A HYBRID FUZZY C-MEDOIDS CLUSTERING USING THE WHALE OPTIMIZATION ALGORITHM

Anh Cuong Nguyen[1], *Thanh Long Ngo*[2], *Dinh Sinh Mai*[2], *The Long Pham*[2]

Abstract

Although the fuzzy c-means algorithm (FCM) has been widely used in many fields, they are sensitive to noise and outliers. Recently, the Fuzzy C-Medoids (FCMdd) algorithm has been shown to be more effective in dealing with noise data. The difference between FCM and FCMdd is the formation mechanism of clusters. At the same time, FCM builds clusters based on membership function and samples in the cluster. FCMdd selects some of the existing actual samples as cluster medoids. This results in FCMdd being able to handle noise better than FCM. This paper presents a hybrid approach of the whale optimization algorithm WOA with FCMdd (FWCMdd) to optimize the clustering process. This hybridization prevents FWCMdd from falling into the local trap and rapidly converging. This solution has been compared with the Fuzzy k-Medoids (FKM) algorithm and the primitive FCMdd. The results indicate that the proposed method is better than most of the evaluation indicators.

Index Terms

Fuzzy C-Medoids, Whale Optimization Algorithm, Clustering, Penalty, Bias.

1. Introduction

Data clustering is an unsupervised learning technique which used in many fields like data mining, image processing, computer vision, geo-informatics, etc. The most common clustering techniques are hierarchy and partitioning. The hierarchical methods create a series of nested partitions of the input data. In contrast, the partitioning methods often optimize a fitness function.

In the clustering approach, each data point of the dataset must be assigned exactly one cluster. After the fuzzy set theory came into being, clustering allowed a data point to belong to more than one cluster [1]. Thus, partition clustering is divided into two methods: crisp and fuzzy [2]. However, fuzzy clustering is based on the idea that each tiny piece of each member object (membership function of $[0, 1]$) is in a particular cluster and is considered to be the best method for capturing the uncertainty of the actual data [3]. The total value of the membership function of a given object on all clusters is

[1]Air Defence - Air Force Academy, [2]Le Quy Don Technical University

always equal to 1. In partition clustering, there appear many clustering families, in which the 2 clusters have the most research: These are cluster by mean [4] (average point) such as k-Means (KM), fuzzy c-Means (FCM) [5], and the other according to medoid (representative point) such as fuzzy k-Medoids (FKM) [6], fuzzy c-Medoids (FCMdd) [7].

Each cluster has a representative point (medoid point) in the medoid-cluster family, which acts as the cluster's centroid. During the clustering process, it is swapping another data point as a medoid to find the minimum value of the objective function. Simultaneously, a better assessment of cluster quality is always considered and studied in many ways. In [8], Yang et al. stated that the traditional FCM algorithm's objective function does not assume any spatial information. Hence, clustering only involves the gray levels independent of the pixels of the image in the segment. This limitation makes FCM very sensitive to noise.

Mei et al. [9] proposed a fuzzy clustering problem around medoids and present it through a unified view, which mentions that there may be more than one medoid in each cluster (FMMdd). In FKM, the membership function value allows an object's assignment to multiple clusters according to different measurement methods. Both FKM and FMMdd offer formulations suitable for many contexts. However, they are non-convex optimization problems. Their association algorithms can lead to many local minima of unknown quality (There is a difference in noise added to the noise indicated by [10]).

Furthermore, FKM and FMMdd are usually sensitive to centroid initialization. Convex fuzzy k-Medoids (CFKM) [10] improves with the assumption that subjects must be divided entirely by one and only one medoid, for which the medoid must be wholly assigned to one and only one cluster. This is like the preliminary clustering of objects to clusters. This will reduce the random substitution of medoid, which at each cluster, the source of replacement of medoid is limited to only preliminary clustered subjects. However, CFKM requires more increased time costs for clustering to be found.

Shihong Yue et al. [11] reformatted based on two reward and penalty functions for the original FCM algorithm's objective function. The reward function is defined as the original data in a given data set. Still, the penalty function is characterized by an additional collection of data about the initial data distribution. These other data are computed around each group of aggregated original data. Their effect extends the objective function's values to limit the specified cluster centroids' tendency to reach these data points. However, building and establishing suitable coefficients in the reward and penalty mechanism is necessary when many data sets are inconsistent in distributed clusters.

The FCMdd algorithm is shown to achieve better clustering results than FCM on noisy data sets in most tests. They have the disadvantage of being prone to local extremes because the selection process for medoids is from data samples [12]. FCMdd is constructed the same way as FCM except that it constructs medoids (one of the actual objects in sample space) that are not centroids (a point in continuity space). In FCMdd, with the current membership, each medoid is decided using a heuristic algorithm to select the object x_q as the medoid of cluster c. Finding medoids in FCMdd can make FCMdd run faster in real-time. However, the consequence of a simplified heuristic search has a trade-off inefficiency. FCMdd is prone to get stuck at a local minimum.

This paper proposes a hybrid algorithm (FWCMdd) between FCMdd and whale optimization algorithm (WOA) [13] to optimize clustering results. This paper's proposal to use WOA to support FCMdd is to develop a better solution through each loop to select the appropriate medoid. Therefore, to take advantage of FCMdd, the objective of this paper is to minimize the possibility of falling to the local extremes of the FCMdd algorithm.

