A possibilistic Fuzzy c-means algorithm based on improved Cuckoo search for data clustering

Do Viet Duc[1*], Ngo Thanh Long[1], Ha Trung Hai[2],
Chu Van Hai[3], Nghiem Van Tam[4]

[1]Le Quy Don University;
[2]Military Information Technology Institute, Academy of Military Science and Technology;
[3]National Defence Academy;
[4]Military Logistics Academy.
[*]Email: ducdoviet@gmail.com
Received 12 Sep 2022; Revised 5 Dec 2022; Accepted 15 Dec 2022; Published 30 Dec 2022.
DOI: https://doi.org/10.54939/1859-1043.j.mst.CSCE6.2022.3-15

ABSTRACT

Possibilistic Fuzzy c-means (PFCM) algorithm is a powerful clustering algorithm. It is a combination of two algorithms Fuzzy c-means (FCM) and Possibilistic c-means (PCM). PFCM algorithm deals with the weaknesses of FCM in handling noise sensitivity and the weaknesses of PCM in the case of coincidence clusters. However, PFCM still has a common weakness of clustering algorithms that is easy to fall into local optimization. Cuckoo search (CS) is a novel evolutionary algorithm, which has been tested on some optimization problems and proved to be stable and high-efficiency. In this study, we propose a hybrid method encompassing PFCM and improved Cuckoo search to form the proposed PFCM-ICS. The proposed method has been evaluated on 4 data sets issued from the UCI Machine Learning Repository and compared with recent clustering algorithms such as FCM, PFCM, PFCM based on particle swarm optimization (PSO), PFCM based on CS. Experimental results show that the proposed method gives better clustering quality and higher accuracy than other algorithms.

Keywords: Possibilistic Fuzzy c-means; Cuckoo Search; Improved Cuckoo Search; Fuzzy clustering.

1. INTRODUCTION

Clustering is an unsupervised classification technique of data mining [1, 2]. Clustering has been used for a variety of applications such as statistics, machine learning, data mining, pattern recognition, bioinformatics, image analysis [3, 4]. Currently, there are many methods to cluster data, but the most popular are two commonly used clustering methods: hard clustering and soft (fuzzy) clustering. K-means [5] is the algorithm that represents hard clustering, that is each data point belongs only to a single cluster. This method makes it difficult to handle data where the patterns can simultaneously belong to many clusters. While Fuzzy c-means (FCM) [6] is an algorithm that represents fuzzy clustering, the membership value indicates the possibility that the data sample will belong to a particular cluster. For each data sample, the sum of the membership degree is equal to 1, and the large membership degree represents the data sample closer to the cluster centroid. However, the FCM is shown to be sensitive to noise and outliers [6]. To overcome these disadvantages Krishnapuram and Keller have presented the possibilistic c-means (PCM) algorithm [7] by abandoning the constraint of FCM and constructing a novel objective function. PCM can deal with noisy data better. But PCM is very sensitive to initialization and sometimes generates coincident clusters. PCM considers the possibility but ignores the critical membership.

To overcome these drawbacks of FCM and PCM algorithms, Pal et al. proposed the

possibilistic fuzzy c-means (PFCM) [8] algorithm with the assumption that membership and typicality are both important for accurate clustering. It is a combination of two algorithms FCM and PCM. PFCM algorithm deals with the weaknesses of FCM in handling noise sensitivity and the weaknesses of PCM in the case of coincidence clusters. However, PFCM still has a common weak point of clustering algorithms that is effortless to fall into local optimization.

