Chaos, Solitons and Fractals 181 (2024) 114670

Contents lists available at ScienceDirect

Chaos, Solitons and Fractals

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

Neighborhood information based semi-supervised fuzzy C-means employing feature-weight and cluster-weight learning

Ali Kadhim Jasim, Jafar Tanha *, Mohammad Ali Balafar

Department of Computer Engineering, Faculty of Electrical and Computer Engineering, University of Tabriz, Iran

ARTICLE INFO

Keywords:
Semi-supervised clustering
Fuzzy C-means
Feature weighting

ABSTRACT

A semi-supervised fuzzy c-means algorithm uses auxiliary class distribution knowledge and fuzzy logic to handle semi-supervised clustering problems, named semi-supervised fuzzy c-means (SSFCM). Despite the performance enhancement obtained by adding additional information about data labels to the clustering process, semi-supervised fuzzy techniques still have several issues. All the data attributes in the feature space are assumed to have equal importance in the cluster formation, while some features may be more informative than others. Additionally, when the local information of samples is not considered through the cluster creation process, the SSFCM-based algorithms have low accuracy. In this research, a novel semi-supervised fuzzy c-means approach is proposed to handle the mentioned issues. The proposed approach is built on feature weighting, cluster weighting, and using neighborhood information. In this method, a novel fuzzy objective function based on feature weighting and cluster weighting is presented, which has two main parts: (1) a semi-supervised term that represents external class knowledge and (2) a spatial penalty term on the membership function that allows a clustering of sample to be affected by its neighbors. Both feature weights and cluster weights are determined adaptively through the clustering procedure. Combining both approaches forms an ideal clustering structure that is less sensitive to the initial centers. Furthermore, the penalty term acts as a regularizer and enhances the accuracy of the proposed approach. Experimental comparisons are conducted on several benchmark datasets to show the performance of the proposed approach. The experimental results further show that the proposed approach outperforms the state-of-the-art methods. The Matlab implementation source code of the proposed method is now publicly accessible at https://github.com/jafartanha/SSFCM-FWCW.

1. Introduction

Data clustering aims to divide the samples into dense groups to maximize the intra-cluster similarity and minimize the inter-cluster similarity [1,2]. Data clustering is crucial in various scientific applications, such as recommender systems, market segmentation, social network analysis, image processing, and biological data analysis [1,3,4].

One of the popular types of clustering approaches, known as centroid-based clustering, is where each cluster is represented by a center point vector [5]. Several algorithms have been developed where fuzzy c-means and k-means are the most popular [6]. Iteratively calculating cluster centers and allocating samples to the clusters is the standard process used in these approaches.

Approaches are categorized into two broad groups [7,8] based on the degree of restriction for allocating the samples to the clusters which are: a) hard clustering and b) soft/fuzzy clustering. Each sample can only be a member of one cluster when using the hard clustering approach. The soft clustering approach, in contrast, considers a membership degree for each sample with respect to each cluster in the range [0,1], which represents the degree of partial membership of the sample in that cluster. A fuzzy point-of-view in cluster assignment yields a superior clustering structure when the data distribution calls for overlapping cluster structures or when the boundary between the clusters is ambiguous [9,10].

Although fuzzy clustering methods are preferable to hard clustering, they still act unsupervised. However, In some real-world applications, experts supply certain types of prior knowledge about the class distribution of the data [2,11]. As a result, semi-supervised clustering, which uses prior knowledge about the labels of samples to improve the unsupervised clustering performance [12], is an interesting approach.

The semi-supervised algorithms are divided into three groups [13,14] based on how this additional information is incorporated into

* Corresponding author at: 29th Bahman Blvd, University of Tabriz, Tabriz 5166616471, Iran.
E-mail addresses: ali.kadhim@tabrizu.ac.ir (A.K. Jasim), tanha@tabrizu.ac.ir (J. Tanha), balafarila@tabrizu.ac.ir (M.A. Balafar).

https://doi.org/10.1016/j.chaos.2024.114670
Received 18 September 2023; Received in revised form 5 December 2023; Accepted 25 February 2024
Available online 8 March 2024

the clustering process: a) distance-based method, b) constraint-based approach, and c) hybrid algorithms. The distance-based category formulates distance/similarity metrics to satisfy the class label constraints, e.g., see [15–17]. The constraint-based approaches employ the available pairwise constraints to direct the clustering process, typically by adding new terms to the objective function, such as in [18,19]. The third category considers a mix of constraint- and distance-based approaches, such as those proposed in [20,21].

Recently, several fuzzy approaches, applied in the semi-supervised applications, were proposed, such as S3FCM [22], CS3FCM [23], LHC-S3FCM [24], and AS3FCM [25], TS3FCM [26], TS3MFCM [27]. Most of these proposed methods are motivated by the SSFCM algorithm, a cutting-edge semi-supervised fuzzy clustering method [19]. The SSFCM presents a reformulated fuzzy objective function embedded with the external class label information and considers the difference between the predicted class and the actual class of the instances as a penalty term [28]. Although these approaches perform better than unsupervised approaches, they still have fundamental shortcomings that prohibit them from identifying the best clusters. These methods assume that all features are equally important in generating the clusters, even if some features might be more informative. Hence, the most optimal cluster structure may not emerge due to this assumption. Although unsupervised clustering has somewhat overcome this problem [29,30], there is still insufficient study on the use of feature weighting strategies in the semi-supervised clustering algorithms.

In addition to the aforementioned issues, many semi-supervised techniques currently in use are based on the conventional FCM algorithm, which is extremely sensitive to noise, and initial centers [31–33], because it employs the Euclidean distance metric. The performance of these solutions can be considerably impacted by noise and outliers in datasets.

This research proposes a novel semi-supervised fuzzy C-means clustering approach to address the abovementioned issues. Feature-weight and Cluster-weight learning are used to build a novel objective function motivated by the SSFCM [19] and FWCW-FCM [29]. A novel objective function is designed based on Feature-weight and Cluster-weight learning. Incorporating the class label information into the objective function improves the algorithm's understanding of the initial number of clusters, better approximation of initial center vectors, and assigning the labeled samples to the correct cluster representing the same label. The effect of significant features in creating the best clusters is bolded when a feature weighting schema is included in the objective function.

Additionally, taking into account various cluster weights results in initial center insensitivity. Moreover, the performance consequences of noisy and outlier samples are avoided by using a non-Euclidean distance measure. Finally, the following are the critical contributions of the proposed approach:

- Using fuzzy logic and prior knowledge of class distribution, a novel semi-supervised fuzzy clustering objective function is developed.
- During the clustering iterations, a feature weighting technique is used, applying an adaptive weight for each feature based on its significance in forming the clusters.
- In addition to feature weighting, a weighting factor is also considered for the clusters, which is set adaptively for each cluster, resulting in insensitivity to initial center selection.
- A spatial penalty term on the membership function is used to allow a sample's clustering to be influenced by its neighbors, improving clustering accuracy.

In terms of clustering accuracy, the performance of the proposed method is evaluated against several cutting-edge methods using a variety of benchmark datasets. The results of the experiments show that the proposed method is superior to the different state-of-the-art approaches. Furthermore, thorough evaluations are carried out to show the effect of cluster weighting, feature weighting, using neighbors, and the labeled data ratio on the performance.

The rest of the paper is structured as follows: The related work is briefly discussed in Section 2, the preliminary methods and the contribution of this research are described in Sections 3 and 4, respectively, and the specifics of the experimental assessment and the results are presented in Section 5. Finally, Section 6 provides the conclusion and the future work.

2. Related work

In this section, the state-of-the-art Semi-Supervised Fuzzy Clustering approaches are reviewed. The concept of using the external class label information, which is present for a limited number of instances, to guide the clustering process is presented by semi-supervised clustering [34]. In constraint-based semi-supervised approaches, the external label information can influence the clustering in two different ways [35]: a) Using the labeled instances only for initializing the cluster centers and b) Injecting the class label information into the expectation-maximization steps of the clustering, for example, by creating a semi-supervised objective function [24]. As a fundamental method for semi-supervised fuzzy clustering, SSFCM [19] reformulates the FCM object function by including a semi-supervised term that considers the degree to which the obtained membership label matches the actual label.

A semi-supervised FCM implementation that aims to adjust the number of clusters to be adaptively equal to or greater than the number of class labels is proposed by Pedrycz and Vukovich [36]. Bouchachia and Pezdrycz [37] modified the SSFCM loss function to address situations with fewer classes than clusters.

Active Fuzzy Constrained Clustering (AFCC) [38] uses the violation costs of pairwise must-link and cannot-link constraints to the loss function to determine the proper number of clusters. Two semi-supervised fuzzy clustering methods—sSFCM and eSFCM—are proposed by Yasunori et al. [39]. A supervised membership degree is added as a new parameter to the FCM in the former. In contrast, in the latter, the supervised parameter is used to reformulate the EFCM objective function.

An adaptation of the AFCC algorithm, SCAPC [40], is a semi-supervised fuzzy clustering technique with paired constraints. The objective function of AFCC has a new cost component added by SPASC, which considers the sum of the weighted distances of the must-link and cannot-link constraints as a penalty.

A weighted semi-supervised version of FCM that embeds prior label information as a membership degree constraint is proposed by SSFCM-HPR [35]. Additionally, samples are separated according to their weights, which show how representative they are within each cluster according to the distance criterion. SSFCM-HPR objective function parameters are optimized using the Hestense–Powell multiplier method.

By including a semi-supervised term (augmented regularization function) into the unsupervised EFCM objective function, Salehi et al. developed the SFC-ER [41] algorithm. In the semi-supervised term, the relative entropy divergence metric is used instead of the geometric distance metric.

A label consistency term objective function has recently been embedded in numerous adaptations for the SSFM technique. LHC-S3FCM [24], which uses local homogenous consistency as a criterion, was introduced in 2018. Each labeled sample's clustering predictions converge with those of its homogeneous immediate neighbors due to the recently added term. Further, the Gaussian kernel similarity measure defines the neighborhood relationship between the samples.

S3FCM [22] is proposed with partial distrust of the external class label knowledge. At each iteration, S3FCM predictions are compared with FCM's unsupervised predictions, available as ground truth data. Any discrepancy in the predictions of samples is considered a penalty. By giving the labeled instances a weight factor that reflects their risk/confidence level, the confidence-weighted S3FCM, known as CS3FCM

[23], improves the LHC-S3FCM's efficiency. The adopted weighting mechanism is based on the degree of agreement between the FCM membership values and the label predictions. AS3FCM [25] improves CS3FCM, which calculates the safety level of labeled instances adaptively along the clustering process instead of using FCM hints.

TS3FCM [26] introduced a Trusted S3SFCM strategy that redesigned a new FCM goal function based on the labeled samples' local neighborhood to save computation time and handle a noisy data point. Based on this simple phase, the initial centers and labeled sample ground truth information are gathered for the clustering process. Consequently, the unlabeled instances' ground truth is determined using just one FCM iteration. A new iteration of TS3FCM called TS3MFCM is proposed by Tuan et al. [27] and is based on the multiple fuzzifiers approach. The semi-supervised objective function is reformed by considering a distinct fuzzifier for each sample chosen based on the sample and its local neighborhood. On the other hand, MCSSFC-C [42] is proposed as an improvement on TS3FCM that considers a separate fuzzifier for each cluster instead of samples. The number of uncertain samples in a cluster represents the uncertainty of that cluster because the fuzzifier value of each cluster is calculated based on the FCM membership values.

3. Preliminaries

The foundational ideas of this work are briefly explored in this part. The SSFCM and FWCW-FCM [29] baseline techniques are discussed in Sections 3.1 and 3.2 respectively.

3.1. SSFCM

As mentioned earlier, SSFCM improves the clustering process by including the prior knowledge of the class labels in the FCM [19]. In SSFCM, the data is considered as a union of N_l labeled samples, and $N_u = N - N_l$ unlabeled samples, where $N_l \ll N_u$, and N represents the number of samples. Adopting the same principles as FCM, a membership degree matrix $U = [u_{nk}]_{K\times N}$ is considered, which represents the membership degree of the nth sample to the kth cluster and obeys the same membership constraints as FCM ($u_{nk} \in [0,1], \sum_{k=1}^{K} u_{nk} = 1$). Accordingly, the objective function of the SSFCM algorithm is defined as:

$$J_{SSFCM} = \sum_{n=1}^{N}\sum_{k=1}^{K} {u_{nk}}^2 d_{nk}^2 + \alpha \sum_{n=1}^{N}\sum_{k=1}^{K} (u_{nk} - b_n f_{nk})^2 d_{nk}^2$$
$$u_{nk} \in [0,1], \sum\nolimits_{k=1}^{K} u_{nk} = 1, \quad \text{where} \quad 1 \le n \le N \quad \text{and } 1 \le k \le K \tag{1}$$

In Eq. (1), the second term represents the prior supervised knowledge, in which, $\mathbf{B} = [\mathrm{b_n}]$ is an N-dimensional label indicator vector with $b_n = 1$ for labeled samples, and $b_n = 0$ for unlabeled samples. Furthermore, $\mathbf{F} = [f_{nk}]$ corresponds to the fuzzy degrees of the labeled samples in which $f_{nk} = 1$ if sample n belongs to the class k, and 0 otherwise. Finally, $\alpha \ge 0$ is the regularization parameter.

