023.108630

[4] X. Chang, F. Nie, Y. Yang, H. Huang, *A convex formulation for semi-supervised multi-label feature selection*, Proceedings of the AAAI Conference on Artificial Intelligence, **28**(1) (2014), 1171-1177. https://doi.org/10.5555/2893873.2894055

[5] R. Chen, Y. Tang, W. Zhang, W. Feng, *Deep multi-view semi-supervised clustering with sample pairwise constraints*, Neurocomputing, **500**(21) (2022), 832-845. https://doi.org/10.1016/j.neucom.2022.05.091

[6] C. Ding, *A new robust function that smoothly interpolates between ℓ_1 and ℓ_2 error functions*, Univerisity of Texas at Arlington Tech Report, (2013), 1621-1630.

[7] H. Dong, Y. Li, Z. Zhou, *Learning from semi-supervised weak-label data*, Proceedings of the AAAI Conference on Artificial Intelligence, **32**(1) (2018). https://doi.org/10.5555/3504035.3504392

[8] H. Gan, Z. Yang, R. Zhou, *Adaptive safety-aware semi-supervised clustering*, Expert Systems with Applications, **212** (2023), 118751. https://doi.org/10.1016/j.eswa.2022.118751

[9] G. González-Almagro, J. L. Suárez, P. Sánchez-Bermejo, J. R. Cano, S. García, *Semi-supervised clustering with two types of background knowledge: Fusing pairwise constraints and monotonicity constraints*, Information Fusion, **102** (2024). https://doi.org/10.1016/j.inffus.2023.102064

[10] Y. Guo, A. Sengur, *NCM: Neutrosophic c-means clustering algorithm*, Pattern Recognition, **48**(8) (2015), 2710-2724. https://doi.org/10.1016/j.patcog.2015.02.018

[11] A. Isazadeh, M. Ghorbani, *Fuzzy c-means and its generalization by ℓ_p-norm space*, WSEAS Transactions Mathematics, **2**(1) (2003), 168-170.

[12] Y. Jia, G. Lu, H. Liu, J. Hou, *Semi-supervised subspace clustering via tensor low-rank representation*, IEEE Transactions on Circuits and Systems for Video Technology, **33**(7) (2023), 3455-3461. https://doi.org/10.1109/TCSVT.2023.3234556

[13] W. Li, S. Wang, X. Guo, E. Zhu, *Deep graph clustering with multi-level subspace fusion*, Pattern Recognition, **134** (2023), 109077. https://doi.org/10.1016/j.patcog.2022.109077

[14] L. Li, J. Zhang, S. Wang, X. Liu, K. Li, K. Li, *Multi-view bipartite graph clustering with coupled noisy feature filter*, Indian Journal of Pure and Applied Mathematics, **35**(12) (2023), 12842-12854. https://doi.org/10.1109/TKDE.2023.3268215

[15] X. Li, Y. Zhang, R. Zhang, *Semi-supervised feature selection via generalized uncorrelated constraint and manifold embedding*, IEEE Transactions on Neural Networks and Learning Systems, **33**(9) (2022), 5070-5079. https://doi.org/10.1109/TNNLS.2021.3069038

[16] S. K. Majhi, S. Bhatachharya, R. Pradhan, S. Biswal, *Fuzzy clustering using salp swarm algorithm for automobile insurance fraud detection*, Indian Journal of Pure and Applied Mathematics, **36**(3) (2019), 2333-2344. https://doi.org/10.3233/JIFS-169944

[17] J. Mei, H. Lv, J. Cao, W. Gong, *Pairwise constrained fuzzy clustering: Relation, comparison and parallelization*, International Journal of Fuzzy Systems, **21**(6) (2019), 1938-1949. https://doi.org/10.1007/s40815-019-00683-1

[18] F. Nie, H. Huang, X. Cai, C. Ding, *Efficient and robust feature selection via joint $\ell_{2,1}$-norms minimization*, Advances in Neural Information Processing Systems, **23** (2010), 1813-1821. https://doi.org/10.5555/2997046.2997098

[19] F. Nie, H. Wang, H. Huang, C. Ding, *Unsupervised and semi-supervised learning via ℓ_1-norm graph*, International Conference on Computer Vision, **32**(11) (2011), 2268-2273. https://doi.org/10.1109/ICCV.2011.6126506

[20] F. Nie, H. Wang, H. Huang, C. Ding, *Adaptive loss minimization for semi-supervised elastic embedding*, Twenty-Third International Joint Conference on Artificial Intelligence, (2013), 1565-1571. https://doi.org/10.5555/2540128.2540353

[21] A. G. Oskouei, M. Hashemzadeh, B. Asheghi, M. A. Balafar, *CGFFCM: Cluster-weight and group-local feature-weight learning in fuzzy c-means clustering algorithm for color image segmentation*, Applied Soft Computing, **113** (2021), 108005. https://doi.org/10.1016/j.asoc.2021.108005

[22] A. G. Oskouei, N. Samadi, J. Tanha, *Feature-weight and cluster-weight learning in fuzzy c-means method for semi-supervised clustering*, Applied Soft Computing, **161** (2024), 111712. https://doi.org/10.1016/j.asoc.2024.111712