Recently, nature inspired approaches have received increased attention from researchers addressing data clustering problems [9]. In order to improve PFCM algorithm, we propose in this paper to use a new metaheuristic approach. It is mainly based on the cuckoo search (CS) algorithm which was proposed by Xin-She Yang and Suash Deb in 2009 [10, 11]. CS is a search method that imitates obligate brood parasitism of some female cuckoo species specializing in mimicking then color and pattern of few chosen host birds. Specifically, from an optimization point of view, CS (i) guarantees global convergence, (ii) has local and global search capabilities controlled via a switching parameter (pa), and (iii) uses Levy flights rather than standard random walks to scan the design space more efficiently than the simple Gaussian process [12, 13]. In addition, the CS algorithm has the advantages of simple structure, few input parameters, easy realization and its superiority in benchmark comparisons [16, 17] against particle swarm optimization (PSO) and genetic algorithm (GA) which makes it a smart selection. But the CS algorithm search results largely affected by the step factor and probability of discovery. When the step size is set too high, it will lead to low search accuracy, or the step length setting is too small, it will lead to slow convergence speed [19]. In order to overcome these drawbacks of CS, Huynh Thi Thanh Binh et al. [20] proposed some improving parameters which help achieve global optimization and enhance search accuracy.

In this paper, a hybrid possibilistic fuzzy c-means PFCM clustering and improved Cuckoo search (ICS) algorithm is proposed. The efficiency of the proposed algorithm is tested on four different data sets issued from the UCI Machine Learning Repository and the obtained results are compared with some recent well-known clustering algorithms. The remainder of this paper is organized as follows. Section 2 briefly introduces some background about PFCM, Cuckoo search algorithm and improved Cuckoo search algorithm. Section 3 proposes a hybrid algorithm of PFCM and ICS. Section 4 gives some experimental results and section 5 draws conclusions and suggests future research directions.

2. BACKGROUND

2.1. Possibilistic fuzzy c-means clustering

Possibilistic Fuzzy c-means (PFCM) algorithm is a strong and reliable clustering algorithm. PFCM overcomes the problem of noise of FCM and coincident cluster problem of PCM. It is a blended version of FCM clustering and PCM clustering. The PFCM algorithm has two types of memberships: a possibilistic (t_{ik}) membership that measures the absolute degree of typicality of a point in any particular cluster and a fuzzy membership (μ_{ik}) that measures the relative degree of sharing of a point among the clusters. Given a dataset $X = \{x_k\}_{k=1}^{n} \in R^M$, the PFCM finds the partition of X into $1 < c < n$ fuzzy subsets by minimizing the following objective function:

$$J_{m,\eta}(U,T,V)=\sum_{i=1}^{c}\sum_{k=1}^{n}(au_{ik}^{m}+bt_{ik}^{\eta})d_{ik}^{2}+\sum_{i=1}^{c}\gamma_i\sum_{k=1}^{n}(1-t_{ik})^{\eta} \quad (1)$$

Where $U=\left[\mu_{ik}\right]_{c\times n}$ is a fuzzy partition matrix that contains the fuzzy membership degree; $T=\left[t_{ik}\right]_{c\times n}$ is a typicality partition matrix that contains the possibilistic membership degree; $V=(v_1,v_2,...,v_c)$ is a vector of cluster centers, *m* is the weighting exponent for the fuzzy partition matrix and η is the weighting exponent for the typicality partition matrix, $\gamma_i>0$ are constants given by the user.

The PFCM model is subject to the following constraints:

$$\sum_{i=1}^{c}u_{ik}=1;\sum_{k=1}^{n}t_{ik}=1;1\le i\le c;1\le k\le n \quad (2)$$

$$a>0,b>0,m>1,\eta>1,0\le\mu_{ik},t_{ik}\le 1 \quad (3)$$

The objective function reaches the smallest value with constraints (2) and (3) when it follows condition:

$$\mu_{ik}=1/\sum_{j=1}^{c}\left(d_{ik}^{2}/d_{jk}^{2}\right)^{1/(m-1)} \quad (4)$$

$$\gamma_i=K\sum_{k=1}^{n}u_{ik}^{m}d_{ik}^{2}/\sum_{k=1}^{n}u_{ik}^{m} \quad (5)$$

Typically, K is chosen as 1.

