Multiple Fuzzification Coefficients in a Fuzzy C-Means Clustering Algorithm

Sanjeev Kumar Chatterjee[1], Nikita Thakur[2]
[1]*Research Scholar, Sai Nath University, Ranchi Jharkhand*
[2]*Associate Professor, Sai Nath University, Ranchi Jharkhand*

Abstract: Clustering is a well researched unsupervised machine learning technique with numerous real-world applications. Besides probabilistic or deterministic methods, fuzzy C-means clustering (FCM) is another popular method for clustering. Clustering efficiency has improved significantly since the FCM method was introduced. These enhancements concentrate on modifying the distance function between elements and the membership representation of the elements in the clusters, or on fuzzifying and defuzzifying methods. This paper suggests a novel fuzzy clustering algorithm that makes use of several fuzzification factors, which are chosen based on the properties of individual data samples.

With a few adjustments, the suggested fuzzy clustering approach uses computation steps that are comparable to FCM. Convergence is guaranteed by deriving the formulas. Using numerous fuzzification coefficients instead of the one coefficient used in the original FCM method is the primary contribution of this approach. Experiments on a number of widely used datasets are then used to assess the new algorithm, and the findings indicate that it is more effective than both the original FCM and alternative clustering techniques.

Keywords: clustering technique; fuzzy clustering; fuzzy C-means clustering; fuzzification coefficient; objective function; performance indices; clustering efficiency; machine learning

1. INTRODUCTION

Clustering is a machine learning technique that involves the grouping of data points into different clusters, where data points in the same cluster have a higher degree of similarity and any two data points in two different clusters have a lower degree of similarity. This technique improvesthe effectiveness of data mining, especially in big-data problems, as the data points are grouped intoclusters with distinctive characteristics [1]. There are many different clustering techniques, such as centroid- based, hierarchical, density-based, distribution-based and grid-based clustering.

In practical problems, some data elements may be missing information or containing uncertaininformation. The fuzzy set theory provides an appropriate method of representing and processing these types of data elements, along with the concept of membership function defined in the range [0,1]. In this concept, each element can belong to more than one cluster. The fuzzy C-means (FCM) algorithm [2], a method of fuzzy clustering, is an efficient algorithm for extracting rules and miningdata from a dataset in which the fuzzy properties are highly common [3]. Many new methods basedon the FCM algorithm were introduced, in order to overcome the limitations and improve the clustering ability of this algorithm in different cases. The performance of these new methods is summarized in [4]. The new and improved FCM-based methods extend or modify the distance metric between elements and cluster center, use different fuzzy measures for the membership of an elementto a cluster or modify the exponential parameter for fuzzifying [3,5].

An improvement direction is given by Hwang et al. in [6], which is the interval type-2 fuzzy C-means clustering (FCMT2I) algorithm. The study gives a fuzzy extension of the fuzzificationcoefficient m, thereby applying the type-2 fuzzy set into the membership matrix U. The parameter$m$ in the FCMT2I algorithm can be any value in the range $[m_L, m_U]$. A few articles [7–10] discuss this approach further; after every iteration, an element belonging to the clusters is represented by aninterval type-2 fuzzy set and further calculations are needed for type-reduction when calculating the cluster centers for the next steps. According to the type-reduction algorithm [6,10], the cluster centers are calculated separately with the exponential parameters m_L, m_U, and then combined through the defuzzification step. In essence, after the type-reduction algorithm is executed, the resulting clustercenter value is the same as the value calculated using an exponential parameter m in the range $[m_L, m_U]$. The exponential parameter m is used to adjust the membership level of the elements in the clusters and its value can vary with clusters and steps. Thus, the FCMT2I algorithm allows the use of different exponential parameters.

This study presents an approach that utilizes multiple different fuzzification coefficients as opposed to only one coefficient in the FCM algorithm. While in other past papers, the determinationof fuzzification coefficients corresponds to the type-reduction calculations for the cluster centers, inthis study, the fuzzification coefficients are determined for each element prior to calculations, depending on the characteristics of such element in the entire dataset. In a dataset, the density of each element, represented by its distance from the "neighboring" elements, is a useful measurement [6,11,12]. This study uses that measurement to determine the exponential parameter for each element during the clustering process. The main contribution of this study is the proposition of the use and manipulation of multiple fuzzification coefficients to enhance the performance of the original FCMalgorithm. The experimental evaluation of the efficiency of the proposed algorithm is conducted using several common datasets from the University of California, Irvine (UCI) machine learning repository.