Adopting the Lagrange multipliers approach, the membership degree of sample n in cluster k (u_{nk}) is computed as follows.

$$u_{nk} = \frac{1}{1+\alpha}\left[\frac{1+\alpha\left(1 - b_n \sum_{h=1}^{K} f_{nh}\right)}{\sum_{h=1}^{K} d_{nk}^2 \Big/ d_{nh}^2}\right] + \alpha b_n f_{nk}, \forall n = 1,\dots,N \ and \forall k = 1,\dots,K \tag{2}$$

For cluster k, the cluster center c_k is computed according to Eq. (3).

$$c_k = \frac{\sum_{n=1}^{N} u_{nk}^2 x_n + \alpha \sum_{n=1}^{N} (u_{nk} - b_n f_{nk})^2 x_n}{\sum_{n=1}^{N} u_{nk}^2 + \alpha \sum_{n=1}^{N} (u_{nk} - b_n f_{nk})^2}, \forall k = 1,\dots,K \tag{3}$$

3.2. FWCW-FCM

FWCW-FCM [29] is provided as an enhanced type of FCM that distinguishes between distinct data features in each cluster. A mechanism for adaptive cluster-weighting is adopted to address the initial center sensitivity issue of FCM. Hence, Eq. (4) is rewritten as the objective function of FCM as follows:

$$J_{FWCW-FCM} = \sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{m=1}^{M} {u_{nk}}^{\sigma} w_{km}^{q} z_k^{p} d^2(x_{nm} - c_{km}) \tag{4}$$

For this objective function, constraints are presented in Eq. (5):

$$\begin{aligned} &u_{nk} \in [0,1], \sum\nolimits_{k=1}^{K} u_{nk} = 1, \text{where } \ 1 \le n \le N \text{ and } \ 1 \le k \le K; \\ &w_{km} \in [0,1], \sum\nolimits_{m=1}^{M} w_{km} = 1 \text{ where } \ 1 \le k \le K \text{ and } \ 1 \le m \le M; \\ &z_k \in [0,1], \sum\nolimits_{k=1}^{K} z_k = 1 \text{ where } \ 1 \le k \le K \text{ and } \ 0 \le p < 1. \end{aligned} \tag{5}$$

In Eq. (4), N, M, and K indicate the sample count, feature space dimension, and cluster count respectively. $\mathbf{U} = [u_{nk}]_{K\times N}$ is a membership matrix in which entry u_{nk} corresponds to the membership degree of the nth sample to the kth cluster center. In the $\boldsymbol{C} = [c_{km}]_{K\times M}$ matrix, c_{km} refers to the mth feature in the kth cluster. Moreover, in the $\boldsymbol{W} = [w_{km}]_{K\times M}$ matrix, w_{km} indicates the weight of the mth feature in the kth cluster and in $z = [z_k]_{1\times K}$ vector, entry $\mathrm{z_k}$ indicates the kth cluster weight. Furthermore, the term $d^2(x_{nm} - c_{km})$ indicates a non-Euclidean distance metric.

Based on the closed-form solution obtained for Eq. (4), formulas for u_{nk}, c_{km}, w_{km}, and z_k are presented in eqs. 6, 7, 8, and 9 respectively.

$$u_{nk} = \frac{1}{\sum_{l=1}^{K}\left[\frac{z_k^p \sum_{m=1}^{M} w_{km}^q \ d^2(x_{nm}-c_{km})}{z_l^p \sum_{m=1}^{M} w_{lm}^q \ d^2(x_{nm}-c_{lm})}\right]^{\frac{1}{(\sigma-1)}}} \tag{6}$$

$$c_{km} = \frac{\sum_{n=1}^{N} {u_{nk}}^{\sigma}\left(exp\left(-\gamma_m (x_{nm} - c_{km})^2\right)\right)(x_{nm})}{\sum_{n=1}^{N} {u_{nk}}^{\sigma}\left(exp\left(-\gamma_m (x_{nm} - c_{km})^2\right)\right)} \tag{7}$$

$$w_{km} = \frac{1}{\sum_{s=1}^{M}\left[\frac{\sum_{n=1}^{N} {u_{nk}}^{\sigma} \ d^2(x_{nm}-c_{km})}{\sum_{n=1}^{N} {u_{nk}}^{\sigma} \ d^2(x_{ns}-c_{ks})}\right]^{\frac{1}{q-1}}} \tag{8}$$

$$z_k = \frac{1}{\sum_{l=1}^{K}\left[\frac{\sum_{n=1}^{N}\sum_{m=1}^{M} {u_{nk}}^{\sigma} w_{km}^q \ d^2(x_{nm}-c_{km})}{\sum_{n=1}^{N}\sum_{m=1}^{M} {u_{nl}}^{\sigma} \ w_{lm}^q \ d^2(x_{nm}-c_{lm})}\right]^{\frac{1}{p-1}}} \tag{9}$$

4. Proposed semi-supervised FCM approach

We use cluster weighting and feature weighting factors in the proposed approach to improve the efficiency of the SSFCM algorithm in light of the challenges outlined for the SSFCM method. With the local weighting of the features, the most important features are given more weights. On the other hand, weighting the clusters results in balanced clusters through the clustering procedure. Fig. 1 presents the general overview of the proposed framework.

Furthermore, a spatial penalty term on the membership function is applied to allow the clustering of a sample to be influenced by its neighbors. Taking the influence of the local neighbors of a sample during clustering helps the proposed algorithm to form better clusters and improves efficiency.

4.1. Feature weighting principle

The principle that should be considered for weighting features in each cluster is as follows: assign a larger weight to a feature that has a

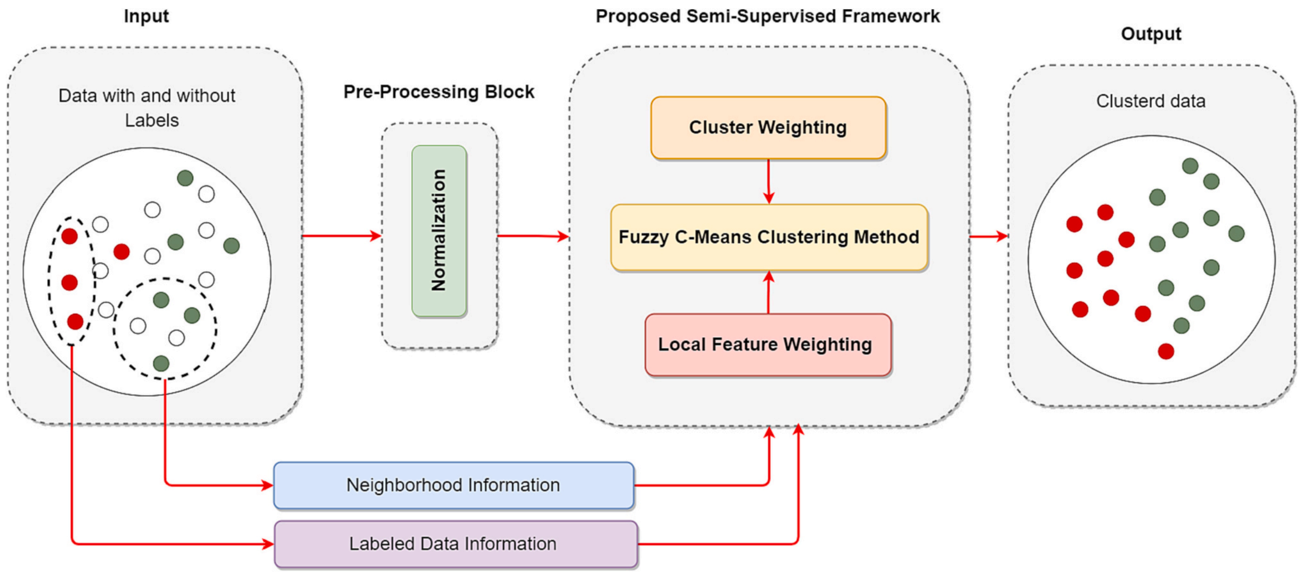

Fig. 1. The general overview of the proposed algorithm.

smaller variance in the related cluster, and a smaller weight to a feature that has a larger variance. In this paper, we refer to this principle as the "*feature weighting principle*".

4.2. Cluster weighting principle

k-partitioning algorithms are sensitive to initialization. This sensitivity exists in both algorithms with and without feature weighting. Feature weighting is necessary but not enough. After a bad initialization, it is possible that some clusters with large SIWD[1]s are merged together, and those with low SIWD are broken into smaller clusters. Therefore, even if there are some clusters with balanced SIWDs in the dataset, after running the algorithm, it is possible that some clusters with unbalanced SIWDs are formed. This is the case that happens in k-means based algorithms.

Cluster weighting solves this problem to a large extent. The principle that is used for weighting clusters is as follows: assign a larger weight to a cluster that has a larger SIWD, and a smaller weight to a cluster that has a smaller SIWD. We call this principle the "*cluster weighting principle*". By assigning a higher weight to a cluster with a big SIWD, we can prevent the formation of clusters with an unbalanced SIWD.

We then formulate the objective function as follows:

$$F(\mathbf{U},\mathbf{C},\mathbf{W},\mathbf{z}) = \sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{m=1}^{M} u_{nk}{}^{2} w_{km}^{q} z_{k}^{p} d^{2}(x_{nm}-c_{km}) + \frac{\alpha_1}{N_R}\sum_{k=1}^{K} \times \sum_{n=1}^{N} u_{nk}{}^{2} \sum\nolimits_{r\in N_n} (1-u_{rk})^2 + \alpha_2 \sum_{n=1}^{N}\sum_{k=1}^{K} \times \sum_{m=1}^{M} (u_{nk}-b_n f_{nk})^2\, w_{km}^{q} z_{k}^{p} d^{2}(x_{nm}-c_{km}) \tag{10}$$

In the first term of the objective function, *N*, *M*, and *K* indicate the data sample count, feature space dimension, and cluster count respectively. $\mathbf{U} = [u_{nk}]_{K\times N}$ is a membership matrix in which entry u_{nk} corresponds to the membership degree of the *n*th sample to the *kth* cluster center. In the $\boldsymbol{C} = [c_{km}]_{K\times M}$ matrix, c_{km} refers to the *mth* feature in the *kth* cluster. Moreover, in the $\boldsymbol{W} = [w_{km}]_{K\times M}$ matrix, w_{km} indicates the weight of the *mth* feature in the *kth* cluster and in $z = [z_k]_{1\times K}$ vector, entry z_k addresses the *kth* cluster weight. The parameter *q* is in the range $q < 0$ and $q > 1$, and the parameter *p* is within the range $0 \le p < 1$. The values of the p and *q* parameters are determined by the user. Additionally, the term $d^2(x_{nm} - c_{km})$ shows a non-Euclidean distance metric and is defined as follows:

$$1 - exp\left(-\gamma_m (x_{nm} - c_{km})^2\right) \tag{11}$$

where γ_m indicates the inverse of the variance of the *mth* feature of *x* dataset:

$$\gamma_m = \frac{I}{var_m}, var_m = \sum_{n=1}^{N} \frac{(x_{nm} - \bar{x}_m)^2}{N}, \bar{x}_m = \sum\nolimits_{n=1}^{N} \frac{x_{nm}}{N}, I = (0,1] \tag{12}$$

In the second term of the objective function, $\alpha_1 \ge 0$ is a regularization parameter that exploits local neighborhood information, N_n represents the set of neighbors for *n*th sample, and N_R represents the number of neighbors. In Subsection 4.1, the neighbors of a sample are explained in more detail.

In the third term of the objective function, $\alpha_2 \ge 0$ isa regularization parameter for the objective function's third part that exploits class information. $\mathbf{B} = [\mathrm{b_n}]$ is a label indicator vector with the length of *N*, where $b_n = 1$ if x_{nm} is labeled and $b_n = 0$, otherwise. Furthermore, $\mathbf{F} = [f_{nk}]$ denotes the fuzzy degrees of the labeled samples where $f_{nk} = 1$ if sample *n* belongs to class *k* and $f_{nk} = 0$ otherwise.

For Eq. (10), the following constraints should be addressed:

$$\begin{aligned} &u_{nk} \in [0,1], \quad \sum\nolimits_{k=1}^{K} u_{nk} = 1 \text{ where } \quad 1 \le n \le N \text{ and } 1 \le k \le K; \\ &w_{km} \in [0,1], \quad \sum\nolimits_{m=1}^{M} w_{km} = 1 \text{ where } \quad 1 \le k \le K \text{ and } 1 \le m \le M; \\ &z_k \in [0,1], \quad \sum\nolimits_{k=1}^{K} z_k = 1 \text{ where } \quad 1 \le k \le K \text{ and } \quad 0 \le p < 1. \end{aligned} \tag{13}$$

Based on the closed-form solution obtained for Eq. (10), formulas for u_{nk}, c_{km}, w_{km}, and z_k are presented in Eqs. (14, 15, 16, and 17), respectively (see Appendix for more details about updating formulas):

$$u_{nk} = \frac{1 + \sum_{l=1}^{K} \frac{\alpha_2 b_n f_{nk}\ d_{nk}}{(1+\alpha_2)d_{nl} + \frac{\alpha_1}{N_R}\sum_{r\in N_n}(1-u_{rl})^2} - \sum_{l=1}^{K} \frac{\alpha_2 b_n f_{nl} d_{nl}}{(1+\alpha_2)d_{nl} + \frac{\alpha_1}{N_R}\sum_{r\in N_n}(1-u_{rl})^2}}{\sum_{l=1}^{K} \frac{(1+\alpha_2)d_{nk} + \frac{\alpha_1}{N_R}\sum_{r\in N_n}(1-u_{rk})^2}{(1+\alpha_2)d_{nl} + \frac{\alpha_1}{N_R}\sum_{r\in N_n}(1-u_{rl})^2}} \tag{14}$$

[1] the sum of their members' intra-cluster distances and member's current features'

$$c_{km} = \frac{\sum_{n=1}^{N}(x_{nm})\left(exp\left(-\gamma_m(x_{nm}-c_{km})^2\right)\right)\left(u_{nk}{}^2+\alpha_2(u_{nk}-b_nf_{nk})^2\right)}{\sum_{n=1}^{N}\left(exp\left(-\gamma_m(x_{nm}-c_{km})^2\right)\right)\left(u_{nk}{}^2+\alpha_2(u_{nk}-b_nf_{nk})^2\right)} \tag{15}$$

When weighing the features in each cluster, the following assumption should be made: give a higher weight to a feature in the associated cluster with a lower variation and a lower weight to a feature with a higher variance. According to this assumption, variance refers to the total of intra-cluster distances rather than its usual and standard interpretation. In Eq. (16), $\sum_{n=1}^{N}\left[u_{nk}{}^2+\alpha(u_{nk}-b_nf_{nk})^2\right] d^2(x_{nm}-c_{km})$ is the variance (sum of intra-cluster distances) of the *m-th* feature in the k-*th* cluster.

$$w_{km} = \frac{1}{\sum_{s=1}^{M}\left[\frac{\sum_{n=1}^{N}\left[u_{nk}{}^2+\alpha_2(u_{nk}-b_nf_{nk})^2\right] d^2(x_{nm}-c_{km})}{\sum_{n=1}^{N}\left[u_{nk}{}^2+\alpha_2(u_{nk}-b_nf_{nk})^2\right] d^2(x_{ns}-c_{ks})}\right]^{\frac{1}{q-1}}} \tag{16}$$

When weighing the clusters, the following assumption should be followed: give a cluster with a bigger SIWD (the sum of the intra-cluster weighted distances) a larger weight, and a cluster with a smaller SIWD a smaller weight. In Eq. (17), $\sum_{n=1}^{N}\sum_{k=1}^{K} w_{km}^{q}\, d^2(x_{nm}-c_{km})\left[u_{nk}{}^2+\alpha_2(u_{nk}-b_nf_{nk})^2\right]$ shows the SIWD (the sum of the intra-cluster weighted distances) for the k-*th* cluster.

$$z_k = \frac{1}{\sum_{l=1}^{K}\left[\frac{\sum_{n=1}^{N}\sum_{k=1}^{K} w_{km}^{q}\, d^2(x_{nm}-c_{km})\left[u_{nk}{}^2+\alpha_2(u_{nk}-b_nf_{nk})^2\right]}{\sum_{n=1}^{N}\sum_{k=1}^{K} w_{lm}^{q}\, d^2(x_{nm}-c_{lm})\left[u_{nl}{}^2+\alpha_2(u_{nl}-b_nf_{nl})^2\right]}\right]^{\frac{1}{p-1}}} \tag{17}$$

The purpose of weighting the clusters is summarized as follows:

1) Preventing the formation of unbalanced clusters during the SIWD-defined clustering process
2) Considering the first objective, cluster weighting is generated automatically during the clustering process, resulting in high-quality clusters independent of the initial centers.