$$t_{ik}=1/\left(1+\left(bd_{ik}^{2}/\gamma_i\right)^{1/(\eta-1)}\right) \quad (6)$$

$$v_i=\frac{\sum_{k=1}^{n}(au_{ik}^{m}+bt_{ik}^{\eta})x_k}{\sum_{k=1}^{n}(au_{ik}^{m}+bt_{ik}^{\eta})} \quad (7)$$

In which, if *b=0* and $\gamma_i=0$, (1) becomes equivalent to the conventional FCM model while if *a = 0*, (1) reduces to the conventional PCM model. The PFCM algorithm will perform iterations according to Eqs. (4)–(7) until the objective function $J_{m,\eta}(U,T,V)$ reaches the minimum value.

The PFCM algorithm can be summarized as follows.

Algorithm 1: Possibilistic Fuzzy C-means Algorithm

Input: Dataset $X=\left\{x_k\right\}_{k=1}^{n}\in R^{M}$, the number of clusters c (*1< c < n*), fuzzifier parameters a, b, m, η, stop condition $T_{\max},\varepsilon$; and *t =0.*

Output: The membership matrix U, T and the centroid matrix V.

Step 1: Initialize the centroid matrix $V^{(0)}$ by choosing randomly from the input dataset X.

Step 2: Repeat

2.1 $t = t + 1$

2.2 Compute matrix $U^{(t)}$ by using Eq. (4)

2.3 Compute typical γ_i by using Eq. (5)

2.4 Compute matrix $T^{(t)}$ by using Eq. (6)

2.5 Update the centroid $V^{(t)}$ by using Eq. (7)

2.6 Check **if** $\left\|V^{(t)} - V^{(t-1)}\right\| \le \varepsilon$ or $t > T_{\max}$. **If** yes **then** stop and go to **Output**, **otherwise** return **Step 2**.

2.2. Cuckoo Search Algorithm

Cuckoo Search (CS) algorithm is a metaheuristic search algorithm which has been proposed recently by Yang and Deb [10, 11]. The algorithm is inspired by the reproduction strategy of cuckoos. The CS algorithm efficiently resolves the optimization problem by simulating the parasitic parenting and Levy flight of the cuckoo. Parasitization refers to the cuckoo does not nest during breeding, but laid its own eggs in other nests, with other birds to reproduce. The cuckoo will find hatching and breeding birds which is similar to their ownself [18], and quickly spawn eggs while the bird is out. Cuckoos egg usually hatch quicker than the other eggs. When this occurs, the young cuckoo would push the non-hatched eggs out of the nest. This behaviour is aimed at reducing the probability of the legitimate eggs from hatching. In addition, by imitating the calls of the host chicks, the young cuckoo chick will gain access to more food.

Levy flights are random walks whose directions are random and their step lengths resulting from the distribution of the Levy. This random walk by creating a certain length of the long and shorter steps in order to balance the local and global optimization. Compared to normal random walks, Levy flights are more effective in discovering large–scale search areas.

In order to simplify the process of cuckoo parasitism in nature, the CS algorithm is based on three idealized rules:

1. Each cuckoo only has one egg at a time and chooses a parasitic bird nest for hatching by random walk.

2. In the selected parasitic bird nest, only the best nest can be retained to the next generation.

3. The number of nests is fixed and there is a probability that a host can discover an alien egg. The host will discard either the egg or the nest if this occurs, and as a result in the building of a new nest in a new location.

With the above three idealized rules, the search for a new bird's nest location path is as follows:

$$x_i^{(t+1)} = x_i^{(t)} + \alpha \oplus Levy(\lambda); i = 1, 2, ..., n \tag{8}$$

In which $x_i^{(t)}$ stands for the *ith* bird's nest position in the t generation, $\alpha(\alpha > 0)$ is the step size control, usually $\alpha = 1$. $Levy(\lambda)$ is Levy random search path, its expression is as follows:

$$Levy(\lambda) = t^{-\lambda}; 1 < \lambda < 3 \quad (9)$$

Cuckoo search algorithm is very effective for global optimization problems since it maintains a balance between local random walk and the global random walk. The balance between local and global random walks is controlled by a switching parameter $p_a \in [0,1]$. After the new solution is generated, some solutions are discarded according to a probability p_a, and then the corresponding new solution is generated by the way of random walks, and iteration is completed. The CS algorithm flows as follows:

Algorithm 2: Cuckoo Search Algorithm

Input: Objective function $f(x), X = (x_1, x_2, ..., x_d)^T$; *stop criterion*; $T_{\max}$.