The rest of this study is organized as follows: Section 2 presents the FCM Algorithm and improvement ideas; Section 3 describes the proposed novel MC-FCM algorithm; and Section 4 provides the experimental results for algorithm evaluation.

2. PRELIMINARIES

2.1. Standard Fuzzy C-Means (FCM) Algorithm

The FCM algorithm [2] partitions a finite collection of N elements $X = \{X_1, X_2, ..., X_N\}$ into a collection of C clusters with respect to some given criteria. Each element $X_i \in X$, $i = 1,2,..., N$ is a vector with d dimensions. We define a way to divide X into C clusters with cluster centers $V_1, V_2, ..., V_C$ in the centroid set V.

In the FCM algorithm, U is a representative matrix for the membership of each element in each cluster. The matrix U has some characteristics as below:

- $U(i,k)$ is the membership value of the element X_i in a cluster with center V_k, $1 \le i \le N$; $1 \le k \le C$
- $0 \le U(i,k) \le 1$, $1 \le i \le N$; $1 \le k \le C$ and $\sum_{k=1}^{C} U(i,k) = 1$ for each X_i.
- The larger $U(i,k)$ is, the higher the degree of confidence that the element X_i belongs to the cluster k.

An objective function is defined such that the clustering algorithm minimizes the objectivefunction

$$J(U,V) = \sum_{i=1}^{N}\sum_{k=1}^{C} U(i,k)^m D(i,k)^2 \quad (1)$$

where,

- $D(i,k)^2 = \|X_i - V_k\|_2$ is the squared distance between the element X_i and the cluster center V_k.
- m is the fuzzification coefficient of the algorithm.

Summary of steps for the FCM algorithm:

Input: the dataset $X = \{X_1, X_2, ..., X_N\}$, the fuzzification coefficient m.

Output: the partition of X into C clusters.

Steps:

- Step 1: Initialize value for V, let $l = 0$, set $\varepsilon > 0$ and $m > 1$.
- Step 2: At the $l-th$ loop, update U according to the formula:

$$U(i,k) = \left[\sum_{j=1}^{C}\left(\frac{D(i,k)}{D(i,j)}\right)^{\frac{2}{m-1}}\right]^{-1} \quad (2)$$

- Step 3: Update V for the next step $(l+1)$, according to the formula:

$$V_k = \frac{\sum_{i=1}^{N} U(i,k)^m X_i}{\sum_{i=1}^{N} U(i,k)^m} \quad (3)$$

- Step 4: If $\|V^{(l)} - V^{(l-1)}\| < \varepsilon$ then go to Step 5; otherwise, let $l = l + 1$ and return to Step 2.

- Step 5: End.

2.2. Interval Type-2 Fuzzy C-Means (FCMT2I) Clustering

FCM is a breakthrough algorithm compared to other well-known clustering algorithms becauseFCM has a more general representation and more accurate clustering results in many cases. However, the FCM algorithm has some disadvantages:

- The algorithm is sensitive to noise and foreign elements;
- Clustering is not accurate and valid for elements located at the boundary between clusters;
- There is no specific criterion to select the value for the parameter m , which is often selectedafter testing multiple times.

With these disadvantages, the FCM algorithm requires more research and development to befurther improved. A research [6] on the extension or modification of the distance metrics $D(i,k)$ uses different fuzzy metrics for $U(i,k)$ or discusses the exponential parameter m . The clustering decisions made in FCM is based on Equation (2), which presents the fuzzy membership of element X_i that is assigned by the relative distance between X_i and V_k . This formula has a characteristic that is, the smaller m , the higher the absolute membership U , as shown in Figure 1 by the case of $C = 2$.

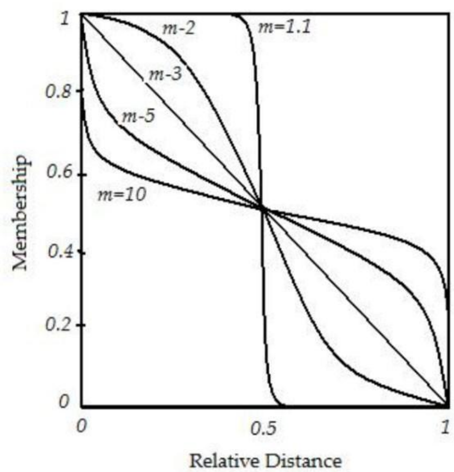

Figure 1. Membership of an element to a cluster for different values of m.