4.3. Candidate neighbors

As stated in the previous sections, a new term, spatial penalty, has been added to the objective function. This section discusses this term and addresses how to find the neighbors.

The parameter α_1 controls the effect of the penalty term. The new penalty term is minimized when the membership value for a specific sample class is significant, and the membership values for that cluster are similarly large in surrounding samples and vice versa. In other words, the membership value of a sample is connected to the membership values of the other samples in its neighborhood.

It is interesting to note that we emphasize more the effect of the neighbors on the memberships of a sample, which some of the samples have actual labels. Labeled data make a significant contribution to better clustering of other unlabeled samples. The existence of some labeled data among unlabeled samples makes the unlabeled samples correctly predicted with high accuracy.

To find the neighbors of a sample, we choose N_R nearest neighbors of its samples. The used metric to find the neighbors of a sample is addressed in Eq. (18).

$$N_n = \min_{1\ to\ N_R}\left\{1-exp\left(-\|x_n-x_{\acute{n}}\|^2\right)\right\} \tag{18}$$

where $\acute{n}\epsilon[1,N]$, N_n represents the set of neighbors for nth sample, and N_R indicates the number of neighbors.

The critical point here is that the added term in the objective function increases the accuracy of the proposed method. On the other hand, kernel distance is used to find the neighborhood of samples. Kernel distance itself is more robust to noise than standard Euclidean distance.

The advantage of the neighboring technique is summarized as follows: the penalty term acts as a regularizer, increasing the accuracy of clustering and better-forming clusters.

4.4. Pseudo-code of the proposed algorithm

Fig. 2 depicts the pseudo-code of the proposed algorithm. To discover the suitable value of p, we use an iteration-based strategy with three parameters: p_{max}, p_{step}, and p_{init}, similar to the method provided in [29]. We begin the clustering procedure by setting p to p_{init}. In each iteration, we increase p by p_{step} until we reach the p_{max}. If an empty cluster or a cluster with only one sample appears, we reduce p by p_{step}. We then select the u_{nk}, w_{km}, and z_{km} that correspond to the p in the previous iteration. The procedure is repeated until the difference between the objective function values is smaller than the threshold value (ε) in two consecutive iterations, or the number of iterations reaches to t_{max}.

We modify the weights to include a memory effect parameter to strengthen the stability of the proposed method. Utilizing this option results in values transitioning between iterations more smoothly. In other words, the impact of the weights from the previous update on the present update is minimal. The parameter β controls this transition, see Eqs. (19) and (20).

$$\boldsymbol{W}^{(t)} = \beta \boldsymbol{W}^{(t-1)} + (1-\beta)\,\boldsymbol{W}^{(t)} \tag{19}$$

$$z^{(t)} = \beta z^{(t-1)} + (1-\beta)\,z^{(t)} \tag{20}$$

where the parameter β is in the range [0,1).

5. Experiments

In this section, the effectiveness of the proposed approach is evaluated based on several benchmark datasets. The results are compared to the baseline and most recent algorithms stated below:

- SSFCM [19]
- FCM [43]
- Seeded- k-means [18]
- SFC-ER [41]
- S3FCM [22]
- CS3FCM [23]
- AS3FCM [25]
- LHC-S3FCM [24]
- TS3FCM [26]
- FWCW-FCM [29]

Parameters ε and t_{max} in the implemented methods are set to 10^{-5} and 100, respectively. In the proposed method, the secondary parameters are set as follows: $p_{max} = 0.5, p_{init} = 0, p_{step} = 0.01, \alpha_1 = 1$ and $\alpha_2 = 1$. Like the method proposed in [29], the parameter β is selected for each dataset from $\{0, 0.1, 0.3\}$, and the parameter q is chosen for each dataset from $\{-10, -8, -6, -4, 2, 4, 6, 10\}$.

5.1. The used datasets

We employ 12 benchmark datasets to thoroughly investigate the consequences of the strategies described in our method and compare them to other approaches, see Table 1. Like similar methods in [22–25], we randomly choose 20 % of each dataset as the labeled subset and the remaining 80 % as the unlabeled subset.

Algorithm 1. Pseudo-code of the proposed method.

Input: Dataset $\boldsymbol{\chi} = \{x_n\}_{n=1}^{N}$, Number of cluster K, Parameters ε, p_{init}, p_{step}, p_{step}, t_{max}, q, β, α_1 and α_2.
Output: Membership matrix **U;**

set $t = 0$
set $p_{init} = 0$
set $z_k^{(0)} = \frac{1}{K}$, $\forall k = 1 \dots K$
set $w_{km}^{(0)} = \frac{1}{M}$, $\forall k = 1 \dots K$
set empty= **FALSE** //No empty or singleton clusters yet detected
$p = p_{init}$
find the neighbors of a sample by Eq. (18)
repeat
$t = t + 1$
Update the cluster assignments **U** by Eq. (14)
If empty or singleton clusters have occurred at a time t **then** //reduce p.
  empty=**TRUE**
  $p = p - p_{step}$
  if $p < p_{init}$ **then**
    return NULL
  end if
  //Revert to the assignments and weights corresponding to the reduce p.
  $u_{nk}^{(t)} = \left[\mathbf{U\_history}^{(p)}\right]_{nk}$, $\forall k = 1 \dots K$, $\forall n = 1 \dots N$
  $z_k^{(t-1)} = \left[\boldsymbol{Z\_history}^{(p)}\right]_k$, $\forall k = 1 \dots K$
  $w_{km}^{(t-1)} = \left[\boldsymbol{w\_history}^{(p)}\right]_{km}$, $\forall m = 1 \dots M$, $\forall k = 1 \dots K$
end if
Update the cluster center **C** by Eq. (15)
if $p < p_{max}$ **and** empty=**FALSE** then //increase p.
  $\mathbf{U\_history}^{(p)} = [u_{nk}^{(t)}]$ //store the current **U** in $\mathbf{U\_history}^{(\boldsymbol{p})}$.
  $\boldsymbol{W\_history}^{(p)} = [w_{km}^{(t-1)}]$ //store the previous $\boldsymbol{W}$ in $\boldsymbol{W\_history}^{(\boldsymbol{p})}$.
  $\boldsymbol{Z\_history}^{(p)} = [z_k^{(t-1)}]$ //store the previous $\boldsymbol{Z}$ in $\boldsymbol{Z\_history}^{(\boldsymbol{p})}$.
  $p = p + p_{step}$
end if
Update the feature weight **W** by Eq. (19)
Update the cluster weight **Z** by Eq. (20)
until $\left|F^{(t)} - F^{(t-1)}\right| < \varepsilon$ ***or*** $t \geq t_{max}$
return U

Fig. 2. Pseudo-code of the proposed method.

5.2. Evaluation criteria

Like the similar methods presented in [22–25], in the experiments, we use clustering accuracy (ACC) to show how well various approaches perform. The predicted label for x_n and its ground truth is denoted by $\widetilde{y}_k$ and y_n, respectively. The clustering accuracy is then calculated as follows:

$$ACC = \frac{\sum_{n=1}^{N} \delta\left(y_n, map\left(\widetilde{y}_k\right)\right)}{N} \quad (20)$$

where the delta function $\delta(x, y)$ has an output of 1 if $x = y$ and 0 otherwise. The Kuhn-Munkres algorithm's permutation mapping function maps the label $\widetilde{y}_k$ to its equivalent.

5.3. Experiment 1: Effect of feature weighting

In this experiment, we evaluate the effect of the feature weighting on the clustering of real-world datasets by comparing the performance of the proposed method twice—once with and once without the feature weighting. Fig. 3 displays the average outcomes of 10 iterations of the algorithms on each dataset.

Fig. 3 demonstrates how the feature weighting strategy improves the performance of the proposed algorithm. As shown in the figure, applying a weighting scheme improves the accuracy rate of the proposed method by an average of 6.36 % compared to the model without the feature weighting. This is further demonstrated by the relative importance of various features across several datasets, including *Bupa*, *Ecoli*, *Iris*, and *Statlog heart*.

For some datasets, such as *Australian, Dermatology, Pima,* and *Waveform*, the results are not significantly improved using the feature weighting. This is because each feature is separable in these datasets,

Table 1
Characteristics of the real-world dataset.

Dataset	Number of data points	Number of dimensions	Number of classes
Iris	150	4	3
Wine	178	13	3
Statlog (heart)	270	13	2
Ecoli	336	8	8
Bupa (liver disorders dataset)	345	7	2
Ionosphere	351	34	2
Dermatology	366	33	6
Breast Cancer Wisconsin (diagnostic)	569	32	2
Balance	625	4	3
Australian credit approval (statlog)	690	14	2
Pima Indians Diabetes	768	8	2
Waveform	5000	21	3

and weighting did not improve the results. It is essential that in the proposed method, in addition to weighting features, other techniques, such as neighborhood and cluster weighting have been used. In Subsections 5.4, 5.6, and 5.7, we will find that other techniques have a more positive effect on these datasets rather than weighting the features.

5.4. Experiment 2: Effect of cluster weighting

In this experiment, we evaluate the effect of cluster weighting on the final clustering of real-world datasets by comparing the performance of the proposed algorithm twice—once with and once without cluster weighting. Fig. 4 displays the average results of 10 iterations of the algorithms on each dataset.

Fig. 4 demonstrates how the cluster weighting method improves the performance of our algorithm. As shown in this figure, applying a weighting scheme improves the accuracy rate of the proposed method on all testing datasets by an average of 4.31 % compared to the mode without cluster weighting. This is highlighted by the fact that distinct

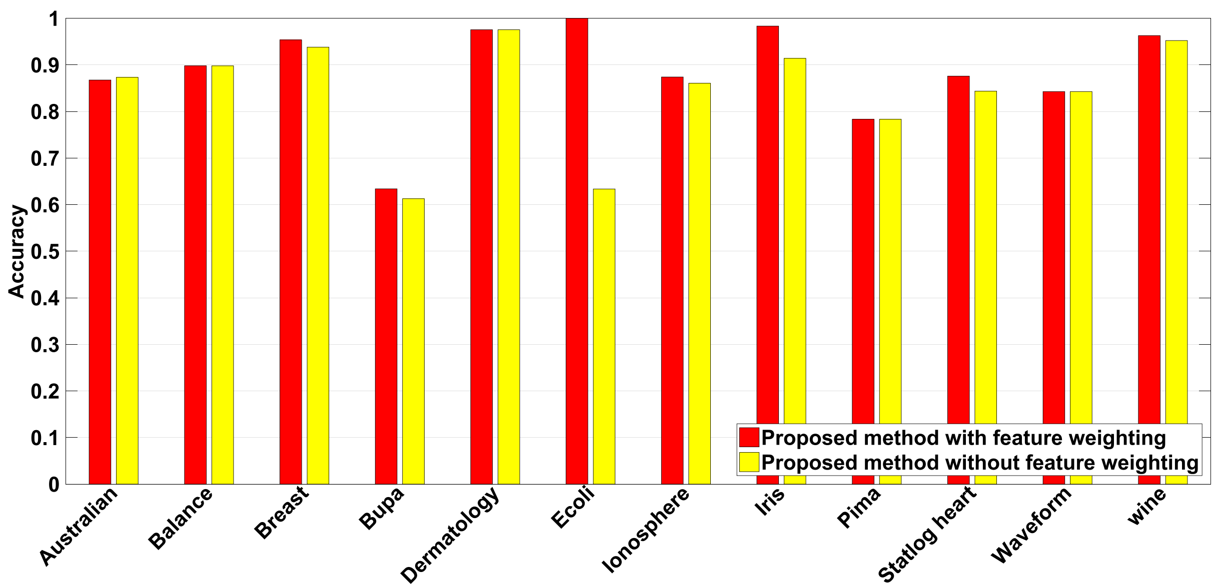

Fig. 3. The impact of feature weighting on the clustering quality.