Output: Postprocess results and visualization.

Step 1: Generate initial population of n host nests $x_i (i = 1, 2, ..., n)$

Step 2: ***While*** $(t \le T_{\max})$ *or (stop criterion)*

2.1 *t = t +1*

2.2 Get a cuckoo randomly by Levy flights evaluate its quality/fitness *Fi*

2.3 Choose a nest among n (say, j) randomly

If $(F_i \le F_j)$ **then** replace j by the new solution

2.4 A fraction (p_a) of worse nests are abandoned and new ones are built;

2.5 Keep the best solutions (or nests with quality solutions);

2.6 Rank the solutions and find the current best.

2.3. Improve Cuckoo Search (ICS) Algorithm

The original CS algorithm is influenced by step size α and probability of discovery p_a , and the step size and discovery probability control the accuracy of CS algorithm global and local search, which has a great influence on the optimization effect of the algorithm. At the start of the algorithm, these constants are selected and have a great influence on the performance of CS. When the step size is set too large, reducing the search accuracy, easy convergence, step length is too small, reducing the search speed, easy to fall into the local optimal. Therefore, the new solutions could be pushed away from the optimal ones because of the continuous adoption of large step lengths, particularly when the number of generation is large enough. In order to address these CS algorithm disadvantages, Huynh Thi Thanh Binh et al. [20] proposed improving p_a and α as follows:

$$p_a(t) = p_{a_{\max}} - \frac{t}{T_{\max}} \left(p_{a_{\max}} - p_{a_{\max}} \right) \quad (10)$$

$$\alpha(t) = \alpha_{\max} e^{\frac{t}{T_{\max}} \ln\left(\frac{\alpha_{\min}}{\alpha_{\max}}\right)} \quad (11)$$

Where $p_{a_{\max}}$ and $p_{a_{\min}}$ are the highest and lowest probability of discovering Cuckoo eggs. $\alpha_{\max}$ and $\alpha_{\min}$ are the maximum step size and the minimum step size, respectively. $T_{\max}$ is the maximum number of generations. t is the index of the current generation.

Following (10) and (11), the value of $p_a(t)$ and $\alpha(t)$ are large in some first loop of ICS in order to create a wide range searching space. After that, they are gradually decreasing so as to increase the convergence rate and maintain good solutions in the populations. Therefore, it may achieve global optimization and speed up the iterative velocity and in the latter part of the algorithm iteration, with a smaller step size which helps enhance search accuracy to achieve local optimization.

3. PROPOSAL METHOD

In this study, we propose an algorithm called PFCM-ICS, which is combined with the PFCM clustering algorithm with the improved Cuckoo search algorithm presented in this paper. The PFCM-ICS algorithm was used the same fitness function as PFCM algorithm. It is described as follows:

$$F_{PFCM-ICS}(U,T,V) = J_{PFCM}(U,T,V) = \sum_{i=1}^{c}\sum_{k=1}^{n}(au_{ik}^{m} + bt_{ik}^{\eta})d_{ik}^{2} + \sum_{i=1}^{c}\gamma_i\sum_{k=1}^{n}(1-t_{ik})^{\eta} \qquad (12)$$

In order to solve the data clustering problem, the ICS algorithm is adapted to reach the centroids of the clusters. For doing this, we suppose that we have *n* objects, and each object is defined by *m* attributes. In this work, the main goal of the ICS is to find *c* centroids of clusters which minimize the fitness function (12). In the ICS mechanism, the solutions are the nests and each nest is represented by a matrix *(c,m)* with *c* rows and *m* columns, where, the matrix rows are the centroids of clusters. After ICS was conducted, the best solution was the best centroids which the fitness function (12) reached the minimum value.