The paper is organized as follows: Section II shows related basics: summarize some basic knowledge about fuzzy clustering FCMdd, introducing the WOA whale optimization algorithm. Section III proposes a method to hybridize WOA and FCMdd. Section IV experimental results. Section V ends with some conclusions and references.

2. Preliminaries

2.1. Fuzzy C-medoids algorithm (FCMdd)

Fuzzy C-medoids clustering algorithm (FCMdd) [7] divides data into clusters, with medoid points representing the centroids of the clusters. According to [14], the use of medoid can reduce the effect of interference. Let a set $X = \{x_1, x_2, \ldots, x_N\}$ of N objects, for each object has M properties . Let $d_{ij} = d\left(x_i, x_j\right)$ describe the difference between the object x_i and the object x_j. Let $Z = \{z_1, z_2, \ldots, z_K\}, z_i \in X$ describe a subset of x with K parts. Let X^K represent the set of all K subsets Z of X. FCMdd minimizes objective function:

$$J_m\left(Z, X\right) = \sum_{j=1}^{N} \sum_{i=1}^{K} u_{ij}^{m} d\left(x_j, z_i\right) \tag{1}$$

where u_{ij} is membership value of the j^{th} object in the i^{th} cluster. m is a fuzzier belong to greater than or equal 1. Medoid specificity problems are often used in practice when it is necessary to specify data objects available to play a certain role. The FCMdd algorithm is then improved by the authors themselves in the initialization steps. Instead of randomly taking K data points as initial medoids, only the first medoid z_1^0 is randomly taken. From the 2^{nd} medoid z_1^0 on, is taken on the principle that is furthest away from the previous medoid point and does not coincide with the existing medoid point. This can help reduce computational complexity.

Compared with the famous FCM clustering also partitioning a set of N data points $x_j, j = 1, \ldots, N$ into K fuzzy clusters with centroids by cluster's distance averaging. According to [2], FCM clustering can effectively increase speed even when applied to multidimensional datasets. The advantages of FCM are ease of deployment, the applicability of multidimensional data, and the ability to model uncertainty in data [15]. The FCM algorithm's time complexity is $O\left(N\right)$, and FCMdd algorithm's one is $O\left(N^2\right)$ [7]. Membership functions can be defined through distances, commonly used according to FCM to describe the following:

$$u_{ij} = \left(\frac{1}{d\left(x_j, z_i\right)}\right)^{1/(m-1)} \Bigg/ \sum\nolimits_{k=1}^{K} \left(\frac{1}{d\left(x_j, z_k\right)}\right)^{1/(m-1)} \tag{2}$$

where m is a fuzzier ($m > 1$, usual get 2). The Equation 2 means that the total membership value of an object x_i across all clusters equals 1. The FCMdd's pseudos code is shown in Algorithm 1.

```
Algorithm 1: Fuzzy C-medoids algorithm
Input: $X = \{x_1, x_2, \ldots, x_N\}$, $K$ clusters, $t = 0$, $T_{max} = 1000$, $J\_\{best\} = \infty$.
Output: Matrix $U$, set of medoids $Z$, objective function $J$.
BEGIN
    1 Initialized $K$ different medoids from $X$: $Z = \{z_1, z_2, \ldots, z_K\}, z_i \in X$.
    2 REPEAT
        2.1 Compute membership function values $U = u_{ij}$ by Equation 2.
        2.2 FOR $i = 1, 2, ..., K$ AND $j = 1, 2, ..., N$:
            a) Save the current medoids $Z^{old} = Z$.
            b) Calculate new medoid $z_i$ with $i = 1, 2, ..., K$:
                $q = \underset{1 \le k \le N}{\arg\min} \sum_{j=1}^{N} u_{ij}^{m} d\left(x_k, x_j\right)$
                $z_i = x_q$
            c) Computing $J$ and $U$: IF $J_{best} > J$ THEN save $Z, U, J_{best} = J$.
        2.3 $t = t + 1$
    3 UNTIL $Z^{old} = Zort = T_{max}$
END.
```

2.2. Whale optimization algorithm

Whale optimization algorithm (WOA) [13] is a herd optimization form inspired by the hunting of humpback whales. The prey (target) of the whale swimming across the water is surrounded by a shrinking funnel-shaped swirl of whales using a bubble strategy. WOA mathematically models prey encirclement, spiral maneuver, and prey search to get the optimal set of parameters for problems using numeric data. This hunt unifies the herd (whales always work together). At a moment, one whale $X\left(t\right)$ position is considered as a reference basis for the position of the remaining whales. If the expected prey position

$X\,(t+1)$ is not within the control radius of $X\,(t)$, then find another random prey position (discovery). On the contrary, the prey is surrounded (exploitation). Updated a new position of whale $X\,(t)$ close to prey position (depends on the displacement vector). The remaining whales have their position updated according to the displacement vector.