For the elements on the cluster boundary, the absolute degree of U must be lower than the elements near the center of the cluster. The reason is that we cannot make a clustering decision withelements at the immediate boundary, so we need the U graph to be more flexible in terms of clusterselection for these elements. Meanwhile, the elements near the center of the cluster should have a steeper U graph to ensure that the clustering control is always in place in the nearest cluster. Since FCM only uses a single m value, the absolute level of U is the same for the elements near the center of the cluster and the elements at the boundary, which is not reasonable. Hwang et al. [6] studied the elements on the boundary of two clusters. It was found that when choosing a large m for the degree of U to cluster center V_1 and choosing a small m for the degree of U to cluster center V_2, the cluster boundary would be distributed with a wide left region and narrow right region, as shown in Figure 2.

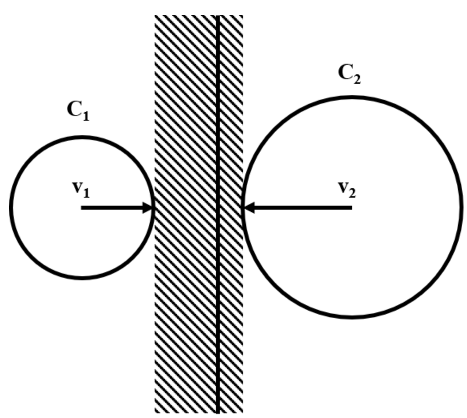

Figure 2. The cluster boundary between two clusters for different values of m.

As seen from the above analysis, different exponential parameters m can be determineddepending on the distance between the element and the cluster center. The interval type-2 fuzzy C-means clustering (FCMT2I) method, proposed in [6], uses two parameters m_L and m_U. These calculations are derived from Equations (2) and (3). However, the membership of X_i in the cluster

k becomes a range of values, hence, the calculation of the cluster center in FCMT2I requires additional steps from the type-reduction algorithm to obtain a single final value. Therefore, even though the performance of the FCMT2I algorithm is, in general, better than that of the FCM with one exponential parameter, the FCMT2I algorithm requires more calculations because the exponential parameters depend on the distance to the cluster center and the cluster center changes after every iteration.

The proposed idea in this study is to assign each element X_i an exponential parameter m_i which is constant throughout the entire clustering process. The determination of m_i depends on the characteristics of the corresponding element X_i in the dataset.

3. **Fuzzy C–Means Clustering with Multiple Fuzzification Coefficients (MC-FCM)**

3.1. The Fuzzification Coefficients

Each element is assigned its own fuzzification coefficient. The fuzzification coefficient of an element is calculated based on the concentration of other elements surrounding that element. If theconcentration is high, meaning a high chance to create a cluster, the fuzzification coefficient will besmall for faster convergence. If the concentration is low, the fuzzification coefficient will be large toincrease the ability to select a cluster through iterations.

With each element X_i, $1 \le i \le N$, let δ_{ij} be the distance between the elements X_i and X_j, $1 \le j \le N$. In the dataset, we can treat the total distance between X_i and N/C elements with the closest distance to X_i as a heuristic metric for the concentration surrounding X_i, notated δ_i^*. Afterward, the fuzzification coefficient m_i can be calculated from δ_i^*, as shown below.

Summary of steps for computing m_i

Input: the dataset $X = \{X_1, X_2, ..., X_N\}$, the interval for fuzzification coefficients $[m_L, m_U]$, $m_L > 1$ and a parameter .

Output: the fuzzification coefficient m_i for each element X_i.

Steps:

- Step 1: Calculate the distance between two elements $\delta_{ij} = \| X_i - X_j \|$, $1 \le i \le N$; $1 \le j \le N$.
- Step 2: Rearrange δ_{ij} with index j, we have $\delta'_{i1}, \delta'_{i2}, ..., \delta'_{iN}$ in non-decreasing order.