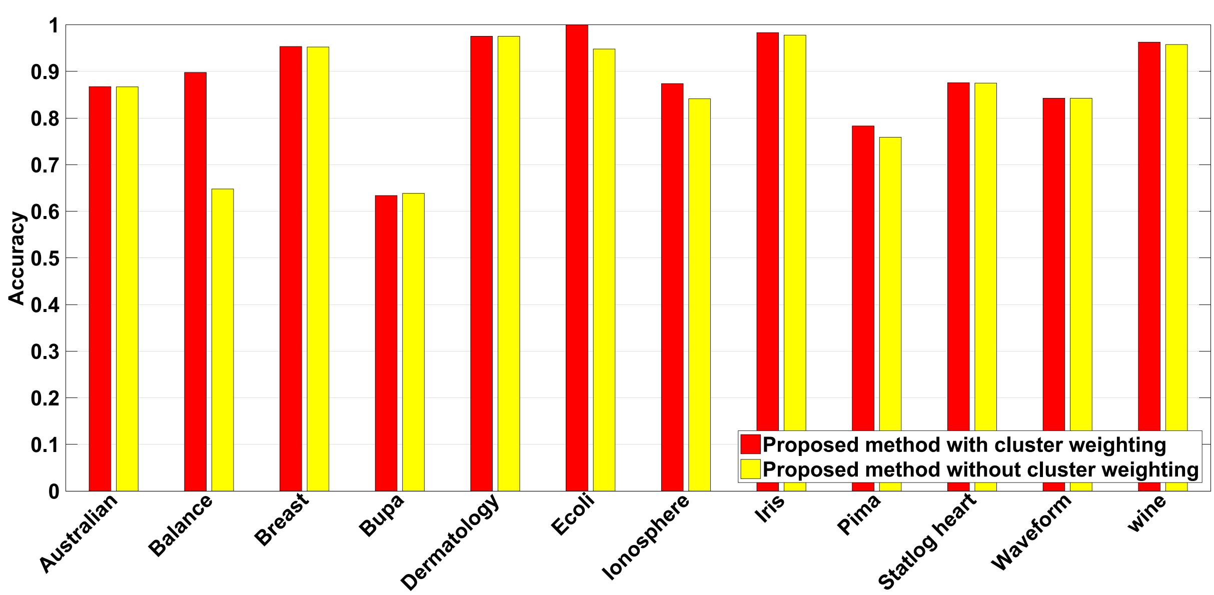

Fig. 4. The impact of cluster weighting on the clustering quality.

clusters in different datasets, such as *Balance*, *Ecoli*, *Pima*, and *Ionosphere*, have relatively different levels of importance.

Figs. 3 and 4 show that the weighting of features, as opposed to the weighting of clusters, substantially impacts the formation of optimal clusters. Although utilizing cluster weighting strategies improves average results, the performance of the proposed approach is comparable to the no cluster weighting option for several datasets, including *Dermatology* and *Bupa*. The proposed technique produces better clusters when the dataset has well-balanced natural groupings. However, this approach may be problematic because it may hinder the clustering process from disclosing the data's inherent structure when imbalanced natural groupings are present in the dataset, a common scenario in practice, such as *Dermatology* and *Bupa* datasets.

5.5. Experiment 3: Effect of labeled data

In this experiment, we test our approach for different ratios to study the behavior of the proposed algorithm for the proportion of labeled data. For this purpose, we test the proposed technique with labeled data rates ranging from 5 % to 45 % with 5 % steps.

Fig. 5 compares the accuracy of our algorithm for increasing the percentage of labeled data. The proposed semi-supervised model performs better with more labeled data. As label information increases, the degree of hesitation decreases, and the proposed model tends to operate with nearly the same accuracy for high percentages.

Cluster centers more accurately identify the class's more genuine centroids as the amount of labeled data increases. As a result, the amount of labeled data significantly impacts the clustering procedure. Some datasets have a significantly imbalanced sample distribution of classes. As a result, not all classes may be covered by the labeled data that was randomly chosen. As a result, the accuracy of this dataset is not increased by an increase in the number of labeled data.

5.6. Experiment 4: Proposed method vs. state-of-the-art methods

This experiment evaluates the performance of the proposed method against various unsupervised and semi-supervised approaches. Table 2 displays the outcomes of different approaches. The table provides the accuracy rate for each technique. This table further shows the improvement rate of the outcomes produced by the proposed approach compared to the top outcome produced by the competing algorithms. The results of alternative methods have been quoted from the relevant literature.

According to Table 2, the proposed technique has the best accuracy for all datasets. All compared methods are less accurate than the proposed method for the tested datasets. For *wine*, *Australian*, *Ionosphere*, *Waveform*, *Balance*, and *Bupa* datasets, the proposed method has the greatest improvement compared to the result of the best result of other algorithms. In these datasets, the accuracy has been improved by >21.58 %.on average. This shows that the proposed method has significantly improved the results in 50 % of the tested datasets. This illustrates the critical role that the proposed techniques play in improving clustering. After the proposed method, the LHC-S3FCM and CS3FCM methods perform well, respectively.

Table 3 shows the average results for the proposed method and other compared algorithms. As shown in this table, our method has the best results. In terms of *Accuracy* criteria, after the proposed method, the methods AS3FCM, SFC-ER, LHC-S3FCM, CS3FCM, S3FCM, and SSFCM have better results, respectively. The methods TS3FCM, SMKIFC, Seeded- k-means, and FCM have the worst results, respectively. The improvement percentage of the proposed method compared to the second-best method (AS3FCM method) is 14.10 %. This value shows that the technique of weighting clusters and features has been able to significantly increase the accuracy of clustering.

For a thorough analysis of the proposed method with baseline algorithms, Fig. 6 displays the outcomes for various combinations of the used technique in the proposed method and the two baseline methods, SSFCM and FWCW-FCM.

This illustration shows how the proposed approach with all strategies outperforms the baseline techniques across all evaluated datasets. This supports the results of our experiment and demonstrates that our proposed approach enhances the accuracy of the semi-supervised clustering task. All methods were compared to one another to assess each algorithm.

5.6.1. Proposed method vs. SSFCM

As demonstrated in Fig. 6, all versions of the proposed technique outperform SSFCM on average across all datasets. Compared to SSFCM, the proposed algorithm has an average accuracy of 88.75 %. This illustrates how our proposed technique enhances the performance of the base SSFCM method. Even without including cluster weighting, feature weighting, or neighbors, the proposed technique performs better than SSFCM. This is due to the non-linear distance criterion used in our algorithm. The accuracy of the proposed method in the earlier case is

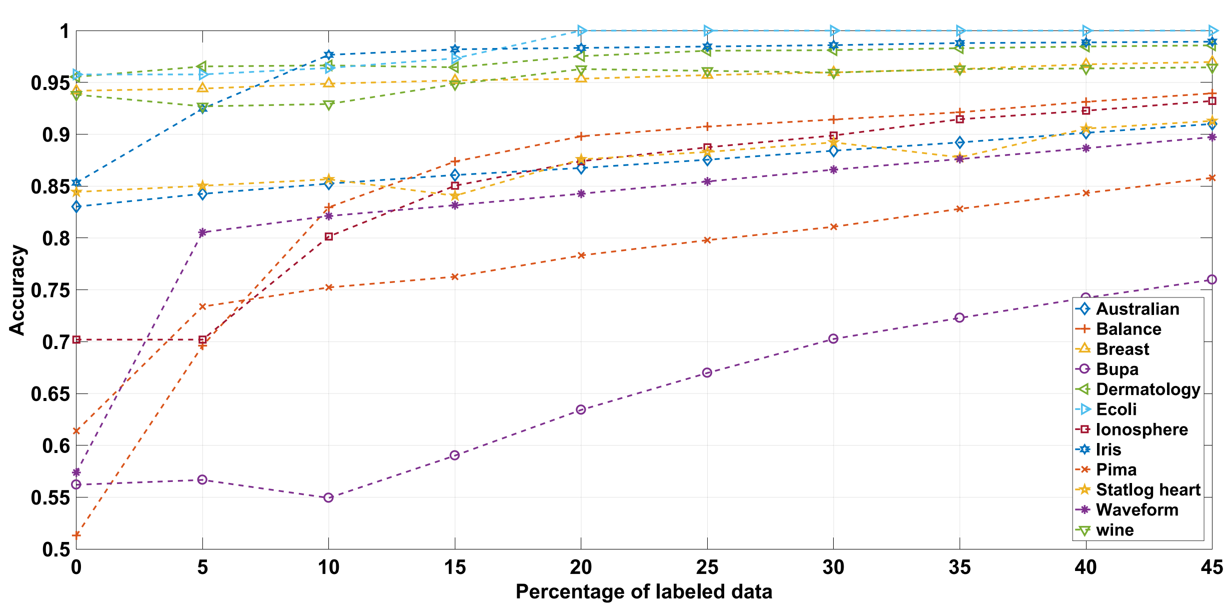

Fig. 5. The behavior of the proposed algorithm for the percentage of labeled data.

Table 2
Proposed method vs. State-of-the-art methods

	Australian	Balance	Breast cancer	Bupa	Dermatology	Ecoli	Ionosphere	Iris	Pima	Statlog (heart)	Waveform	wine
TS3FCM	–	0.5680	–	–	0.6690	–	–	0.7640	–	0.7720	–	0.6210
Seeded- k-means	0.5650	0.5300	0.8500	0.4600	0.9400	–	–	0.8910	0.6780	0.7900	0.5000	0.6800
FCM	0.5600	0.5100	0.8500	0.5200	0.7800	0.8460	0.7090	0.8930	0.6800	0.7970	0.4900	0.6700
S3FCM	0.6090	–	0.8520	–	–	–	–	0.9000	–	0.8010	0.5400	–
SSFCM	0.5970	0.5340	0.8520	0.5210	0.9100	0.9370	0.7120	0.9050	0.6970	0.8100	0.5400	0.6800
CS3FCM	0.6350	0.5260	0.8620	0.5600	0.9100	–	0.5400	0.9480	0.7200	0.8150	0.7600	0.6900
LHC-S3FCM	0.6300	0.7900	0.8620	0.5650	0.9400	–	0.7200	0.9500	0.7250	0.8180	0.7650	0.7000
SFC-ER	–	–	0.8610	0.5220	0.6000	0.9640	–	0.9330	0.7180	0.8250	0.7700	–
AS3FCM	–	0.7400	0.8680	0.5720	0.9550	–	–	–	0.7190	0.8300	0.7610	–
Our	**0.8675**	**0.8980**	**0.9536**	**0.6339**	**0.9754**	**1**	**0.8740**	**0.9833**	**0.7832**	**0.8759**	**0.8427**	**0.9629**
Improvement percentage	**36.61**	**13.67**	**9.86**	**10.82**	**2.13**	**3.09**	**21.38**	**1.68**	**8.02**	**5.53**	**9.44**	**37.55**

Table 3
The average results for the proposed method and other compared algorithms.

TS3FCM	Seeded- k-means	FCM	S3FCM	SSFCM	CS3FCM	LHC-S3FCM	SFC-ER	AS3FCM	Our
0.6788	0.6884	0.69208	0.7404	0.7245	0.7241	0.7695	0.7741	0.7778	**0.8875**

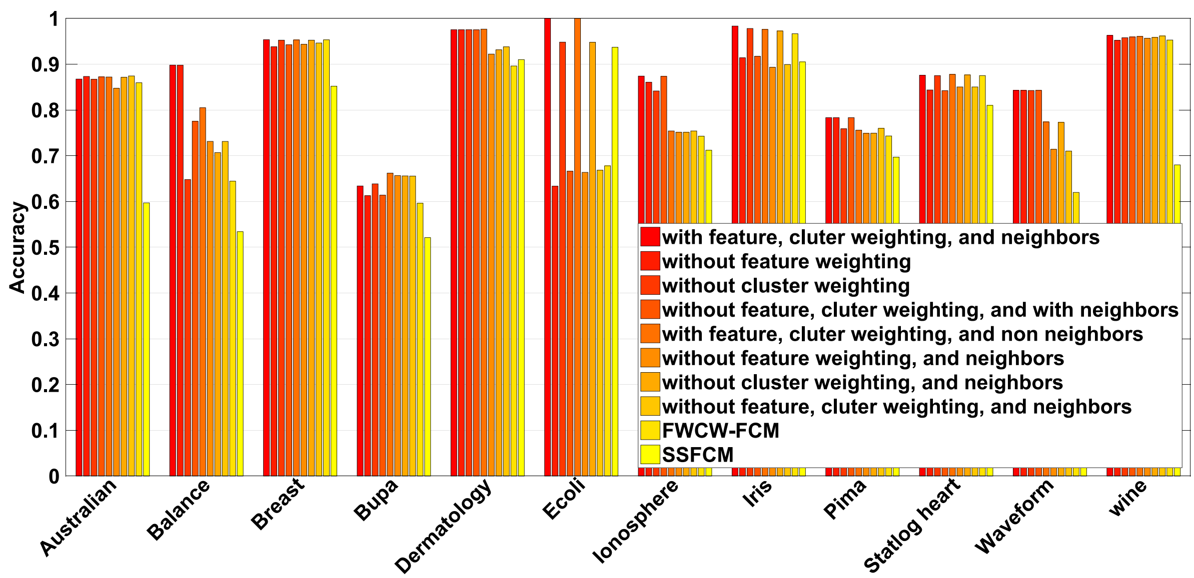

Fig. 6. Proposed method vs. baseline methods.

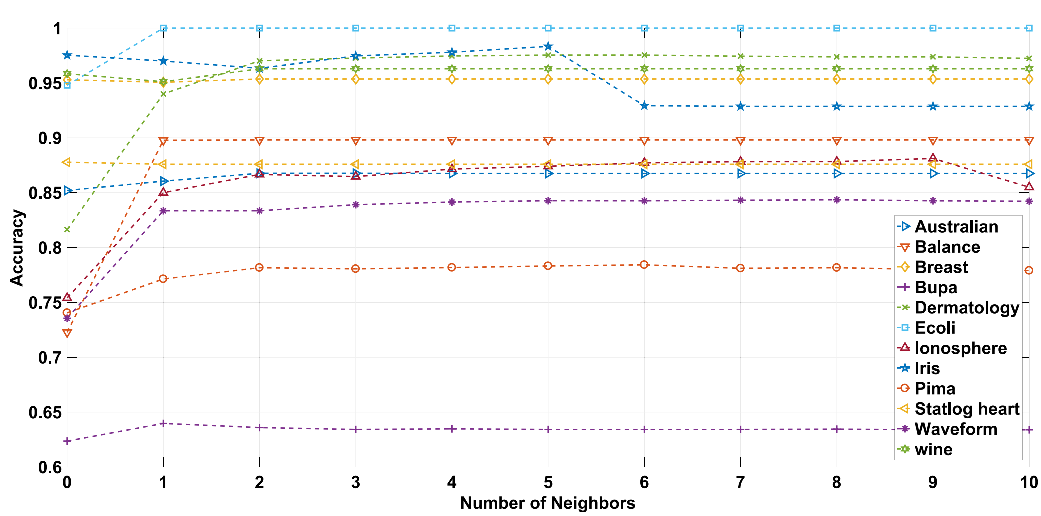

Fig. 7. The behavior of the proposed algorithm for the number of neighbors.