2.2.1. The siege of prey (Exploration): In the WOA, the optimal position at-next-loop of the current whale is $X*(t)$ (prey position), the position of the remaining whales was considered optimal. In the next loop, all whales update their positions according to prey position $X*(t)$.

$$\overrightarrow{D} = [\overrightarrow{C} * \overrightarrow{X^*}\,(t)] - \overrightarrow{X}\,(t) \tag{3}$$

$$\vec{X}(t+1) = \vec{X}^*(t) - [\vec{A} * \vec{D}] \tag{4}$$

where $\vec{D}$ is the displacement vector, t is the number of iterations up to now, $\overrightarrow{X}$ is the position vector of each whale, $\vec{X}^*$ is the position vector of the prey in the t^{th} loop. The WOA states how to compute factor vectors $\overrightarrow{A}$ and $\overrightarrow{C}$ as follows:

$$\overrightarrow{A} = 2 * [\overrightarrow{a} * \overrightarrow{r}_1] - \overrightarrow{a} \tag{5}$$

$$\overrightarrow{C} = 2 * \overrightarrow{r}_2 \tag{6}$$

where $\overrightarrow{r}_1, \overrightarrow{r}_2$ are two random vectors in $[0,1]$ so that any position in the search space can be reached by adjusting the values of the vectors $\overrightarrow{A}$ and $\overrightarrow{C}$. $\overrightarrow{a}$ is reduced linearly from 2 to 0 during the iteration, describing the shrinking radius of the spiral orbit.

2.2.2. Bubble siege strategy: The bubble siege strategy is a combination of two narrow enclosure approaches and the helical helical position modeled mathematically. The shrinking encircling mechanism reduce the distance the whale and the prey by $\vec{a}$ of Equation 6, followed random $\overrightarrow{A}$ in $[-a,a]$. When any value $\overrightarrow{A}$ in $[-1,1]$, the new whale's position can belong to the range of the whale and the prey. Equation 8 created a spiral trajectory between the current position of the whale and its prey; in a loop, that whale's new position is in that orbit: created a spiral line between the current whale's position and the his prey to mimic the whale's movement, at a moment, his new position is on it:

$$\vec{D}' = \vec{X}^*(t) - \vec{X}(t) \tag{7}$$

$$\vec{X}(t+1) = \vec{D}' * e^{bl} * cos(2\pi l) + \vec{X}^*(t) \tag{8}$$

in which $\vec{D}'$ is the distance between the whale and the prey, b is the constant for the logarithm shape, l is random in $[-1,1]$, denoting the elasticity of the spiral. WOA can choose to either miniature enclosure or spiral motion, simulated by probability p (be long to $[0,1]$).

$$\vec{X}(t+1) = \begin{cases} \vec{X}^*(t) - [\vec{A} * \vec{D}] \quad if \quad p < 0.5 \\ \vec{D}' * e^{bl} * cos(2\pi l) + \vec{X}^*(t) \quad if \quad p \geq 0.5 \end{cases} \tag{9}$$

2.2.3. Search for prey: WOA uses $\left|\vec{A}\right| > 1$ to force prey away from the current focus whale. At that time, the whale will not follow the current prey but search and update the position according to other random alternate prey.

$$\overrightarrow{D} = [\overrightarrow{C} * \overrightarrow{X}_{rand}] - \overrightarrow{X} \tag{10}$$

$$\overrightarrow{X}(t+1) = \overrightarrow{X}_{rand} - [\overrightarrow{A} * \overrightarrow{D}] \tag{11}$$

where $\overrightarrow{X}_{rand}$ is a random position vector. Each iteration, the whales get updated their position from Equation 11. The WOA's pseudos code is presented in Algorithm 2.

```
Algorithm 2: The Whale Optimization Algorithm
Input: $X = \{x_1, x_2, \ldots, x_N\}$, $K$ clusters, $t = 0$, $T_{max} = 1000$.
Output: $X^*$ is the best search agents.
1. Initializing the whale population $X_i, i = 1...n$ is the solution to look for.
2. Calculate the value of the fitness function:
2.1 WHILE $t < T_{max}$
        FOR EACH the current Whale $i$:
            IF ($p < 0.5$)
                IF ($|A| < 1$)
                    Updates position of the current whale by Equation 5.
                ELSE
                    Randomly select a prey $X_{rand}$.
                    Updated position of the whales by Equation 11.
                Updated position of the whales by Equation 8.
        Check if any prey gets out of the search space and improve it.
        Calculate the fitness function value.
        Update $X^*$ if there is better solution.
        $t = t + 1$.
2.2 END WHILE
```

3. Proposal to hybrid WOA with FCMdd

Based on the presented content, we proposed an improved solution for FCMdd to optimize clustering results with objective function as follow:

$$J = \sum_{i=1}^{K} \sum_{j=1}^{N} u_{ij}^{*} d\left(z_i, x_j\right) \tag{12}$$

According to medoid-based clustering methods, it is necessary to alternate data points as medoid so that J reaches the minimum value. The selection of z_i medoid will determine the appropriate $U^{*} = \left\{u_{ij}^{*}\right\}$ matrix for better results. Here $\left\{u_{ij}^{*}\right\}$ need to apply the new calculation according to the location of the selected medoid. We use existing FCMdd and FCM clustering formulas to find the new formula for $\left\{u_{ij}^{*}\right\}$ according to the proposed algorithm.

3.1. Propose on how to initiate and search for medoids

Using FCM to get a membership functional matrix $\mu_{FCM} = \{\mu_{ij}\}$, in each cluster, select initial medoid z_i as the points with the highest of μ_{ij}. In the original FCMdd algorithm, the process of finding an alternative medoid closest to the average centroid should satisfy Equation 3) makes the complexity difficult to improve. In essence, the procedure for finding the medoid is randomized and stores the best values so reducing the objective function will take a long time to converge. We propose to use WOA to better step-by-step identification of medoid search. When keeping the centroid of the other clusters fixed, WOA gives the optimal position v_i^{*} as the new centroid for the current cluster, and v_i^{*} is the base to start choosing the centroid of other clusters. Selecting the centroids of the clusters as the major ones, the adjustment as above will gradually get the best set of centroids.