[23] T. Qi, X. Feng, B. Gao, K. Wang, *An end-to-end graph convolutional network for semi-supervised subspace clustering via label self-expressiveness*, Knowledge-Based Systems, **286** (2024), 111393. https://doi.org/10.1016/j.knosys.2024.111393

[24] S. Shalev-Shwartz, Y. Singer, *Efficient learning of label ranking by soft projections onto polyhedra*, The Journal of Machine Learning Research, **7** (2006), 1567-1599. https://doi.org/10.5555/1248547.1248605

[25] P. Smyth, *Model selection for probabilistic clustering using cross-validated likelihood*, Statistics and Computing, **10**(1) (2000), 63-72. https://doi.org/10.1023/A:1008940618127

[26] N. X. Thao, M. Ali, F. Smarandache, *An intuitionistic fuzzy clustering algorithm based on a new correlation coefficient with application in medical diagnosis*, Journal of Intelligent and Fuzzy Systems, **36**(1) (2019), 189-198. https://doi.org/10.3233/JIFS-181084

[27] T. M. Tuan, M. D. Sinh, T. I. Khang, T. T. Ngan, N. L. Giang, *A new approach for semi-supervised fuzzy clustering with multiple fuzzifiers*, International Journal of Fuzzy Systems, **24** (2022), 3688-3701. https://doi.org/10.1007/s40815-022-01363-3

[28] Y. Wan, Y. Zhong, A. Ma, *Fully automatic spectral-spatial fuzzy clustering using an adaptive multi-objective memetic algorithm for multi-spectral imagery*, IEEE Transactions on Geoscience and Remote Sensing, **57**(4) (2019), 2324-2340. https://doi.org/10.1109/TGRS.2018.2872875

[29] W. Wang, M. A. Carreira-Perpinán, *Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application*, arXiv preprint arXiv:1309.1541, (2013). https://doi.org/10.48550/arXiv.1309.1541

[30] Y. Wang, L. Chen, J. Zhou, T. Li, Y. Yu, *Pairwise constraints based semi-supervised fuzzy clustering with multi-manifold regularization*, Information Sciences, **638** (2023), 118994. https://doi.org/10.1016/j.ins.2023.118994

[31] Z. Wang, S. Wang, L. Bai, W. Wang, Y. Shao, *Semi-supervised fuzzy clustering with fuzzy pairwise constraints*, IEEE Transactions on Fuzzy Systems, **30**(9) (2022), 3797-3811. https://doi.org/10.1109/TFUZZ.2021.3129848

[32] H. Wang, J. Wang, G. Wang, *Clustering validity function fusion method of FCM clustering algorithm based on dempster-shafer evidence theory*, International Journal of Fuzzy Systems, **24**(1) (2022), 650-675. https://doi.org/10.1007/s40815-021-01170-2

[33] G. Wang, J. Wang, H. Wang, J. Liu, *Component-wise design method of fuzzy c-means clustering validity function based on CRITIC combination weighting*, The Journal of Supercomputing, **79**(13) (2023), 14571-14601. https://doi.org/10.1007/s11227-023-05234-y

[34] C. Wu, J. Zhang, *Robust semi-supervised spatial picture fuzzy clustering with local membership and KL-divergence for image segmentation*, International Journal of Machine Learning and Cybernetics, **13**(4) (2022), 963-987. https://doi.org/10.1007/s13042-021-01429-y

[35] C. Wu, J. Zhang, C. Huang, X. Guo, *Robust dynamic semi-supervised picture fuzzy clustering with KL divergence and local information*, Cognitive Computation, **14**(3) (2022), 970-988. https://doi.org/10.1007/s12559-021-09988-6

[36] L. Yuan, X. Yang, Z. Xing, Y. Ma, *Latent multi-view semi-nonnegative matrix factorization with block diagonal constraint*, Axioms, **11**(12) (2022), 722. https://doi.org/10.3390/axioms11120722

[37] L. Zadeh, *Fuzzy sets*, Information and Control, **8**(3) (1965), 338-353. https://doi.org/10.1016/S0019-9958(65)90241-X

[38] K. Zhang, P. Feng, G. Zhang, T. Xie, J. Hou, X. He, *The Bi-level optimal configuration model of the CCHP system based on the improved FCM clustering algorithm*, Processes, **9**(6) (2021), 907. https://doi.org/10.3390/pr9060907

[39] R. Zhang, X. Li, H. Zhang, F. Nie, *Deep fuzzy k-means with adaptive loss and entropy regularization*, IEEE Transactions on Fuzzy Systems, **28**(11) (2019), 2814-2824. https://doi.org/10.1109/TFUZZ.2019.2945232

[40] Y. Zhang, Y. Ma, X. Yang, *Multi-label feature selection based on logistic regression and manifold learning*, Applied Intelligence, **52** (2022), 9256-9273. https://doi.org/10.1007/s10489-021-03008-8

[41] X. Zheng, C. Tang, X. Liu, E. Zhu, *Multi-view clustering via matrix factorization assisted k-means*, Neurocomputing, **534** (2023), 45-54. https://doi.org/10.1016/j.neucom.2023.03.004

[42] J. Zhou, W. Pedrycz, X. Yue, C. Gao, Z. Lai, J. Wan, *Projected fuzzy c-means clustering with locality preservation*, Pattern Recognition, **113** (2021), 107748. https://doi.org/10.1016/j.patcog.2020.107748