81.24 %.

5.6.2. Proposed method vs. FWCW-FCM

For all datasets, all versions of the proposed approach yield better results on average than FWCW-FCM. The average accuracy of the proposed method is 88.75 %, compared to FWCW-FCM's 79.40 %. This demonstrates how our approach performs better than the baseline FWCW-FCM algorithm. It is simple to understand why the proposed approach is superior to FWCW-FCM. Our method uses semi-supervised learning, whereas the FWCW-FCM method is unsupervised. On the other side, data neighborhood has been employed, but the FWCW-FCM technique does not use data neighborhood.

5.6.3. SSFCM vs. FWCW-FCM

The comparison of these two methodologies is critical. The SSFCM method is based on semi-supervised learning, whereas the FWCW-FCM approach is unsupervised. SSFCM is likely to generate better results because it incorporates information from some labels in the dataset. However, regarding accuracy, this approach is 6.94 % worse than FWCW-FCM. This discovery highlights the significance of feature and cluster weighting. Although the FWCW-FCM method lacks a mechanism for managing labeled samples, it has generated superior results by weighting features and clusters.

The following results are obtained by examining the three techniques used in the proposed method and comparing these methods without using the mentioned technique. These results are presented with a complete peer-to-peer comparison with and without the mentioned

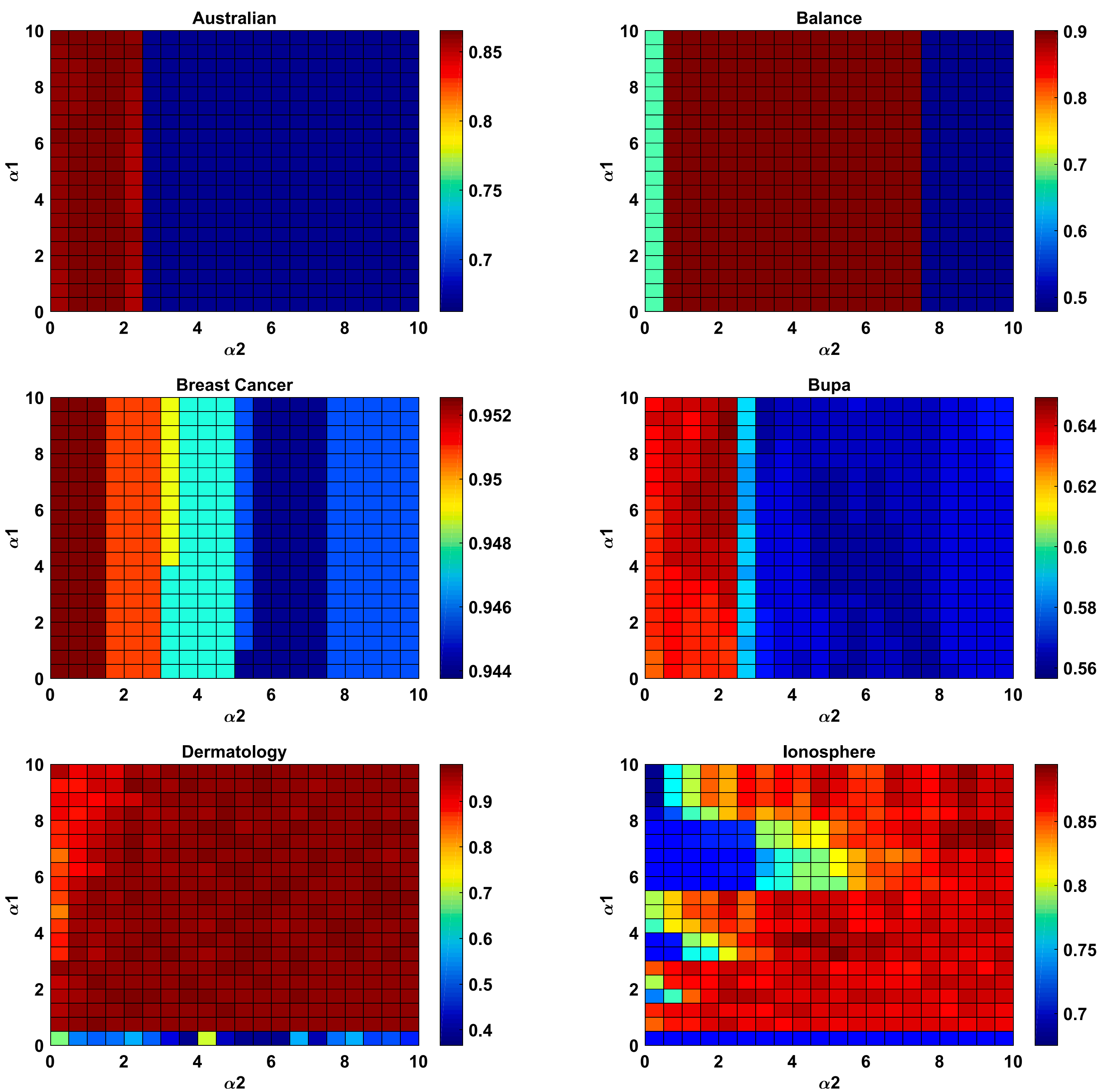

Fig. 8. The effectiveness of our technique for different values of α_1 and α_2.

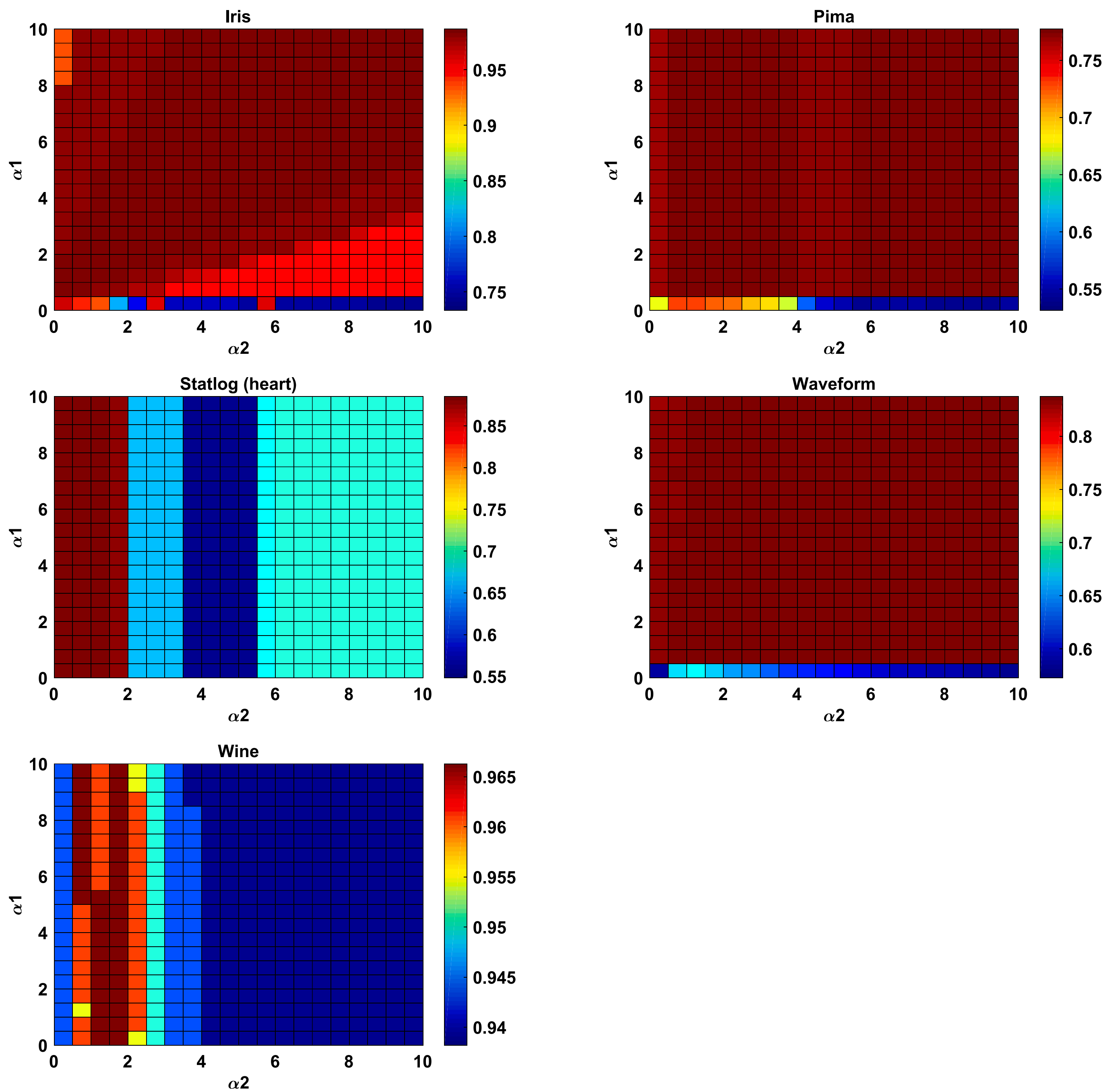

Fig. 8. (*continued*).

technique.

1. Considering the neighborhood, the average results have improved by 3.14 %.
2. Considering the weighting of the features, the average results have improved by 5.42 %.
3. Considering the weighting of the clusters, the average results have improved by 1.64 %.

5.7. Experiment 5: Effect of neighbors

To study the behavior of the proposed algorithm for the number of neighbors, we test the proposed method for different numbers of neighbors in this experiment. For this purpose, we tend the proposed method with different neighbors from 0 to 10 with step 1. A value of zero means that no neighbors are used.

Fig. 7 compares the effectiveness of our proposed technique for raising the number of neighbors in terms of accuracy. Without using any neighbors, the average accuracy is 82.98 %. As the number of neighbors increases, the average accuracy increases on all datasets so that for a value of 5, it reaches its maximum value of 88.76 %. From neighborhood number 6 onwards, the accuracy decreases, so for the number 10, the accuracy decreases to 88 %. The reason for this is quite apparent. As the number of neighbors increases, more distant neighbors and possibly outliers are considered nearest neighbors.

5.8. Experiment 6: Effect of α_1 and α_2 parameters

To study the behavior of the proposed algorithm according to α_1 and α_2 parameters, we test the proposed method for different values in this experiment. For this purpose, we tend the proposed method with the value from 0 to 10 with step 0.5 for both parameters.

Fig. 8 shows the effectiveness of our technique for different values of α_1 and α_2 in terms of the accuracy. The results show that, on average, when α_1 and α_2 parameters are in the range $[0.5, 2]$, then the accuracy is high in almost all datasets. In the experiments, we set the value of these two parameters to 1. For the value of 1 for all datasets, the average accuracy is 87.64 %, the most significant accuracy among all the different values. Regardless of the value of α_1 ($\alpha_1 = 0$), for all values of α_2 $\alpha_2 \epsilon [0, 10]$, the average accuracy is 71.31 %. While without considering α_2 ($\alpha_2 = 0$), the accuracy is 82.89 %. This shows that α_1 (that is, neighborhood) has a more significant impact on the results, so the accuracy decreases by about 12 % regardless of neighborhood.

6. Conclusion

Many studies have been performed on semi-supervised clustering, which has grown to be a popular research topic. Despite the efficiency advantage of using auxiliary class information during clustering, several issues remain unaddressed. Most semi-supervised fuzzy clustering algorithms ignore feature weight and cluster weight learning, which prevents the formation of an ideal clustering structure. On the other hand, semi-supervised algorithms have low accuracy since they do not consider sample local information.

The semi-supervised fuzzy objective function proposed in this paper is based on prior class knowledge, feature weighting, cluster weighting, and neighborhood. During the clustering process, the feature weighting technique adaptively provides a different weight to each feature associated with each cluster. Additionally, a sample's neighbors act like a regularizer, increasing the proposed method's accuracy. Several experimental investigations were carried out to compare the performance of the proposed strategy with the most recent methods using several benchmark datasets. Furthermore, the impact of feature weighting, cluster weighting, and using neighbors on the algorithm's efficiency was investigated.

This is the first attempt to add feature weight and cluster weight learning in semi-supervised fuzzy clustering. There are still some unexplored areas in this field which are: 1) extending this objective function to situations where the provided previous knowledge includes instances incorrectly labeled and 2) noise can affect the finding of correct neighbors. Hence, the accuracy may decrease due to noise. Finding neighbors not sensitive to noise seems like a promising research direction.

CRediT authorship contribution statement

Ali Kadhim Jasim: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Resources, Software, Validation, Visualization, Writing – original draft. **Jafar Tanha:** Conceptualization, Formal analysis, Project administration, Supervision, Writing – review & editing. **Mohammad Ali Balafar:** Supervision, Writing – review & editing.

Declaration of competing interest

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

Data availability

The authors do not have permission to share data.

Appendix A. Appendix

In this section, we describe how to obtain the update equations for u_{nk}, c_{km}, w_{km}, and z_k. First, we write the Lagrange function of Eq. (10):

$$\widetilde{F} = \sum_{n=1}^{N}\sum_{k=1}^{K}\sum_{m=1}^{M} {u_{nk}}^2 w_{km}^q z_k^p d^2(x_{nm} - c_{km}) + \frac{\alpha_1}{N_R}\sum_{k=1}^{K}\sum_{n=1}^{N} {u_{nk}}^2 \sum_{r\in N_n}(1-u_{rk})^2 + \alpha_2 \sum_{n=1}^{N}\sum_{k=1}^{K} \times \sum_{m=1}^{M}(u_{nk} - b_n f_{nk})^2 w_{km}^q z_k^p d^2(x_{nm} - c_{km}) - \delta\left(\sum\nolimits_{k=1}^{K} u_{nk} - 1\right) - \psi\left(\sum\nolimits_{m=1}^{M} w_{km} - 1\right) - \omega\left(\sum\nolimits_{k=1}^{K} z_k - 1\right) \tag{A.1}$$

where δ، ψ و ω are the parameters of the Lagrange multiplier.