Calculate $\delta_i^* = \sum_{j=1}^{N/C} \delta'_{ij}$

$i1 \quad i2 \quad iN$

- Step 3: Calculate $\delta_{min} = \min_{i=1,\ldots,N} \{\delta_i^*\}$ and $\delta_{max} = \max_{i=1,\ldots,N}\{\delta_i^*\}$. For each X_i, $1 \le i \le N$, calculate

$$m_i = m_L + (m_U - m_L)\left(\frac{\delta_i - \delta_{min}}{\delta_{max} - \delta_{min}}\right)^{\alpha} \tag{4}$$

The algorithm to calculate m_i is carried out once before clustering, with a complexity of $O(N^2)$. The parameter α is utilized to adjust the mapping of the distance between X_i and its neighboring elements with the exponential parameter in the range of $[m_L, m_U]$. If $\alpha = 1$, it indicates linear mapping; if $\alpha < 1$, the distribution is skewed toward m_U; and if $\alpha > 1$, the distribution skewed toward m_L.

3.2. Derivation of the MC-FCM Clustering Algorithm

In this section, we derive the fuzzy C-means clustering algorithm with multiple fuzzification coefficients (MC-FCM). The objective function to be minimized in the MC-FCM algorithm isformulated as follows,

$$J(U,V) = \sum_{i=1}^{N}\sum_{k=1}^{C} U(i,k)^{m_i} D(i,k)^2 \tag{5}$$

where $X = \{X_1, X_2, ..., X_N\}$ is the dataset, N is the number of elements, C is the number of clusters, $U(i,k)$ is the membership value of element X_i in the cluster with center V_k, $0 \le U(i,k) \le 1$, $1 \le i \le N$, $1 \le k \le C$, $\sum_{k=1}^{C} U(i,k) = 1$, $\sum_{i=1}^{N} U(i,k) > 0$, $m_i \in [m_L, m_U]$ is the fuzzification coefficient ofand X_i and $D(i,k)^2 = \| X_i - V_k \|_2$ is the distance between two vectors X_i V_k.

To solve the optimization problem shown in Equation (5), we employ the Lagrange multiplier method. Let

$$L = \sum_{i=1}^{N}\sum_{k=1}^{C} U(i,k)^{m_i} D(i,k)^2 - \sum_{i=1}^{N} \lambda_i \left(\sum_{k=1}^{C} U(i,k) - 1 \right) \tag{6}$$

Furthermore, with

$$\frac{\partial L}{\partial U(i,k)} = 0, \quad 1 \le i \le N, 1 \le k \le C$$

$$\frac{\partial L}{\partial V_k} = 0, \quad 1 \le k \le C$$

we can compute V_k using

$$\left(\sum_{i=1}^{N} U(i,k)^{m_i} (X_i - V_k)^2 \right)$$

$$\frac{\partial L}{\partial V_k} = \sum_{i=1}^{N} U(i,k)^{m_i} \cdot 2 \cdot (X_i - V_k) \cdot (-1) = 0$$

hence, we have

$$\sum_{i=1}^{N} U(i,k)^{m_i} X_i = \sum_{i=1}^{N} U(i,k)^{m_i} V_k$$

or

$$V_k = \frac{\sum_{i=1}^{N} U(i,k)^{m_i} X_i}{\sum_{i=1}^{N} U(i,k)^{m_i}} \tag{7}$$

Next, we compute $U(i,k)$ using

$$\frac{\partial L}{\partial U(i,k)} = m_i U(i,k)^{m_i - 1} D(i,k)^2 - \lambda_i = 0$$

which implies

$$U(i,k) = \left(\frac{\lambda_i}{m_i D(i,k)^2}\right)^{\frac{1}{m_i - 1}} = \left(\frac{\lambda_i}{m_i}\right)^{\frac{1}{m_i - 1}} \left(\frac{1}{D(i,k)}\right)^{\frac{2}{m_i - 1}} \tag{8}$$

Because $\sum_{j=1}^{C} U(i,j) = 1$, we have $\sum_{j=1}^{C}\left(\frac{\lambda_i}{m_i D(i,j)^2}\right)^{\frac{1}{m_i-1}} = 1$,

or $\lambda_i^{\frac{1}{m_i-1}} = \frac{1}{\sum_{k=1}^{C}\left(\frac{1}{m_i D(i,k)^2}\right)^{\frac{1}{m_i-1}}}$

or

$$\lambda_i = \frac{1}{\left[\sum_{j=1}^{C}\left(\frac{1}{m_i D(i,j)^2}\right)^{\frac{1}{m_i-1}}\right]^{m_i-1}} = \frac{m_i}{\left[\sum_{j=1}^{C}\left(\frac{1}{D(i,j)^2}\right)^{\frac{1}{m_i-1}}\right]^{m_i-1}} \tag{9}$$