How to perform, use the loop to look at each cluster in turn. At the current cluster i, consider medoid z_i, and at the same time keep $(K-1)$ medoids z_k in other clusters (temporarily consider the best z_k, fixed to focus on finding new z_i). WOA needs to indicate that the best new alternative to z_i is v_i^{*}. If the WOA cannot find any v_i^{*}, then randomly select any v_i^{*}. On the contrary, (if found), WOA indicates an optimal location v_i^{*} to replace z_i according to Equation 5, for the convenience of observation, it can be rewritten to:

$$\vec{v_i^{*}} = \vec{z_i} - [\vec{A} * \vec{D}] \tag{13}$$

where A and D are WOA parameters explained in Equation 5.

Algorithm 3: The FWCMdd Algorithm

Input: $X = \{x_1, x_2, \ldots, x_N\}$, K is the number of pre-clusters, $t = 0$, $T_{max} = 1000$, $J\_\{best\} = \infty$.

Output: The membership function values U^*; The id of the medoids z; The fitness function value J_m.

```
1. Load database $X$, initialize the variables: $U, K$.
2. Select $K$ different points $z_i$ as medoid form $X$, using FCM to get set $Z$:
      $Z = \{\mathrm{z}_t\}_{t=1}^{K} | \mathrm{z}_t = \mathrm{x}_k$ with $u_{ik} = max\{\mathrm{u_{ij}}\}_{j=1}^{N}$
3. WHILE $(t < T_{max})$
      3.1 FOR EACH cluster $i$ DO:
            a. Calculate distances $D^{(t)} = d(z_i, x_j)$, $i = 1, ..., K; j = 1, ..., N$.
            b. FOR EACH $z_i$ DO (Using Equation of WOA)
                  + Update $a, A, C, l, p$ (of WOA).
                  + IF $(p < 0.5)$ THEN
                        IF $(|A| < 1)$ THEN Update $z_i$ by Equation 21.
                        ELSE
                              Random $z_i$.
                              Update $z_k (k \neq i)$ by Equation 23.
                  + ELSE
                        Update $z_k (k \neq i)$ by Equation 23, with $\bar{D} = |v_i^* - z_i|$
                        $v_k^* = \bar{D} * e^{bl} * cos(2\pi l) + z_k$
            Calculate the matrix $U^*$ according **Algorithm 4 (PEP)**.
            Calculate the fitness function value $J_m$ by Equation 12.
            IF $(J < J_{best})$ THEN $J_{best} = J$. save $J_{best}, U^*, z$.
      3.2 $t + +$
4. END WHILE
```

3.2. Propose on how to correct the bias

Since the clustering-medoid family requires a data point as a cluster centroid (representative point), it is necessary to choose a data point to replace position v_i^* (as indicated by WOA). We propose selecting the candidate point closest to v_i^*. Because this selection is biased, it is necessary to optimize the expected value of the target function, or penalty for the chosen position. From here it can be assumed that the value of the objective function J_v calculated by the set $V = \{v_i\}$ that WOA indicates is the expected value, and that the set $Z = \{z_i\}$ to compute J_z is the real value. J_z is as close to the expected value J_v as possible, so in the calculation of other parameters, these two values are said to be equal. To calculate $U^* = \left\{u_{ij}^*\right\}$ in terms of $U = \{u_{ij}\}$, we give $J_v = J_z$ and subdivide these two objective functions into clusters i.

$$J = J_v = \sum\nolimits_{i=1}^{K} J_{vi}; J = J_z = \sum\nolimits_{i=1}^{K} J_{zi} \tag{14}$$

To evaluate the role of each cluster, calculate the weight of each cluster k:

$$W_{vk} = \frac{J_{vk}}{J_v}; W_{zk} = \frac{J_{zk}}{J_z}; \forall k = 1 \ldots K \tag{15}$$

where W_{vk}, W_{zk} is the weight of the k^{th} cluster by two ways of calculating the objective function with the expected cluster center $V = \{v_i^*\}$, and with the actual center set $Z = \{z_i^*\}$. J_{vk}, J_{zk} are the value parts of J in the k^{th} cluster. Do the W_{vk}/W_{zk} division and get:

$$\lambda_k = \frac{W_{vk}}{W_{zk}} = \frac{J_{vk}}{J_v} \Big/ \frac{J_{zk}}{J_z} = \frac{J_{vk} \cdot J_z}{J_{zk} \cdot J_v}, \forall k = 1 \ldots K \tag{16}$$

Since the value $J_z = J_v$ we have:

$$\lambda_k = \frac{J_{vk}}{J_{zk}}, \forall k = 1 \ldots K \tag{17}$$

Expanding an i^{th} cluster:

$$\lambda_i = \frac{J_{vi}}{J_{zi}} = \frac{\sum_{j=1}^{N} \mu_{ij} d_{vij}}{\sum_{j=1}^{N} u_{ij} d_{zij}} with \begin{cases} d_{vij} = \|v_i - x_j\|, \\ d_{zij} = \|z_i - x_j\|. \end{cases} \tag{18}$$