Theorem 1. *If* $\boldsymbol{C}$, $\mathbf{W}$, *and* $\boldsymbol{Z}$ *are assumed to be fixed,* $\boldsymbol{U}$ *is a strict local minimum of* $\mathbf{F}(\mathbf{U})$ *if and only if* $\boldsymbol{U}$ is calculated using Eq. (14).

Proof. Considering the gradient of $\widetilde{\mathrm{F}}(\boldsymbol{U})$ to u_{nk} and set equal to 0, we obtain:

$$\frac{\partial \widetilde{\mathrm{F}}}{\partial u_{nk}} = 0$$

$$\sum\nolimits_{m=1}^{M} 2\, u_{nk}\, w_{km}^{q}\, z_k^p d^2(x_{nm} - c_{km}) + \frac{\alpha_1}{N_R} 2\, u_{nk} \sum_{r\in N_n} (1 - u_{rk})^2 + \alpha_2 \sum\nolimits_{m=1}^{M} (2\, u_{nk} - 2b_n f_{nk})\, w_{km}^{q}\, z_k^p d^2(x_{nm} - c_{km}) - \delta = 0$$

$$2u_{nk} \sum\nolimits_{m=1}^{M} w_{km}^{q}\, z_k^p d^2(x_{nm} - c_{km}) + \frac{\alpha_1}{N_R} 2\, u_{nk} \sum_{r\in N_n} (1 - u_{rk})^2 + 2\alpha_2 (u_{nk} - b_n f_{nk}) \sum\nolimits_{m=1}^{M} w_{km}^{q}\, z_k^p d^2(x_{nm} - c_{km}) - \delta = 0$$

$$(\, 2u_{nk} + 2\alpha_2 u_{nk} - 2\alpha_2 b_n f_{nk}) \sum\nolimits_{m=1}^{M} w_{km}^{q}\, z_k^p d^2(x_{nm} - c_{km}) + \frac{\alpha_1}{N_R} 2\, u_{nk} \sum_{r\in N_l} (1 - u_{rk})^2 = \delta \tag{A.2}$$

$$2u_{nk} \left((1+\alpha_2) \sum\nolimits_{m=1}^{M} w_{km}^{q}\, z_k^p d^2(x_{nm} - c_{km}) + \frac{\alpha_1}{N_R} \sum_{r\in N_n} (1 - u_{rk})^2 \right) - 2\alpha_2 b_n f_{nk} \sum\nolimits_{m=1}^{M} w_{km}^{q}\, z_k^p d^2(x_{nm} - c_{km}) = \delta$$

$$2u_{nk} = \frac{\delta + 2\alpha_2 b_n f_{nk} \sum_{m=1}^{M} w_{km}^{q}\, z_k^p d^2(x_{nm} - c_{km})}{(1+\alpha_2) \sum_{m=1}^{M} w_{km}^{q}\, z_k^p d^2(x_{nm} - c_{km}) + \frac{\alpha_1}{N_R} \sum_{r\in N_n} (1 - u_{rk})^2}$$

Now, considering the gradient of $\widetilde{\mathrm{F}}(\boldsymbol{U})$ to u_{nl} and set equal to 0, we obtain δ:

$$\frac{\partial \widetilde{\mathrm{F}}}{\partial u_{nl}} = 0$$

$$\sum\nolimits_{m=1}^{M} 2\, u_{nl}\, w_{lm}^{q}\, z_l^p d^2(x_{nm} - c_{lm}) + \frac{\alpha_1}{N_R} 2\, u_{nl} \sum_{r\in N_n} (1 - u_{rl})^2 + \alpha_2 \sum\nolimits_{m=1}^{M} (2\, u_{nl} - 2b_n f_{nl})\, w_{lm}^{q}\, z_l^p d^2(x_{nm} - c_{lm}) - \delta = 0$$

$$\delta = \sum\nolimits_{m=1}^{M} 2\, u_{nl}\, w_{lm}^{q}\, z_l^p d^2(x_{nm} - c_{lm}) + \frac{\alpha_1}{N_R} 2\, u_{nl} \sum_{r\in N_n} (1 - u_{rl})^2 + \alpha_2 \sum\nolimits_{m=1}^{M} (2\, u_{nl} - 2b_n f_{nl})\, w_{lm}^{q}\, z_l^p d^2(x_{nm} - c_{lm}) \tag{A.3}$$

$$\delta = 2u_{nl} \left((1+\alpha_2) \sum\nolimits_{m=1}^{M} w_{lm}^{q}\, z_l^p d^2(x_{nm} - c_{lm}) + \frac{\alpha_1}{N_R} \sum_{r\in N_n} (1 - u_{rl})^2 \right) - 2\alpha_2 b_n f_{nl} \sum\nolimits_{m=1}^{M} w_{lm}^{q}\, z_l^p d^2(x_{nm} - c_{lm})$$

It follows from Eqs. (A.2) and (A.3) that:

$$2u_{nk} = \frac{2u_{nl} \left((1+\alpha_2) \sum_{m=1}^{M} w_{lm}^{q} z_l^p d^2(x_{nm} - c_{lm}) + \frac{\alpha_1}{N_R} \sum_{r\in N_n} (1 - u_{rl})^2 \right) - 2\alpha_2 b_n f_{nl} \sum_{m=1}^{M} w_{lm}^{q} z_l^p d^2(x_{nm} - c_{lm}) + 2\alpha_2 b_n f_{nk} \sum_{m=1}^{M} w_{km}^{q} z_k^p d^2(x_{nm} - c_{km})}{(1+\alpha_2) \sum_{m=1}^{M} w_{km}^{q} z_k^p d^2(x_{nm} - c_{km}) + \frac{\alpha_1}{N_R} \sum_{r\in N_l} (1 - u_{rk})^2}$$

$$u_{nk} = \frac{u_{nl} \left((1+\alpha_2) \sum_{m=1}^{M} w_{lm}^{q} z_l^p d^2(x_{nm} - c_{lm}) + \frac{\alpha_1}{N_R} \sum_{r\in N_n} (1 - u_{rl})^2 \right) - \alpha_2 b_n f_{nl} \sum_{m=1}^{M} w_{lm}^{q} z_l^p d^2(x_{nm} - c_{lm}) + \alpha_2 b_n f_{nk} \sum_{m=1}^{M} w_{km}^{q} z_k^p d^2(x_{nm} - c_{km})}{(1+\alpha_2) \sum_{m=1}^{M} w_{km}^{q} z_k^p d^2(x_{nm} - c_{km}) + \frac{\alpha_1}{N_R} \sum_{r\in N_n} (1 - u_{rk})^2}$$

$$u_{nl} = \frac{u_{nk} \left((1+\alpha_2) \sum_{m=1}^{M} w_{km}^{q} z_k^p d^2(x_{nm} - c_{km}) + \frac{\alpha_1}{N_R} \sum_{r\in N_n} (1 - u_{rk})^2 \right) + \alpha_2 b_n f_{nl} \sum_{m=1}^{M} w_{lm}^{q} z_l^p d^2(x_{nm} - c_{lm}) - \alpha_2 b_n f_{nk} \sum_{m=1}^{M} w_{km}^{q} z_k^p d^2(x_{nm} - c_{km})}{(1+\alpha_2) \sum_{m=1}^{M} w_{lm}^{q} z_l^p d^2(x_{nm} - c_{lm}) + \frac{\alpha_1}{N_R} \sum_{r\in N_n} (1 - u_{rl})^2}$$

$$u_{nl} = \frac{u_{nk} \left((1+\alpha_2) \sum_{m=1}^{M} w_{km}^{q} z_k^p d^2(x_{nm} - c_{km}) + \frac{\alpha_1}{N_R} \sum_{r\in N_n} (1 - u_{rk})^2 \right) - \alpha_2 b_n f_{nk} \sum_{m=1}^{M} w_{km}^{q} z_k^p d^2(x_{nm} - c_{km}) + \alpha_2 b_n f_{nl} \sum_{m=1}^{M} w_{lm}^{q} z_l^p d^2(x_{nm} - c_{lm})}{(1+\alpha_2) \sum_{m=1}^{M} w_{lm}^{q} z_l^p d^2(x_{nm} - c_{lm}) + \frac{\alpha_1}{N_R} \sum_{r\in N_n} (1 - u_{rl})^2}$$

$$u_{nl} = u_{nk} \frac{(1+\alpha_2) \sum_{m=1}^{M} w_{km}^{q} z_k^p d^2(x_{nm} - c_{km}) + \frac{\alpha_1}{N_R} \sum_{r\in N_n} (1 - u_{rk})^2}{(1+\alpha_2) \sum_{m=1}^{M} w_{lm}^{q} z_l^p d^2(x_{nm} - c_{lm}) + \frac{\alpha_1}{N_R} \sum_{r\in N_n} (1 - u_{rl})^2} - \frac{\alpha_2 b_n f_{nk} \sum_{m=1}^{M} w_{km}^{q} z_k^p d^2(x_{nm} - c_{km})}{(1+\alpha_2) \sum_{m=1}^{M} w_{lm}^{q} z_l^p d^2(x_{nm} - c_{lm}) + \frac{\alpha_1}{N_R} \sum_{r\in N_n} (1 - u_{rl})^2} + \frac{\alpha_2 b_n f_{nl} \sum_{m=1}^{M} w_{lm}^{q} z_l^p d^2(x_{nm} - c_{lm})}{(1+\alpha_2) \sum_{m=1}^{M} w_{lm}^{q} z_l^p d^2(x_{nm} - c_{lm}) + \frac{\alpha_1}{N_R} \sum_{r\in N_n} (1 - u_{rl})^2} \tag{A.4}$$

considering $\widehat{d}_{nl} = (1+\alpha_2) \sum_{m=1}^{M} w_{lm}^{q} z_l^p d^2(x_{nm} - c_{lm}) + \frac{\alpha_1}{N_R} \sum_{r\in N_n} (1 - u_{rl})^2$, $\sum_{l=1}^{K} u_{nl} = 1$, and $\widehat{d}_{nk} = (1+\alpha_2) \sum_{m=1}^{M} w_{km}^{q} z_k^p d^2(x_{nm} - c_{km}) + \frac{\alpha_1}{N_R} \sum_{r\in N_n} (1 - u_{rk})^2$, Eq. (A.5) is obtained:

$$u_{nl} = u_{nk} \frac{\widehat{d}_{nk}}{\widehat{d}_{nl}} - \frac{\alpha_2 b_n f_{nk} \sum_{m=1}^{M} w_{km}^{q}\, z_k^p d^2(x_{nm} - c_{km})}{\widehat{d}_{nl}} + \frac{\alpha_2 b_n f_{nl} \sum_{m=1}^{M} w_{lm}^{q}\, z_l^p d^2(x_{nm} - c_{lm})}{\widehat{d}_{nl}}$$

$$\sum_{l=1}^{K} u_{nl} = \sum_{l=1}^{K} u_{nk}\frac{\widehat{d}_{nk}}{\widehat{d}_{nl}} - \frac{\alpha_2 b_n f_{nk} \sum_{m=1}^{M} w_{km}^{q}\, z_k^p d^2(x_{nm}-c_{km})}{\widehat{d}_{nl}} + \frac{\alpha_2 b_n f_{nl} \sum_{m=1}^{M} w_{lm}^{q}\, z_l^p d^2(x_{nm}-c_{lm})}{\widehat{d}_{nl}}$$

$$1 = \sum_{l=1}^{K} u_{nk}\frac{\widehat{d}_{nk}}{\widehat{d}_{nl}} - \sum_{l=1}^{K}\frac{\alpha_2 b_n f_{nk} \sum_{m=1}^{M} w_{km}^{q}\, z_k^p d^2(x_{nm}-c_{km})}{\widehat{d}_{nl}} + \sum_{l=1}^{K}\frac{\alpha_2 b_n f_{nl} \sum_{m=1}^{M} w_{lm}^{q}\, z_l^p d^2(x_{nm}-c_{lm})}{\widehat{d}_{nl}}$$

$$1 = u_{nk}\sum_{l=1}^{K} \frac{\widehat{d}_{nk}}{\widehat{d}_{nl}} - \sum_{l=1}^{K}\frac{\alpha_2 b_n f_{nk} \sum_{m=1}^{M} w_{km}^{q}\, z_k^p d^2(x_{nm}-c_{km})}{\widehat{d}_{nl}} + \sum_{l=1}^{K}\frac{\alpha_2 b_n f_{nl} \sum_{m=1}^{M} w_{lm}^{q}\, z_l^p d^2(x_{nm}-c_{lm})}{\widehat{d}_{nl}}$$

$$u_{nk} = \frac{1 + \sum_{l=1}^{K}\frac{\alpha_2 b_n f_{nk} \sum_{m=1}^{M} w_{km}^{q}\, z_k^p d^2(x_{nm}-c_{km})}{\widehat{d}_{nl}} - \sum_{l=1}^{K}\frac{\alpha_2 b_n f_{nl} \sum_{m=1}^{M} w_{lm}^{q}\, z_l^p d^2(x_{nm}-c_{lm})}{\widehat{d}_{nl}}}{\sum_{l=1}^{K}\frac{\widehat{d}_{nk}}{\widehat{d}_{nl}}}$$

As a result, the membership updating function is obtained.

$$u_{nk} = \frac{1 + \sum_{l=1}^{K}\frac{\alpha_2 b_n f_{nk} \sum_{m=1}^{M} w_{km}^{q}\, z_k^p d^2(x_{nm}-c_{km})}{(1+\alpha_2)\sum_{m=1}^{M} w_{lm}^{q}\, z_l^p d^2(x_{nm}-c_{lm}) + \frac{\alpha_1}{N_R}\sum_{r\in N_n}(1-u_{rl})^2} - \sum_{l=1}^{K}\frac{\alpha_2 b_n f_{nl} \sum_{m=1}^{M} w_{lm}^{q}\, z_l^p d^2(x_{nm}-c_{lm})}{(1+\alpha_2)\sum_{m=1}^{M} w_{lm}^{q}\, z_l^p d^2(x_{nm}-c_{lm}) + \frac{\alpha_1}{N_R}\sum_{r\in N_n}(1-u_{rl})^2}}{\sum_{l=1}^{K}\frac{(1+\alpha_2)\sum_{m=1}^{M} w_{km}^{q}\, z_k^p d^2(x_{nm}-c_{km}) + \frac{\alpha_1}{N_R}\sum_{r\in N_n}(1-u_{rk})^2}{(1+\alpha_2)\sum_{m=1}^{M} w_{lm}^{q}\, z_l^p d^2(x_{nm}-c_{lm}) + \frac{\alpha_1}{N_R}\sum_{r\in N_n}(1-u_{rl})^2}} \tag{A.5}$$

Theorem 2. *If U, $\mathbf{W}$, and Z are assumed to be fixed, C is a strict local minimum of $\mathbf{F}(\mathbf{C})$ if and only if C* is calculated using Eq. (15).