Replacing (9) into (8), we have

$$U(i,k) = \left(\frac{\lambda_i}{m_i D(i,k)^2}\right)^{\frac{1}{m_i-1}} = \frac{\left(\frac{1}{D(i,k)^2}\right)^{\frac{1}{m_i-1}}}{\sum_{j=1}^{C}\left(\frac{1}{D(i,j)^2}\right)^{\frac{1}{m_i-1}}} = \frac{1}{\sum_{j=1}^{C}\left(\frac{D(i,k)}{D(i,j)}\right)^{\frac{2}{m_i-1}}}$$

or $$U(i,k) = \frac{1}{\sum_{j=1}^{C}\left(\frac{D(i,k)}{D(i,j)}\right)^{\frac{2}{m_i-1}}} \qquad (10)$$

Summary of steps for the MC-FCM algorithm

Input: the dataset $X = \{X_1, X_2, ..., X_N\}$, the fuzzification coefficients m_i of each X_i.

Output: the partition of X into C clusters.

Steps:

- Step 1: Initialize value for V, let $l = 0$, set $\varepsilon > 0$.
- Step 2: At the l th loop, update U according to Equation (10).
- Step 3: Update V for the next step $(l + 1)$, according to Equation (7).
- Step 4: If $\left\| V^{(l)} - V^{(l-1)} \right\| < \varepsilon$, then go to Step 5; otherwise, let $l = l + 1$ and return to Step 2.
- Step 5: End.

The MC-FCM algorithm has similar steps to the FCM, except for Equations (10), (7) replacing(2), (3) in FCM.

4. **Evaluation of the Proposed MC-FCM Algorithm**

In order to evaluate the performance of the proposed MC-FCM algorithm, we tested it usingseveral UCI-benchmark datasets [13], described in Table 1 and compared it with FCM and FCMT2I.

The algorithm was built and tested using C#. To make our comparison objective, we adopted thefollowing performance indices outlined in [14]:

- The Davies–Bouldin (DB) index is based on a ratio involving within-group and between-group distances. $DB = \frac{1}{C}\sum_{j=1}^{C} D_j$, where $D_j = \max_{k \neq j} D_{j,k}$, $D_{j,k}$ is the within-to-between cluster spread for the j-th and k–th clusters, i.e., $D_{j,k} = \left(\bar{d}_j + \bar{d}_k\right) / d_{j,k}$, where $\bar{d}_j$ and $\bar{d}_k$ are the average within-group distances for the j-th and k–th clusters, respectively and $d_{j,k}$ is the inter-group distance between these clusters. These distances are defined as $\bar{d}_j = (1/N_j)\sum_{X_i \in C_j} \left\| X_i - \bar{X}_j \right\|$ and $d_{j,k} = \left\| \bar{X}_j - \bar{X}_k \right\|$. Here, D_j represents the worst-case within-to-between cluster spread involving the j-th cluster. Minimizing D_j for all clusters minimizes the DB index. Hence, good partitions, which are comprised of compact and separated clusters, are distinguished by small values of DB.
- The Alternative Silhouette Width Criterion (ASWC) is the ratio between the inter-group distance and the intra-group distance. $ASWC = \frac{1}{N}\sum_{i=1}^{N} S_{X_i}$, where $S_{X_i} = \frac{b_{k,i}}{a_{k,i} + \varepsilon}$. Let us consider that the i-th element of the dataset, X_i belongs to a given cluster $k \in \{1,2,...,C\}$, then $a_{k,i}$ is the average distance of X_i to all other elements in this cluster, $d_{q,i}$ is the average distance of X_i to all elements in another cluster q, with $q \neq k$, $b_{k,i}$ is the minimum of $d_{q,i}$ computed over $q \in 1,2,...,C,$ with $q \neq k$ and ε is a small constant (e.g., 10^{-6} for normalized data) used to avoid division by zero when $a_{k,i} = 0$. Large ASWC values indicate good partitions.
- The PBM index is also based on the within-group and between-group distances.