Apply the distribution law of the addition obtained:

$$J_{vi} = J_{zi} \cdot \lambda_i = \lambda_i \sum_{j=1}^{N} u_{ij} d_{ij} = \sum_{j=1}^{N} (\lambda_i u_{ij}) d_{ij}, withd_{ij} = \|z_i - x_j\| \tag{19}$$

So from Equation 12 and Equation 19 deduced:

$$u_{ij}^* = \lambda_i u_{ij} \tag{20}$$

So the value of the matrix $U* = \left\{u_{ij}^*\right\}$ need to find is multiplying the coefficient λ_i by the value $U = \{u_{ij}\}$ of the standard KCMdd cluster. Naturally, $U* = \left\{u_{ij}^*\right\}$ still have to normalize for each object to comply with the constraint that the total value of the membership function of each point for all clusters must be equal to 1. Updating the z_i^{new} closest to v_i^* of WOA. Or is:

$$q = \arg \min_{x_j \in X} d\left(v_i^*, x_j\right); z_i^{new} = x_q \tag{21}$$

Apply WOA to update other v_k^* points according to Equation 5

$$\vec{v_k^*} = \vec{z}_k - [\vec{A} * \vec{D}] \tag{22}$$

Pick out the z_k^{new}:

$$q = \arg \min_{x_j \in X} d\left(v_k^*, x_j\right); z_k^{new} = x_q \tag{23}$$

Based on the above argument, we propose the FWCMdd method which is a hybrid between WOA and FCMdd, using Equation 12 as the value of fitness function. Where PEP (Position Expectation Penalty) is proposed as follows:

```
Algorithm 4: The Position Expectation Penalty (PEP) algorithm
Input: Dataset: $K$, the suggested centroids set: $V = \{v_i\}_{i=1}^{K}$
Output: Set of medoids: $Z = \{z_i\}_{i=1}^{K}, z_i = x_q \in X$, Set value matrix $U* = \left\{u_{ij}^{*}\right\}$.
    1 Get $X, V$.
    2 Determine the medoids $Z = \{z_i\}$ by Equation 21.
    3 Calculate the distance matrix $Dv_{ij} = \{d(v_i, x_j)\}$, $Dz_{ij} = \{d(z_i, x_j)\}$.
    4 Calculate the membership matrix $\mu_{ij}, u_{ij}$ by Equation 2.
    5 Calculate conversion coefficient matrix by cluster $\lambda = \{\lambda_k\}$ by Equation 17.
    6 Calculate $U^* = \left\{u_{ij}^{*}\right\}$ by Equation 20.
```

Computational complexity: Due to the proposed algorithm using 2 loops with N nested objects the worst probability and maximum is N cases. Using the rule of taking max and ignoring single lines of constant value is $O(1)$, so this complexity is reduced to $O(N^2)$.

4. Experiments

The proposal was experimented on several UCI data sets [16] with the number of instances, attributes and clusters in Table 1. In this study, the authors chose to test on UCI datasets due to the richness and diversity of data types. In practice, most data sets have noise due to data acquisition or data normalization. Moreover, to increase the reliability of the method, the authors have selected 10 datasets on different subjects for testing and used 10 different indexes to evaluate the clustering quality.

The tests in the article are installed on Matlab 2018 software, the computer: core i5, 16 Gb RAM, 1.7GHz. The input parameters are selected according to the suggestions from the original algorithms. The clustering results are presented in Table 2. The clustering evaluation indicators are recorded at the top of the column, if there is the symbol $(-)$, the smaller the value, the better. Similar to $(+)$, the better the value. The formula for calculating these indices is set according to [17], [18]. Better metrics are automatically highlighted.

The ratios are computed using the following formula:

- The objective function value is as small as possible, according to Equation 12. The partition entropy VPE index (a is the logarithmic radix, usually $a = 10$), measures the scalar of the fuzzy amount in a membership function, giving good clustering performance the smaller it is.

$$V_{PE} = -\left(1/N\right)\sum\nolimits_{i=1}^{K}\sum\nolimits_{j=1}^{N} u_{ij}\log_a\left(u_{ij}\right) \tag{24}$$

- The smaller Fukuyama – Sugeno Index value (VFS) considers compact and segregation, the better.

$$\begin{gathered} V_{FS} = J_m\left(U,V\right) - K_m\left(U,V\right) = J_m - \sum\nolimits_{j=1}^{N}\sum\nolimits_{i=1}^{K} u_{ij}^m \left\|v_i - \bar{x}\right\| \\ where \bar{x} = \left(1/N\right)\sum\nolimits_{j=1}^{N} x_j \end{gathered} \tag{25}$$

- The smaller Xie – Beni index (VXB) to measure the smaller the overall average tightness and distinctiveness, the better the partitioning results:

$$V_{XB} = J_m \Big/ \left(N \times \min\nolimits_{i,k=1,\ldots,K,i\neq k}\left\|v_i - v_k\right\|^2\right) \tag{26}$$

- The smaller Kwon index value, to eliminate the monotonous downtrend of VXB when the number of K clusters approaches the number of data points N, the better.