Proof. Considering the gradient of $\widetilde{\mathrm{F}}(\boldsymbol{C})$ to c_{km} and set equal to 0, we obtain:

$$\frac{\partial\widetilde{\mathrm{F}}}{\partial c_{km}} = 0$$

$$\sum_{n=1}^{N} 2{u_{nk}}^2 w_{km}^{q}\, z_k^p(-\gamma_m)(x_{nm}-c_{km})\left(exp\left(-\gamma_m(x_{nm}-c_{km})^2\right)\right) + \alpha_2\sum_{n=1}^{N} 2(u_{nk}-b_n f_{nk})^2\, w_{km}^{q} z_k^p(-\gamma_m)(x_{nm}-c_{km})\left(exp\left(-\gamma_m(x_{nm}-c_{km})^2\right)\right) = 0 \tag{A.6}$$

The following is an extension of Eq. (A.6):

$$\begin{aligned}
&2\, w_{km}^{q}\, z_k^p\,(-\gamma_m)\sum_{n=1}^{N} {u_{nk}}^2\,(x_{nm})\left(exp\left(-\gamma_m(x_{nm}-c_{km})^2\right)\right)\\
&-2\, w_{km}^{q}\, z_k^p\,(-\gamma_m)\,(c_{km})\sum_{n=1}^{N} {u_{nk}}^2\,\left(exp\left(-\gamma_m(x_{nm}-c_{km})^2\right)\right)\\
&+\alpha_2 2\, w_{km}^{q}\, z_k^p\,(-\gamma_m)\sum_{n=1}^{N}(u_{nk}-b_n f_{nk})^2\,(x_{nm})\left(exp\left(-\gamma_m(x_{nm}-c_{km})^2\right)\right)\\
&-\alpha_2 2\, w_{km}^{q}\, z_k^p\,(-\gamma_m)\,(c_{km})\sum_{n=1}^{N}(u_{nk}-b_n f_{nk})^2\,\left(exp\left(-\gamma_m(x_{nm}-c_{km})^2\right)\right) = 0
\end{aligned} \tag{A.7}$$

$$c_{km} = \frac{\sum_{n=1}^{N} {u_{nk}}^2\,(x_{nm})\left(exp\left(-\gamma_m(x_{nm}-c_{km})^2\right)\right) + \alpha_2\sum_{n=1}^{N}(u_{nk}-b_n f_{nk})^2\,(x_{nm})\left(exp\left(-\gamma_m(x_{nm}-c_{km})^2\right)\right)}{\sum_{n=1}^{N} {u_{nk}}^2\,\left(exp\left(-\gamma_m(x_{nm}-c_{km})^2\right)\right) + \alpha_2\sum_{n=1}^{N}(u_{nk}-b_n f_{nk})^2\,\left(exp\left(-\gamma_m(x_{nm}-c_{km})^2\right)\right)}$$

$$c_{km} = \frac{\sum_{n=1}^{N}\left({u_{nk}}^2\,(x_{nm})\left(exp\left(-\gamma_m(x_{nm}-c_{km})^2\right)\right) + \alpha_2(u_{nk}-b_n f_{nk})^2\,(x_{nm})\left(exp\left(-\gamma_m(x_{nm}-c_{km})^2\right)\right)\right)}{\sum_{n=1}^{N}\left({u_{nk}}^2\,\left(exp\left(-\gamma_m(x_{nm}-c_{km})^2\right)\right) + \alpha_2(u_{nk}-b_n f_{nk})^2\,\left(exp\left(-\gamma_m(x_{nm}-c_{km})^2\right)\right)\right)}$$

$$c_{km} = \frac{\sum_{n=1}^{N}(x_{nm})\left(exp\left(-\gamma_m(x_{nm}-c_{km})^2\right)\right)\left({u_{nk}}^2 + \alpha_2(u_{nk}-b_n f_{nk})^2\right)}{\sum_{n=1}^{N}\left(exp\left(-\gamma_m(x_{nm}-c_{km})^2\right)\right)\left({u_{nk}}^2 + \alpha_2(u_{nk}-b_n f_{nk})^2\right)}$$

As a result, the center updating function is obtained.

Theorem 3. *If U, $\mathbf{C}$, and Z are assumed to be fixed, W is a strict local minimum of $\mathbf{F}(\mathbf{W})$ if and only if W* is calculated using Eq. (16).

Proof. Considering the gradient of $\widetilde{\mathrm{F}}(\boldsymbol{W})$ to w_{km} and set equal to 0, we obtain:

$$\begin{aligned}
&\frac{\partial\widetilde{\mathrm{F}}}{\partial w_{km}} = 0\\
&q\, w_{km}^{q-1}\, z_k^p\sum_{n=1}^{N} {u_{nk}}^2\, d^2(x_{nm}-c_{km}) + \alpha_2 q\, w_{km}^{q-1}\, z_k^p\sum_{n=1}^{N}(u_{nk}-b_n f_{nk})^2\, d^2(x_{nm}-c_{km}) - \psi = 0\\
&q\, w_{km}^{q-1}\, z_k^p\left[\sum_{n=1}^{N} {u_{nk}}^2\, d^2(x_{nm}-c_{km}) + \alpha_2\sum_{n=1}^{N}(u_{nk}-b_n f_{nk})^2\, d^2(x_{nm}-c_{km})\right] = \psi\\
&w_{km}^{q-1} = \frac{\psi}{q z_k^p\left[\sum_{n=1}^{N} {u_{nk}}^2\, d^2(x_{nm}-c_{km}) + \alpha_2\sum_{n=1}^{N}(u_{nk}-b_n f_{nk})^2\, d^2(x_{nm}-c_{km})\right]}\\
&w_{km} = \left[\frac{\psi}{q z_k^p\left[\sum_{n=1}^{N} {u_{nk}}^2\, d^2(x_{nm}-c_{km}) + \alpha_2\sum_{n=1}^{N}(u_{nk}-b_n f_{nk})^2\, d^2(x_{nm}-c_{km})\right]}\right]^{\frac{1}{q-1}}
\end{aligned} \tag{A.8}$$

Now, Considering the gradient of $\widetilde{\mathrm{F}}(\boldsymbol{W})$ to w_{ks} and set equal to 0, we get ψ:

$$\frac{\partial\widetilde{\mathrm{F}}}{\partial w_{ks}} = 0$$

$$\psi = q\, w_{ks}^{q-1}\, z_k^p \left[\sum\nolimits_{n=1}^{N} u_{nk}{}^2\, d^2(x_{ns} - c_{ks}) + \alpha_2 \sum\nolimits_{n=1}^{N} (u_{nk} - b_n f_{nk})^2\, d^2(x_{ns} - c_{ks}) \right] \tag{A.9}$$

We can rewrite from Eqs. (A.8) and (A.9):

$$\begin{aligned}
w_{km} &= \left[\frac{w_{ks}^{q-1} \left[\sum_{n=1}^{N} u_{nk}{}^2\, d^2(x_{ns} - c_{ks}) + \alpha_2 \sum_{n=1}^{N} (u_{nk} - b_n f_{nk})^2\, d^2(x_{ns} - c_{ks}) \right]}{\left[\sum_{n=1}^{N} u_{nk}{}^2\, d^2(x_{nm} - c_{km}) + \alpha_2 \sum_{n=1}^{N} (u_{nk} - b_n f_{nk})^2\, d^2(x_{nm} - c_{km}) \right]} \right]^{\frac{1}{q-1}} \\
w_{km} &= w_{ks} \left[\frac{\sum_{n=1}^{N} u_{nk}{}^2\, d^2(x_{ns} - c_{ks}) + \alpha_2 \sum_{n=1}^{N} (u_{nk} - b_n f_{nk})^2\, d^2(x_{ns} - c_{ks})}{\sum_{n=1}^{N} u_{nk}{}^2\, d^2(x_{nm} - c_{km}) + \alpha_2 \sum_{n=1}^{N} (u_{nk} - b_n f_{nk})^2\, d^2(x_{nm} - c_{km})} \right]^{\frac{1}{q-1}} \\
w_{ks} &= w_{km} \left[\frac{\sum_{n=1}^{N} u_{nk}{}^2\, d^2(x_{nm} - c_{km}) + \alpha_2 \sum_{n=1}^{N} (u_{nk} - b_n f_{nk})^2\, d^2(x_{nm} - c_{km})}{\sum_{n=1}^{N} u_{nk}{}^2\, d^2(x_{ns} - c_{ks}) + \alpha_2 \sum_{n=1}^{N} (u_{nk} - b_n f_{nk})^2\, d^2(x_{ns} - c_{ks})} \right]^{\frac{1}{q-1}} \\
\sum_{s=1}^{M} w_{ks} &= \sum_{s=1}^{M} w_{km} \left[\frac{\sum_{n=1}^{N} \left[u_{nk}{}^2 + \alpha_2 (u_{nk} - b_n f_{nk})^2 \right] d^2(x_{nm} - c_{km})}{\sum_{n=1}^{N} \left[u_{nk}{}^2 + \alpha_2 (u_{nk} - b_n f_{nk})^2 \right] d^2(x_{ns} - c_{ks})} \right]^{\frac{1}{q-1}}
\end{aligned} \tag{A.10}$$

Considering $\sum_{s=1}^{M} w_{ks} = 1$, Eq. (A.11) is obtained:

$$\begin{aligned}
1 &= \sum_{s=1}^{M} w_{km} \left[\frac{\sum_{n=1}^{N} \left[u_{nk}{}^2 + \alpha_2 (u_{nk} - b_n f_{nk})^2 \right] d^2(x_{nm} - c_{km})}{\sum_{n=1}^{N} \left[u_{nk}{}^2 + \alpha_2 (u_{nk} - b_n f_{nk})^2 \right] d^2(x_{ns} - c_{ks})} \right]^{\frac{1}{q-1}} \\
w_{km} &= \frac{1}{\sum_{s=1}^{M} \left[\frac{\sum_{n=1}^{N} \left[u_{nk}{}^2 + \alpha_2 (u_{nk} - b_n f_{nk})^2 \right] d^2(x_{nm} - c_{km})}{\sum_{n=1}^{N} \left[u_{nk}{}^2 + \alpha_2 (u_{nk} - b_n f_{nk})^2 \right] d^2(x_{ns} - c_{ks})} \right]^{\frac{1}{q-1}}}
\end{aligned} \tag{A.11}$$

Thus, the updating function of the feature weighting is obtained.

Theorem 3. *If U, $\mathbf{C}$, and W are assumed to be fixed, Z is a strict local minimum of $\mathbf{F}(\mathbf{Z})$ if and only if Z* is calculated using Eq. (17).

Proof. Considering the gradient of $\widetilde{\mathrm{F}}(\mathbf{Z})$ to z_k and set equal to 0, we obtain:

$$\frac{\partial\widetilde{\mathrm{F}}}{\partial z_k} = 0$$

$$\begin{aligned}
&\sum_{n=1}^{N} \sum_{m=1}^{M} u_{nk}{}^2 w_{km}^q \mathrm{p}\, z_k^{p-1} d^2(x_{nm} - c_{km}) + \alpha_2 \sum_{n=1}^{N} \sum_{m=1}^{M} (u_{nk} - b_n f_{nk})^2 w_{km}^q\, \mathrm{p}\, z_k^{p-1} d^2(x_{nm} - c_{km}) - \omega = 0 \\
&\mathrm{p}\, z_k^{p-1} \sum_{n=1}^{N} \sum_{m=1}^{M} u_{nk}{}^2 w_{km}^q d^2(x_{nm} - c_{km}) + \alpha_2 \mathrm{p}\, z_k^{p-1} \sum_{n=1}^{N} \sum_{m=1}^{M} (u_{nk} - b_n f_{nk})^2 w_{km}^q\, d^2(x_{nm} - c_{km}) - \omega = 0 \\
&\mathrm{p}\, z_k^{p-1} \left[\sum_{n=1}^{N} \sum_{m=1}^{M} u_{nk}{}^2 w_{km}^q d^2(x_{nm} - c_{km}) + \alpha_2 \sum_{n=1}^{N} \sum_{m=1}^{M} (u_{nk} - b_n f_{nk})^2 w_{km}^q\, d^2(x_{nm} - c_{km}) \right] = \omega \\
&z_k = \left[\frac{\omega}{\mathrm{p} \left[\sum_{n=1}^{N} \sum_{m=1}^{M} u_{nk}{}^2 w_{km}^q d^2(x_{nm} - c_{km}) + \alpha_2 \sum_{n=1}^{N} \sum_{m=1}^{M} (u_{nk} - b_n f_{nk})^2 w_{km}^q\, d^2(x_{nm} - c_{km}) \right]} \right]^{\frac{1}{p-1}}
\end{aligned} \tag{A.12}$$

Now, considering the gradient of $\widetilde{\mathrm{F}}(\mathbf{Z})$ to z_l and set equal to 0, we get ω:

$$\frac{\partial\widetilde{\mathrm{F}}}{\partial z_l} = 0$$

$$\begin{aligned}
\omega &= \mathrm{p}\, z_l^{p-1} \sum_{n=1}^{N} \sum_{k=1}^{K} u_{nl}{}^m\, w_{lm}^q\, d^2(x_{nm} - c_{lm}) + \alpha_2 \mathrm{p}\, z_l^{p-1} \sum_{n=1}^{N} \sum_{m=1}^{M} (u_{nl} - b_n f_{nl})^2 w_{lm}^q\, d^2(x_{nm} - c_{lm}) \\
\omega &= \mathrm{p}\, z_l^{p-1} \left[\sum_{n=1}^{N} \sum_{k=1}^{K} u_{nl}{}^m\, w_{lm}^q\, d^2(x_{nm} - c_{lm}) + \alpha_2 \sum_{n=1}^{N} \sum_{m=1}^{M} (u_{nl} - b_n f_{nl})^2 w_{lm}^q\, d^2(x_{nm} - c_{lm}) \right]
\end{aligned} \tag{A.13}$$