$PBM = \left(\frac{1}{C} \frac{E_1}{E_C} D_C \right)^2$, where $E_1 = \sum_{i=1}^{N} \left\| X_i - \overline{X} \right\|$ denotes the sum of distances between the elements and the grand mean of the data, $E_C = \sum_{j=1}^{C} \sum_{X_i \in C_j} \left\| X_i - \overline{X}_j \right\|$ represents the sum of within-group distances and $D_C = \max_{j,k=1,\ldots,C} \left\| \overline{X}_j - \overline{X}_k \right\|$ is the maximum distance between group centroids. The best partition is indicated when PBM is maximized.

- The Rand index (RI) can be seen as an absolute criterion that allows the use of properly labeled datasets for performance assessment of clustering results. This simple and intuitive index handles two hard partition matrices (R and Q) of the same dataset. The reference partition, R, encodes the class labels, while the partition Q partitions the data into C clusters and is the one to be evaluated. We have $RI = \frac{f_{11} + f_{00}}{N(N-1)/2}$, where f_{11} denotes the number of pairs of data elements belonging to the same class in R and to the same cluster in Q, f_{00} denotes the number of pairs of data elements belonging to different classes in R and to different clusters in Q. Large RI values indicate compatible clustering with the given class labels.
- Mean accuracy (MA), $MA = \min_{j=1,\ldots,C} \left(\frac{a_j}{b_j} \right)$, where a_j is the number of elements in the cluster j after clustering and b_j is the actual number of elements in cluster j. Large MA values often indicate good clustering.

The DB, ASWC and PBM indices show the compact and separated level of the clusters, while the MA and RI indices measure the quality of the clusters for the labeled datasets, to see the compatibility between the clusters and the labeled groups. When performing the evaluation in thisstudy, we focused on the MA and DB indices. Since the clustering algorithm is affected by the initialstep, each run with the parameters set was conducted several times.

Table 1. Summarized descriptions of the experimental datasets.

Dataset	**Samples**	**Attributes**	**Classes**	**Description**
ECOLI	336	7	8	This dataset consists of 7 characteristics of 8 *E. coli* bacteria types used to identify them.
HEART	303	13	2	This dataset consists of 13 symptoms used to determine if one has heart disease.
WDBC	569	32	2	This dataset consists of 32 metrics obtained from X-ray images of breast cancer tumors used todetermine if one has breast cancer.
IRIS	150	4	3	This dataset consists of 4 characteristics of 3 types of irises used to identify them.
WINE	178	13	3	This dataset consists of 13 chemical constituents in 3 types of Italian wine used to identify them.

We considered these scenarios:

(i) Perform FCM with *m* □ 2 several times and record the run with the best MA index result;

(ii) Perform FCM with changing m and record the run with the best MA index result;

(iii) Perform MC-FCM with changingresult; m_L, m_U and and record the run with the best MA index

(iv) Perform MC-FCM with the same m_L and m_U as in (iii), adjust and record the run with the best DB index result;

(v) Perform FCMT2I several times with the same m_L and m_U as in (iii) and record the run with the best MA index result.

The comparison results of various clustering algorithms on the five UCI-benchmark datasets areshown below.

Table 2. Experimental results with the ECOLI dataset, 336 samples, 8 clusters, m_L □ 5.5, m_U □ 6.5.

Algorithms	**DB**	**ASWC**	**PBM**	**RI**	**MA**
FCM, *m* □ 2	2.5855	0.8183	**0.0098**	0.8403	0.7652
FCM, *m* □ 6.1	2.8955	0.8657	0.0091	0.8604	0.8077
MC-FCM, □ □ 0.1	2.8329	**0.8995**	0.0084	**0.8699**	**0.8244**
MC-FCM, □ □ 1.9	**2.4021**	0.895	0.0089	0.8644	0.8125

FCMT2I	3.4561	0.8581	0.0091	0.8546	0.8184

Table 3. Experimental results with the HEART dataset, 303 samples, 2 clusters, $m_L = 1.1$, $m_U = 4.1$.

Algorithms	**DB**	**ASWC**	**PBM**	**RI**	**MA**
FCM, $m = 2$	0.7445	**0.8182**	0.8118	0.5154	0.5926
FCM, $m = 3$	0.9044	0.8159	0.8102	0.5213	0.6074
MC-FCM, □ = 0.8	0.7319	0.8140	0.8124	**0.5229**	**0.6148**
MC-FCM, □ = 1.7	**0.7306**	0.8159	0.8102	0.5213	0.6074

FCMT2I	0.7684	0.8166	**0.8186**	0.5168	0.5963

Table 4. Experimental results with the WDBC dataset, 569 samples, 2 clusters, m_L □ 3.1, m_U □ 9.1.