$$V_K = \frac{\sum\nolimits_{j=1}^{N}\sum\nolimits_{i=1}^{K} u_{ij}^2\left\|x_j - v_i\right\|^2 + \left(1/K\right)\sum\nolimits_{i=1}^{K}\left\|v_i - \bar{x}\right\|^2}{\min\nolimits_{i\neq k}\left\|v_i - v_k\right\|^2} \text{where} \bar{x} = \sum\nolimits_{j=1}^{N} x_j/N \tag{27}$$

- The smaller the VT index value, using penalty functions in both the numerator and the sample, eliminates the decreasing trend when $K \rightarrow N$, enhances the numerical stability when $m \rightarrow \infty$, the better.

$$V_T = \frac{\sum\limits_{i=1}^{K}\sum\limits_{j=1}^{N} u_{ij}^2\left\|x_j - v_i\right\|^2 + \frac{1}{K(K-1)}\sum\limits_{i=1}^{K}\sum\limits_{k=1;k\neq i}^{K}\left\|v_i - v_k\right\|^2}{\min\limits_{i\neq k}\left\|v_i - v_k\right\|^2 + 1/K} \tag{28}$$

The lower the value SEP of the isolation distance between fuzzy clusters is the better.

$$\begin{gathered} SEP = \frac{D_{\max}^2}{D_{\min}^2}\sum_{i=1}^{K}\left(\sum_{k=1}^{K}\left\|v_i - v_k\right\|^2\right)^{-1} \text{where} \\ D_{\min} = \min\nolimits_{i\neq k}\left\|v_i - v_k\right\|, D_{\max} = \max\nolimits_{i\neq k}\left\|v_i - v_k\right\| \end{gathered} \tag{29}$$

- NC (novel compression) improves the performance of the partition coefficient PC, the larger the value of NC, the higher the internal compression and the better the fuzzy clustering results.

$$NC = \sum_{j=1}^{N}\sum_{i=1}^{K}\frac{u_{ij}^2}{u_{\max}} where u_{\max} = \max_{1\leq j\leq N}\left\{\sum_{i=1}^{K} u_{ij}^2\right\} \tag{30}$$

The VPC index indicates the number of average member values taken between pairs of fuzzy subset, by combining into a single number, the larger value is the better.

$$V_{PC} = \left(1/N\right)\sum\nolimits_{i=1}^{K}\sum\nolimits_{j=1}^{N} u_{ij}^2 \tag{31}$$

- The greater its modified PC index (MPC), the better the partitioning results:

$$V_{MPC} = 1 - (K/(K-1))\,(1 - V_{PC}) \tag{32}$$

Set up parameters for the experiment: In the membership formula, the objective function, $MAX_ITER = 100$ (maximum loop index), $b = 1$, constant in WOA (Eq. 24).

The experimental results show that the smaller the Cost, VPE, VFS, VXB, VK, VT, Sep indexes, the better the cluster quality. The higher the NC, VPC and VMPC indexes, the better the cluster quality. The clustering results by indicators are shown in Table 2, indicating that the proposed method gave significantly better results in most test cases. There are 100 measure values, in which the algorithm proposed FWCMdd shows better values at 88/100 values, only 9/100 values are FKM algorithm for better results, and 3/100 standard FCMdd value gives a better result.

FWCMdd has 2/10 datasets giving better values in ten indicators; 5/10 datasets give better values at nine indicators; 2/10 data sets are better for the eight indicators, and 1/10 dataset give good values in the seven indicators.

Table 1. List of test datasets from UCI database

Name	Datasets	No. Instances	No. Atributes	No. Clusters
Data1	Auto MPG Data Set	398	8	3C
Data2	Breast Cancer Wisconsin	699	10	2C
Data3	Liver Disorders Data Set	345	7	2C
Data4	Contraceptive Method Choice	1473	9	3C
Data5	Glass Identification	214	10	6C
Data6	Hayes-Roth	132	5	3C
Data7	Iris Data Set	150	4	3C
Data8	Computer Hardware	209	9	7C
Data9	Wine	178	13	3C
Data10	Vehicle-xaa	94	18	4C

Table 2 shows that the proposed algorithm gives the best results in most of the indicators in all ten experimental datasets. With this result, the support of the WOA algorithm for FCMdd clustering helped find the optimal parameter sets to increase clustering results' accuracy. This study also showed the potential of WOA in assigning optimal points for mean family clustering algorithms and serving as a basis for calibrating the medoid family clustering algorithm.

The combination uses the WOA algorithm to correct the medoids so that they are closer to the actual cluster centers. Thereby limiting the risk of falling into local optimization and helping the algorithm to be stable and tend to converge faster. Therefore, the proposed algorithm gives better clustering results than the pre-improvement algorithm.