We can deduct from Eqs. (A.12) and (A.13) that:

$$z_k = \left[\frac{\mathrm{p}\, z_l^{p-1} \left[\sum_{n=1}^{N} \sum_{k=1}^{K} {u_{nl}}^m\, w_{lm}^q\, d^2(x_{nm} - c_{lm}) + \alpha_2 \sum_{n=1}^{N} \sum_{m=1}^{M} \left(u_{nl} - b_n f_{nl}\right)^2 w_{lm}^q\, d^2(x_{nm} - c_{lm}) \right]}{\mathrm{p} \left[\sum_{n=1}^{N} \sum_{m=1}^{M} {u_{nk}}^2 w_{km}^q d^2(x_{nm} - c_{km}) + \alpha_2 \sum_{n=1}^{N} \sum_{m=1}^{M} \left(u_{nk} - b_n f_{nk}\right)^2 w_{km}^q\, d^2(x_{nm} - c_{km}) \right]} \right]^{\frac{1}{p-1}}$$

$$z_k = z_l \left[\frac{\left[\sum_{n=1}^{N} \sum_{k=1}^{K} {u_{nl}}^m\, w_{lm}^q\, d^2(x_{nm} - c_{lm}) + \alpha_2 \sum_{n=1}^{N} \sum_{m=1}^{M} \left(u_{nl} - b_n f_{nl}\right)^2 w_{lm}^q\, d^2(x_{nm} - c_{lm}) \right]}{\left[\sum_{n=1}^{N} \sum_{m=1}^{M} {u_{nk}}^2 w_{km}^q d^2(x_{nm} - c_{km}) + \alpha_2 \sum_{n=1}^{N} \sum_{m=1}^{M} \left(u_{nk} - b_n f_{nk}\right)^2 w_{km}^q\, d^2(x_{nm} - c_{km}) \right]} \right]^{\frac{1}{p-1}}$$

$$z_l = z_k \left[\frac{\left[\sum_{n=1}^{N} \sum_{m=1}^{M} {u_{nk}}^2 w_{km}^q d^2(x_{nm} - c_{km}) + \alpha_2 \sum_{n=1}^{N} \sum_{m=1}^{M} \left(u_{nk} - b_n f_{nk}\right)^2 w_{km}^q\, d^2(x_{nm} - c_{km}) \right]}{\left[\sum_{n=1}^{N} \sum_{k=1}^{K} {u_{nl}}^m\, w_{lm}^q\, d^2(x_{nm} - c_{lm}) + \alpha_2 \sum_{n=1}^{N} \sum_{m=1}^{M} \left(u_{nl} - b_n f_{nl}\right)^2 w_{lm}^q\, d^2(x_{nm} - c_{lm}) \right]} \right]^{\frac{1}{p-1}} \tag{A.14}$$

$$z_l = z_k \left[\frac{\sum_{n=1}^{N} \sum_{k=1}^{K} w_{km}^q\, d^2(x_{nm} - c_{km}) \left[{u_{nk}}^2 + \alpha_2 (u_{nk} - b_n f_{nk})^2 \right]}{\sum_{n=1}^{N} \sum_{k=1}^{K} w_{lm}^q\, d^2(x_{nm} - c_{lm}) \left[{u_{nl}}^2 + \alpha_2 (u_{nl} - b_n f_{nl})^2 \right]} \right]^{\frac{1}{p-1}}$$

$$\sum_{l=1}^{K} z_l = \sum_{l=1}^{K} z_k \left[\frac{\sum_{n=1}^{N} \sum_{k=1}^{K} w_{km}^q\, d^2(x_{nm} - c_{km}) \left[{u_{nk}}^2 + \alpha_2 (u_{nk} - b_n f_{nk})^2 \right]}{\sum_{n=1}^{N} \sum_{k=1}^{K} w_{lm}^q\, d^2(x_{nm} - c_{lm}) \left[{u_{nl}}^2 + \alpha_2 (u_{nl} - b_n f_{nl})^2 \right]} \right]^{\frac{1}{p-1}}$$

Considering that $\sum_{l=1}^{K} z_l = 1$, Eq. (A.15) is obtained:

$$1 = z_k \sum_{l=1}^{K} \left[\frac{\sum_{n=1}^{N} \sum_{k=1}^{K} w_{km}^q\, d^2(x_{nm} - c_{km}) \left[{u_{nk}}^2 + \alpha_2 (u_{nk} - b_n f_{nk})^2 \right]}{\sum_{n=1}^{N} \sum_{k=1}^{K} w_{lm}^q\, d^2(x_{nm} - c_{lm}) \left[{u_{nl}}^2 + \alpha_2 (u_{nl} - b_n f_{nl})^2 \right]} \right]^{\frac{1}{p-1}}$$

$$z_k = \frac{1}{\sum_{l=1}^{K} \left[\frac{\sum_{n=1}^{N} \sum_{k=1}^{K} w_{km}^q\, d^2(x_{nm} - c_{km}) \left[{u_{nk}}^2 + \alpha_2 (u_{nk} - b_n f_{nk})^2 \right]}{\sum_{n=1}^{N} \sum_{k=1}^{K} w_{lm}^q\, d^2(x_{nm} - c_{lm}) \left[{u_{nl}}^2 + \alpha_2 (u_{nl} - b_n f_{nl})^2 \right]} \right]^{\frac{1}{p-1}}} \tag{A.15}$$

Thus, the updating function of the cluster weighting is obtained.

References

[1] Nabatian M, Tanha J, Ebrahimzadeh AR, Phirouznia A. An adaptive scaling technique to quantum clustering. International Journal of Modern Physics C 2023; 34(01):2350002. https://doi.org/10.1142/s012918312350002x.

[2] Tanha J, Samadi N, Abdi Y, Razzaghi-Asl N. CPSSDS: conformal prediction for semi-supervised classification on data streams. Inf. Sci. 2022;584:212–34.

[3] Taghizabet A, Tanha J, Amini A, Mohammadzadeh J. A semi-supervised clustering approach using labeled data. Scientia Iranica 2023;30(1):104–15. https://doi.org/10.24200/sci.2022.58519.5772.

[4] Oskouei AG, Balafar MA, Akan T. A brain MRI segmentation method using feature weighting and a combination of efficient visual features. In: Applied computer vision and soft computing with interpretable AI. Chapman and Hall/CRC; 2024. p. 15–34.

[5] Ikotun AM, Ezugwu AE, Abualigah L, Abuhaija B, Heming J. K-means clustering algorithms: a comprehensive review, variants analysis, and advances in the era of big data. Inf. Sci. 2022;622:178–210.

[6] Borlea I-D, Precup R-E, Borlea A-B, Iercan D. A unified form of fuzzy C-means and K-means algorithms and its partitional implementation. Knowl.-Based Syst. 2021; 214:106731.

[7] Ferraro MB, Giordani P. Soft clustering. Wiley Interdisciplinary Reviews: Computational Statistics 2020;12(1):e1480.

[8] Golzari Oskouei A, Balafar MA, Motamed C. EDCWRN: efficient deep clustering with the weight of representations and the help of neighbors. Appl. Intell. 2023;53 (5):5845–67. https://doi.org/10.1007/s10489-022-03895-5. 03/01 2023.

[9] Sivarathri S, Govardhan A. Experiments on hypothesis fuzzy k-means is better than k-means for clustering. International Journal of Data Mining & Knowledge Management Process (IJDKP) 2014;4.

[10] Golzari Oskouei A, Balafar MA, Motamed C. RDEIC-LFW-DSS: ResNet-based deep embedded image clustering using local feature weighting and dynamic sample selection mechanism. Inf. Sci. 2023;646:119374. https://doi.org/10.1016/j.ins.2023.119374. 10/01/ 2023.

[11] Van Engelen JE, Hoos HH. A survey on semi-supervised learning. Mach. Learn. 2020;109(2):373–440.

[12] Reddy Y, Viswanath P, Reddy BE. Semi-supervised learning: a brief review. Int J Eng Technol 2018;7(1.8):81.

[13] Qin Y, Ding S, Wang L, Wang Y. Research progress on semi-supervised clustering. Cogn. Comput. 2019;11:599–612.

[14] González-Almagro G, Luengo J, Cano J-R, García S. Enhancing instance-level constrained clustering through differential evolution. Appl. Soft Comput. 2021; 108:107435.

[15] Yin X, Chen S, Hu E, Zhang D. Semi-supervised clustering with metric learning: an adaptive kernel method. Pattern Recogn. 2010;43(4):1320–33.

[16] Yin X, Shu T, Huang Q. Semi-supervised fuzzy clustering with metric learning and entropy regularization. Knowl.-Based Syst. 2012;35:304–11.

[17] De Amorim RC, Mirkin B. Minkowski metric, feature weighting and anomalous cluster initializing in K-means clustering. Pattern Recogn. 2012;45(3):1061–75.

[18] Basu S. Semi-supervised clustering by seeding. In: Proc. ICML; 2002. p. 2002.

[19] Pedrycz W, Waletzky J. Fuzzy clustering with partial supervision. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) 1997;27(5): 787–95.

[20] Bilenko M, Basu S, Mooney RJ. Integrating constraints and metric learning in semi-supervised clustering. In: Proceedings of the twenty-first international conference on Machine learning; 2004. p. 11.

[21] Wei S, Li Z, Zhang C. Combined constraint-based with metric-based in semi-supervised clustering ensemble. Int. J. Mach. Learn. Cybern. 2018;9:1085–100.

[22] Gan H. Safe semi-supervised fuzzy ${C} $-means clustering. IEEE Access 2019;7: 95659–64.

[23] Gan H, Fan Y, Luo Z, Huang R, Yang Z. Confidence-weighted safe semi-supervised clustering. Eng. Appl. Artif. Intell. 2019;81:107–16.

[24] Gan H, Fan Y, Luo Z, Zhang Q. Local homogeneous consistent safe semi-supervised clustering. Expert Syst. Appl. 2018;97:384–93.

[25] Gan H, Yang Z, Zhou R. Adaptive safety-aware semi-supervised clustering. Expert Syst. Appl. 2023;212:118751.

[26] Huan PT, et al. TS3FCM: trusted safe semi-supervised fuzzy clustering method for data partition with high confidence. Multimed. Tools Appl. 2022;81(9):12567–98.

[27] Tuan TM, Thong PH, Ngan TT. An improvement of trusted safe semi-supervised fuzzy clustering method with multiple fuzzifiers. Journal of Computer Science and Cybernetics 2022;38(1):47–61.

[28] Casalino G, Castellano G, Mencar C. Data stream classification by dynamic incremental semi-supervised fuzzy clustering. International Journal on Artificial Intelligence Tools 2019;28(08):1960009.
[29] Hashemzadeh M, Oskouei AG, Farajzadeh N. New fuzzy C-means clustering method based on feature-weight and cluster-weight learning. Appl. Soft Comput. 2019;78:324–45.
[30] Li H, Wei M. Fuzzy clustering based on feature weights for multivariate time series. Knowl.-Based Syst. 2020;197:105907.
[31] Abu Arqub O, Singh J, Alhodaly M. Adaptation of kernel functions-based approach with Atangana–Baleanu–Caputo distributed order derivative for solutions of fuzzy fractional Volterra and Fredholm integrodifferential equations. Mathematical Methods in the Applied Sciences 2023;46(7):7807–34. https://doi.org/10.1002/mma.7228.
[32] Abu Arqub O, Singh J, Maayah B, Alhodaly M. Reproducing kernel approach for numerical solutions of fuzzy fractional initial value problems under the Mittag–Leffler kernel differential operator. Mathematical Methods in the Applied Sciences 2023;46(7):7965–86. https://doi.org/10.1002/mma.7305.
[33] Abu Arqub O. Adaptation of reproducing kernel algorithm for solving fuzzy Fredholm–Volterra integrodifferential equations. Neural Comput. & Applic. 2017; 28(7):1591–610. https://doi.org/10.1007/s00521-015-2110-x. 07/01 2017.
[34] Taha K. Semi-supervised and un-supervised clustering: a review and experimental evaluation. Inf. Syst. 2023;114:102–78. 102178.
[35] Zeng S, Tong X, Sang N, Huang R. A study on semi-supervised FCM algorithm. Knowl. Inf. Syst. 2013;35:585–612.
[36] Pedrycz W, Vukovich G. Fuzzy clustering with supervision. Pattern Recogn. 2004; 37(7):1339–49.
[37] Bouchachia A, Pedrycz W. Data clustering with partial supervision. Data Min. Knowl. Disc. 2006;12:47–78.
[38] Grira N, Crucianu M, Boujemaa N. Active semi-supervised fuzzy clustering. Pattern Recogn. 2008;41(5):1834–44.
[39] Yasunori E, Yukihiro H, Makito Y, Sadaaki M. On semi-supervised fuzzy c-means clustering. In: 2009 IEEE International Conference on Fuzzy Systems. IEEE; 2009. p. 1119–24.
[40] Gao C-F, Wu X-J. A new semi-supervised clustering algorithm with pairwise constraints by competitive agglomeration. Appl. Soft Comput. 2011;11(8): 5281–91.
[41] Salehi F, Keyvanpour MR, Sharifi A. SMKFC-ER: semi-supervised multiple kernel fuzzy clustering based on entropy and relative entropy. Inf. Sci. 2021;547:667–88.
[42] Tuan TM, et al. A new approach for semi-supervised fuzzy clustering with multiple Fuzzifiers. International Journal of Fuzzy Systems 2022;24(8):3688–701.
[43] Bezdek JC, Ehrlich R, Full W. FCM: the fuzzy c-means clustering algorithm. Comput. Geosci. 1984;10(2–3):191–203.