We propose PFCM-ICS algorithm for the data clustering through the following steps:

Algorithm 3: PFCM-ICS Algorithm

Input: Dataset $X = \{x_k\}_{k=1}^{n} \in R^M$, the number of clusters c (*1< c < n*), fuzzifier parameters a, b, m, η, stop condition $T_{\max}$; number of populations *p* and *t =0.*

Output: F_{Best}, V_{Best},the membership matrix U, T.

Step 1: Initialization

1.1 Initialize population of nests by using the FCM algorithm.

$p_{nests} = \left[V_j^{(0)}\right]; j = 1,..., p;$

$V_j^{(0)} = \left[v_i^{(0)}\right]; i = 1,..., c; V_j^{(0)} \in R^{CxM}$

1.2 Calculate fitness of all nests by using Eqs. (4) - (6) and (12).

1.3 Sort to find the best fitness F_{Best} and it is also best centroids V_{Best}

Step 2: Hybrid algorithm of PFCM and ICS

2.1 *t = t +1*

2.2 Generate new solution *i* by Eqs. (8-9) and (11).

2.3 Calculate *Fi* by using Eqs. (4) - (6) and (12).

2.4 Select random nest *j (i#j).*

If (*Fi* < *Fj*) ***then*** Replace *j* by new solution *i*

2.5 Sort to keep the best fitness

2.6 Generate a fraction p_a by using Eq. (10) of new solutions to replace the worse nests by random. Calculate fitness of these nests by Eqs. (4) - (6) and (12).

2.7 Sort to find the best fitness F_{Best}, V_{Best}

2.8 Check **If** ($t > T_{\max}$) **then** go to **Step 3**, **otherwise** return **Step 2**.

Step 3: Compute matrix

3.1 Compute matrix $U^{(t)}$ by using Eq. (4)

3.2 Compute typical γ_i by using Eq. (5)

3.3 Compute matrix $T^{(t)}$ by using Eq. (6)

The PFCM-ICS algorithm will perform iterations until the fitness function $F_{PFCM-ICS}(U,T,V)$ reaches the minimum value, and this algorithm's computational complexity with $T_{\max}$ is $O\left((p+6)T_{\max}Mnc\right)$.

4. EXPERIMENTAL RESULTS AND DISCUSSIONS

4.1. Dataset description

In this section, we perform several experiments to verify the performance of the proposed algorithm. The experiments were tested on the four datasets from the UCI Machine Learning Repository. All four UCI datasets we use in our experiments are common databases that can easily be accessed at: https://archive.ics.uci.edu/ml/index.php. In table 1, we describe the typical features of the datasets including iris, wine, seeds, breast cancer datasets.

Table 1. *The characteristics of the datasets.*

Dataset	Number of Instances	Number of Features	Number of Clusters
Iris	150	4	3
Wine	178	13	3
Seeds	210	7	3
Breast Cancer	569	32	2

4.2. Parameter initialization and validity measures

In order to verify the effectiveness of the proposed approach, experimental algorithms include FCM [6], PFCM [8], PFCM-PSO [29], PFCM-CS and PFCM-ICS. The algorithms are executed for a maximum of 500 iterations and $\varepsilon = 10^{-6}$. For all algorithms, we first ran FCM algorithm with m =2 to determine the initial centroids. With the algorithms PFCM, PFCM-PSO, PFCM-CS and PFCM-ICS, K= 1 was selected to calculate the value γ_i by using Eq. (5). The parameters of the PFCM, PFCM-PSO, PFCM-CS and PFCM-ICS algorithms were selected as follows: $a = b = 1$, $m = n = 2$. In the PFCM-PSO algorithm, the parameters $c_1 = c_2 = 2.05$ and $\omega = 0.9$ as suggested in the paper [30]. The parameters of the PFCM-CS algorithm, the population size, step size,

probability were selected p=15, $\alpha = 0.01$, $p_a = 0.25$, respectively. With the algorithm PFCM-ICS, the population size was selected p=15, the step size was calculated by using Eq. (11) with $\alpha_{\max} = 0.5; \alpha_{\min} = 0.01$, the probability was calculated by using Eq. (10) with $p_{a_{\max}} = 0.5; p_{a_{\min}} = 0.05$.