Table 2. Some results of cluster quality assessment by validity indexes

Data	Alg.	**Cost**	**VPE**	VFS	**VXB**	**VK**	**VT**	**Sep**	**NC**	**VPC**	**VMPC**
1	FKM	286.819	1.439	35.547	0.365	145.778	245.250	2.336	163.112	0.410	0.115
	FCMdd	228.891	1.363	**3.958**	0.908	363.349	151.155	2.214	177.002	0.445	0.167
	FWCMdd	**179.845**	**0.933**	66.557	**0.320**	**128.855**	**118.903**	**1.688**	**253.267**	**0.636**	**0.455**
2	FKM	537.871	0.741	**45.126**	0.142	99.726	537.384	0.469	465.401	0.666	0.332
	FCMdd	506.776	0.809	270.569	1.132	791.374	630.570	0.553	396.189	0.637	0.274
	FWCMdd	**404.998**	**0.400**	297.002	**0.087**	**60.774**	**420.203**	**0.211**	**589.982**	**0.844**	**0.688**
3	FKM	125.421	0.915	19.373	**0.476**	**164.566**	55.200	0.705	193.099	0.560	0.119
	FCMdd	104.985	0.988	22.033	15.675	540.803	644.435	0.897	176.245	0.511	0.022
	FWCMdd	**104.135**	**0.875**	**18.205**	1.201	414.565	**41.552**	**0.649**	**201.555**	**0.584**	**0.168**
4	FKM	164.405	1.496	329.479	1.905	280.884	198.004	2.518	550.492	0.374	0.061
	FCMdd	135.381	1.535	483.156	44.430	654.492	148.834	2.672	393.698	0.360	0.041
	FWCMdd	**130.657**	**1.290**	**216.338**	**0.400**	**58.905**	**140.958**	**2.107**	**701.584**	**0.476**	**0.214**
5	FKM	148.43	2.211	18.495	**0.469**	**105.172**	136.304	12.724	68.925	0.322	0.186
	FCMdd	90.916	2.411	**8.966**	33.869	728.167	67.432	12.249	53.928	0.252	0.102
	FWCMdd	**73.102**	**1.745**	11.106	1.220	269.104	**50.373**	**11.748**	**98.322**	**0.459**	**0.351**
6	FKM	111.958	1.515	13.541	**0.298**	**39.802**	105.998	2.514	50.897	0.386	0.078
	FCMdd	100.384	1.554	25.513	0.992	131.234	88.313	2.658	48.509	0.367	0.051
	FWCMdd	**96.299**	**1.377**	**13.265**	0.448	59.658	**86.063**	**2.233**	**59.094**	**0.448**	**0.172**
7	FKM	74.543	1.319	**9.126**	0.220	33.853	44.259	2.069	71.856	0.479	0.219
	FCMdd	71.254	1.481	15.950	3.115	46.746	48.725	2.405	59.952	0.400	0.100
	FWCMdd	**38.212**	**0.718**	22.566	**0.151**	**23.654**	**18.490**	**1.421**	**111.069**	**0.740**	**0.611**
8	FKM	125.617	2.335	33.890	218.982	492.729	102.767	17.88	69.413	0.332	0.221
	FCMdd	61.028	2.643	**4.578**	10.572	222.419	42.930	18.736	47.549	0.228	0.090
	FWCMdd	**49.922**	**1.955**	7.137	**1.158**	**252.126**	**28.979**	**16.959**	**87.507**	**0.419**	**0.322**
9	FKM	165.368	1.546	34.551	3.254	583.676	168.943	2.626	64.844	0.364	0.046
	FCMdd	139.879	1.542	29.468	0.848	151.567	122.419	2.597	65.599	0.369	0.053
	FWCMdd	**125.854**	**1.249**	**1.641**	**0.208**	**37.473**	**108.695**	**1.986**	**90.984**	**0.511**	**0.267**
10	FKM	100.281	1.912	**0.448**	0.858	84.427	135.346	5.057	30.696	0.327	0.102
	FCMdd	84.481	1.991	23.189	3.528	332.386	100.203	5.375	28.312	0.301	0.068
	FWCMdd	**69.982**	**1.599**	7.960	**0.668**	**64.790**	**75.766**	**4.383**	**42.345**	**0.450**	**0.267**

5. Conclusion

In this paper, the WOA is applied to find the optimal centroids for the FCMdd clustering problem. Using the WOA algorithm can significantly improve the clustering algorithm's accuracy, especially the medoid family cluster. The experimental results of clustering on ten different data sets show that the proposed algorithm gives significantly better results than the FKM and FCMdd algorithms. This result also shows the potential in combining clustering algorithms with optimization techniques that can help clustering algorithms work more efficiently.

In the future, we will study the combination of the WOA algorithm with clustering methods based on particle calculation to help improve computation time on large data sets, multidimensional data.

References

[1] S. C. C. W. L. Cai. and D. Q. Zhang, “Fast and robust fuzzy c-means clustering algorithms incorporating local information for image segmentation,” *Pattern Recognition*, vol. 40(3), pp. 825–838, 2007.

[2] H. Frigui and C. Hwang, “Fuzzy clustering and aggregation of relational data with instance-level constraints,” *IEEE Trans. Fuzzy Systems*, vol. 16(6), pp. 1565–1581, 2008.

[3] L. Hu and K. C. Chan, “Fuzzy clustering in a complex network based on content relevance and link structures,” *IEEE Trans. on Fuzzy Systems*, vol. 24(2), pp. 456–470, 2016.

[4] P. Pooja and V. Singh, “Comparison between standard k-mean clustering and improved k-mean clustering,” *Int Journal of Computer App.*, vol. 146, pp. 39–42, 2016.

[5] R. E. J. Bezdek and W. Full, “Fcm: The fuzzy c-means clustering algorithm,” *Computer and Geoscience*, vol. 10, no. 2-3, pp. 191–203, 1984.

[6] M. A. Akhras, “An efficient fuzzy k-medoids method,” *Journal of World Applied Sciences*, vol. 10, pp. 574–583, 2010.