Algorithms	DB	ASWC	PBM	RI	MA
FCM, m □ 2	1.2348	**2.2109**	**23.036**	0.7504	0.8541
FCM, m □ 6	1.0618	2.0409	22.566	0.7707	0.8682
MC-FCM, □ □ 0.7	0.6508	1.588	20.147	**0.8365**	**0.9104**
MC-FCM, □ □ 0.4	**0.6298**	1.4938	19.897	0.8216	0.9051
FCMT2I	0.7847	1.588	20.147	0.8365	0.9104

Table 5. Experimental results with the IRIS dataset, 150 samples, 3 clusters, m_L □ 1.1 , m_U □ 9.1.

Algorithms	DB	ASWC	PBM	RI	MA
FCM, m □ 2	3.4835	**1.7587**	**0.1574**	0.8797	0.8933
FCM, m □ 9	2.0737	1.6771	0.1498	0.9124	0.9267
MC-FCM, □ □ 2.5	2.1388	1.6824	0.1471	**0.9195**	**0.9333**
MC-FCM, □ □ 9.9	2.0714	1.6794	0.1489	0.8797	0.92
FCMT2I	**1.3406**	1.7548	0.1377	0.8464	0.8533

Table 6. Experimental results with the WINE dataset, 178 samples, 3 clusters, m_L □ 1.1 , m_U □ 6.1 .

Algorithms	DB	ASWC	PBM	RI	MA
FCM, m □ 2	2.6983	1.2521	2.3675	0.7105	0.6854
FCM, m □ 10	2.2146	1.3040	**2.3711**	0.7204	0.7079
MC-FCM, □ □ 0.7	3.7023	1.2867	2.2668	**0.7363**	**0.7303**
MC-FCM, □ □ 8.7	**1.2995**	1.3131	2.3649	0.7187	0.7022
FCMT2I	2.0272	**1.3308**	2.2763	0.7254	0.6910

The results of the 5 indices for each algorithm implemented in 5 different datasets are shown inTables 2–6, with the best index results bolded in the tables. The results show that the proposed MC-FCM algorithm gives better results for most of the indices in all five datasets. The MA index resultsusing MC-FCM are consistently the best when compared to FCM and FCMT2I. For the DB index, MC-FCM gives the significantly better results in four out of five datasets. Other indices are similar for all cases. Regarding the number of iterations in the clustering process using MC-FCM, after averaging multiple runs, the WINE and IRIS datasets have a fewer number of iterations than FCM and a similar number of iterations compared to FCMT2I. The HEART dataset has similar numbers ofiterations compared to both FCM and FCMT2I, while the WDBC and ECOLI datasets have a similarnumber of iterations compared to FCM and fewer than

FCMT2I. Therefore, overall, the proposed MC-FCM algorithm outperformed the FCM and FCMT2I algorithms in terms of both clustering accuracy and clustering efficiency.

The MC-FCM algorithm can be improved even further by determining the parameter m_i for the elements to replace the algorithm in Section 3.1. The value m_i represents the fuzzy parameter, which corresponds to the type-2 fuzzy set membership representation of the element X_i in the clusters. The next step for this research is to apply hedge algebraic type-2 fuzzy sets [15–17] to determine the parameter m_i.

5. Conclusions

This study proposed a novel clustering algorithm FCM using multiple fuzzification coefficients, corresponding to each element of the dataset. The formulas for calculating $U(i,k)$ and V_k were

derived to ensure algorithm convergence. The experimental results using several UCI-benchmark datasets demonstrated that the proposed MC-FCM algorithm gave better results in terms of accuracy and efficiency compared to the standard FCM and FCMT2I algorithms. Calculations of fuzzificationcoefficients for each element in the preprocessing steps before clustering were based on its distanceto its neighboring elements. The content of this study can be further expanded to find more appropriate exponential parameters by applying different kinds of type-2 fuzzy sets.

Contributions: Conceptualization, T.D.K.; methodology, T.D.K., N.D.V. and M.-K.T.; software, N.D.V.; validation, T.D.K., N.D.V., M.-K.T. and M.F.; formal analysis, N.D.V.; writing—original draft preparation, T.D.K. and M.-K.T.; writing—review and editing, M.F.; supervision, T.D.K. and M.F. All authors have read and agreedto the published version of the manuscript.