[7] O. N. L. Y. R. Krishnapuram, A. Joshi, “Low-complexity fuzzy relational clustering algorithms for web mining,” *IEEE Trans. Fuzzy Systems*, vol. 9, p. 595–607, 2001.

[8] S. H. Y. Yang, “Image segmentation by fuzzy c-means clustering algorithm with a novel penalty term,” *Computing and Informatics*, vol. 26, p. 17–31, 2007.

[9] L. C. J.P. Mei, “Fuzzy relational clustering around medoids: a unified view,” *Fuzzy Sets and Systems*, vol. 183, p. 44–56, 2011.

[10] S. B. D.N. Pinheiro, D. Aloise, “Convex fuzzy k-medoids clustering,” *Fuzzy Sets and Systems*, vol. 389, no. 1, pp. 66–92, 2020.

[11] W. L. Y. W. S.Yue, K. Zhang, “A fuzzy clustering approach using reward and penalty functions,” *6th Int Conf on Fuzzy Systems and Knowledge Discovery*, 2009.

[12] B. B. D. Karaboga, “A powerful and efficient algorithm for numerical function optimization artificial bee colony (abc) algorithm,” *Journal of Global Optimization*, vol. 39(3), p. 459–471, 2007.

[13] S. Mirjalili and A. Lewi, “The whale optimization algorithm,” *Advancement Engineering Softwares*, vol. 95, p. 51–67, 2016.

[14] Z. L. B. Liu, C. Li, “An entropy-based metric for assessing the purity of single cell populations,” *Nat Commun*, vol. 11, pp. 31–55, 2020.

[15] P. F. A. Jain, M. Murty, “Date clustering: a review,” *ACM Computing Surveys (CSUR)*, vol. 31(3), pp. 264–323, 1999.

[16] https://archive.ics.uci.edu/ml/index.php

[17] Y. Z. W. Wang, “On fuzzy cluster validity indices,” *Fuzzy Sets and Systems*, vol. 158, pp. 2095–2117, 2007.

[18] J. C. Y. Liu, X. Zhang and H. Chao, “A validity index for fuzzy clustering based on bipartite modularity,” *Journal of Electrical and Computer Engineering*, vol. 2719617, 2019.

Manuscript received 15-04-2021; Accepted 02-08-2021

■

Anh Cuong Nguyen is a lecturer at the Air Force Defense Academy, Hanoi, Vietnam. He received the M.S degree in Computer Science in 2013 from Le Quy Don Technical University, Hanoi, Vietnam. His research interests are fuzzy logic, fuzzy clustering, image processing techniques, pattern recognition. Email: nguyenanhcuonghvktqs@gmail.com.

Thanh Long Ngo received the M.Sc, Ph.D degrees in Computer Science from Le Quy Don Technical University (LQDTU), in 2003 and 2009, respectively. He is an Associate Professor at Faculty of Information Technology, LQDTU. His current research interests include computational intelligence, type-2 fuzzy logic, pattern recognition and image processing. Email: ngotlong@gmail.com.

Dinh Sinh Mai is a lecturer at Institute of Techniques for Special Engineering, Le Quy Don Technical University (LQDTU). He received the B.S. (2009), M.S. (2013) and PhD. (2021) degrees in GeoInformatics and Computer Science from LQDTU. His research interests are fuzzy clustering, remote sensing image processing techniques, pattern recognition and geographic information system (GIS) technologies. Email: maidinhsinh@lqdtu.edu.vn.

The Long Pham received the PhD and PhDSc degrees from Belorussian State Univesity, Minsk, in 1982 and 1987, respectively. He is currently a Professor at Faculty of Information Technology, Le Quy Don Technical University. His research interests include optimization, fuzzy logic and virtual reality. He is also Vice President of Vietnam Mathematics Society. Email: longpt@mta.edu.vn.

PHÂN CỤM C-MEDOIDS MỜ LAI SỬ DỤNG THUẬT TOÁN TỐI ƯU HÓA CÁ VOI

Nguyễn Anh Cường, Ngô Thành Long, Mai Đình Sinh, Phạm Thế Long

Tóm tắt

Mặc dù thuật toán c-mean mờ (FCM) đã được sử dụng rộng rãi trong nhiều lĩnh vực, nhưng chúng rất nhạy cảm với nhiễu và các giá trị ngoại lai. Gần đây, thuật toán C-Medoids mờ (FCMdd) đã được chứng minh là hiệu quả hơn trong việc xử lý dữ liệu nhiễu. Sự khác biệt giữa FCM và FCMdd là cơ chế hình thành các cụm; trong khi FCM xây dựng các cụm dựa trên chức năng thành viên và các mẫu trong cụm, FCMdd chọn một số mẫu thực tế hiện có làm trung gian cụm. Điều này dẫn đến việc FCMdd có thể xử lý nhiễu tốt hơn FCM. Bài báo này đề xuất một cách tiếp cận kết hợp giữa thuật toán tối ưu hóa cá voi (WOA) với FCMdd để tối ưu hóa quá trình phân cụm. Sự lai ghép này ngăn cản FWCMdd rơi vào bẫy cục bộ và giúp nhanh chóng hội tụ. Giải pháp này đã được so sánh với thuật toán k-Medoids mờ (FKM) và FCMdd ban đầu. Kết quả chỉ ra rằng phương pháp đề xuất tốt hơn so với FKM và FCMdd trên hầu hết các chỉ số đánh giá.