References

1. Everitt, B.S.; Landau, S.; Leese, M.; Stahl, D. *Cluster Analysis*, 5th ed.; John Wiley & Sons, Ltd.: Hoboken, NJ, USA, 2011.
2. Bezdek, J.C.; Ehrlich, R.; Full, W. FCM: The fuzzy c-mean clustering algorithm. *Comput. Geosci.* **1984**, *10*,191–203.
3. Ruspini, E.H.; Bezdek, J.C.; Keller, J.M. Fuzzy Clustering: A Historical Perspective. *IEEE Comput. Intell. Mag.* **2019**, *14*, 45–55.
4. Gosain, A.; Dahiya, S. Performance Analysis of Various Fuzzy Clustering Algorithms: A Review. *ProcediaComput. Sci.* **2016**, *79*, 100–111.
5. Arora, J.; Khatter, K.; Tushir, M. Fuzzy c-Means Clustering Strategies: A Review of Distance Measures. *Softw. Eng.* **2018**, *731*, 153–162.
6. Hwang, C.; Rhee, F.C.-H. Uncertain Fuzzy Clustering: Interval Type-2 Fuzzy Approach to C-Means. *IEEETrans. Fuzzy Syst.* **2007**, *15*, 107–120.
7. Ji, Z.; Xia, Y.; Sun, Q.; Cao, G. Interval-valued possibilistic fuzzy C-means clustering algorithm. *Fuzzy SetsSyst.* **2014**, *253*, 138–156.
8. Linda, O.; Manic, M. General Type-2 Fuzzy C-Means Algorithm for Uncertain Fuzzy Clustering. *IEEE Trans. Fuzzy Syst.* **2012**, *20*, 883–897.
9. Pagola, M.; Jurio, A.; Barrenechea, E.; Fernández, J.; Bustince, H. Interval-valued fuzzy clustering. In Proceedings of the 16th World Congress of the International Fuzzy Systems Association (IFSA) and 9th Conference of the European Society for Fuzzy Logic and Technology (EUSFLAT), Paris, France, 30 June–3 July 2015; pp. 1288–1294.
10. Wu D.; Mendel, J.M. Enhanced Karnik-Mendel Algorithms for Interval Type-2 Fuzzy Sets and Systems. InProceedings of the NAFIPS '07, Annual Meeting of the North American Fuzzy Information Processing Society, San Diego, CA, USA,24–27 June 2007; pp. 184–189.
11. Du, M.; Ding, S.; Xue, Y. A robust density peaks clustering algorithm using fuzzy neighborhood. *Int. J. Mach. Learn. Cyber.* **2017**, *9*, 1131–1140.
12. Trabelsi, M.; Frigui, H. Robust fuzzy clustering for multiple instance regression. *Pattern Recognit.* **2019**, *90*, 424–435.

13. Bache, K.; Lichman, M. UCI Machine Learning Repository. Irvine, CA, USA: Univ. California, School of Information and Computer Science, 2013. Available online: http://archive.ics.uci.edu/ml (accessed on 19 September 2019).
14. Vendramin, L.; Campello R.J.G.B.; Hruschka, E.R. Relative Clustering Validity Criteria: A Comparative Overview. *Stat. Anal. Data Min.* **2010**, *3*, 209–235.
15. Nguyen, C.H.; Tran, D.K.; Nam, H.V.; Nguyen, H.C. Hedge Algebras, Linguistic-Valued Logic and Their Application to Fuzzy Reasoning. *Int. J. Uncertain. Fuzziness Knowl. Based Syst.* **1999**, *7*, 347–361.
16. Anh Phong, P.; Dinh Khang, T.; Khac Dong, D. A fuzzy rule-based classification system using Hedge Algebraic Type-2 Fuzzy Sets. In Proceedings of the Annual Conference of the North American Fuzzy Information Processing Society (NAFIPS), El Paso, TX, USA, 31 October–4 November 2016, pp. 265–270.
17. Khang, T.D.; Phong, P.A.; Dong, D.K.; Trang, C.M. Hedge Algebraic Type-2 Fuzzy Sets. In Proceedings ofConference: FUZZ-IEEE 2010, IEEE International Conference on Fuzzy Systems, Barcelona, Spain, 18–23 July 2010; pp. 1850–1857.