To assess the performance of algorithms, we use the following evaluation indicators as follows: Bezdek partition coefficient index (PC-I) [22], Dunn separation index (D-I), the classification entropy index (CE-I) [23], the Silhouette score (SC) [31], the Separation index (S-I) [32], the sum squared error index (SSE) [24] and Davies Bouldin index [25]. Large values for indexes PC-I, D-I and SC are good for clustering results, while small values for indexes CE-I, S-I, DB-I and SSE are good for clustering results. In addition, the clustering results were measured using the accuracy measure r defined in [27] as:

$$r = \frac{1}{n}\sum_{i=1}^{c} a_i \tag{13}$$

Where a_i is the number of data occurring in both the i^{th} cluster and its corresponding true cluster, and n is the number of data points in the dataset. The higher value of accuracy measure r proves superior clustering results with perfect clustering generating a value $r = 1$.

4.3. Results and discussion

Table 2. *Index assessment of algorithms FCM, PFCM, PFCM-PSO, PFCM-CS, and PFCM-ICS on Iris dataset.*

Method	DB-I	D-I	SSE	SC	PC-I	CE-I	S-I	Accuracy
FCM	0.7738	0.0347	7.0668	0.4959	0.7425	0.4672	0.1206	0.8866
PFCM	0.7697	0.0701	7.0552	0.4994	0.7434	0.466	0.1206	0.8933
PFCM-PSO	0.7867	0.0634	7.0546	0.4829	0.7389	0.4658	0.1208	0.9133
PFCM-CS	**0.7648**	0.0701	7.0445	**0.5022**	0.7462	0.4658	0.1206	0.9133
PFCM- ICS	**0.7648**	**0.0735**	**7.0373**	**0.5022**	**0.7489**	**0.4641**	**0.1201**	**0.9266**

Table 3. *Index assessment of algorithms FCM, PFCM, PFCM-PSO, PFCM-CS, and PFCM-ICS on Wine dataset.*

Method	DB-I	D-I	SSE	SC	PC-I	CE-I	S-I	Accuracy
FCM	1.3181	0.1423	49.9553	0.2993	0.5033	0.8546	0.2709	0.9494
PFCM	1.3184	0.1423	49.5581	0.3003	0.5118	0.8426	0.2668	0.9551
PFCM-PSO	**1.3062**	0.1728	49.3863	**0.2991**	0.5214	0.8376	0.2661	0.9607
PFCM-CS	1.3115	0.1893	49.4105	0.3005	0.5216	0.8282	0.2655	0.9607
PFCM- ICS	1.3126	**0.1923**	**49.2541**	0.3001	**0.5253**	**0.8232**	**0.2651**	**0.9663**

We have conducted clustering on the different algorithms such as FCM, PFCM, PFCM-PSO, PFCM-CS and PFCM-ICS on four datasets. The experimental results are shown in some tables from 2 to 6. The clustering results obtained on the datasets Iris, Wine, Seeds, Breast Cancer are described in table 2, table 3, table 4, table 5, respectively.

Table 4. *Index assessment of algorithms FCM, PFCM, PFCM-PSO, PFCM-CS, and PFCM-ICS on Seeds dataset.*

Method	DB-I	D-I	SSE	SC	PC-I	CE-I	S-I	Accuracy
FCM	0.8795	0.0835	22.0885	0.4229	0.6815	0.5728	0.1467	0.8952
PFCM	0.8795	0.0866	22.0773	0.4229	0.6822	0.5719	0.1466	0.8959
PFCM-PSO	0.8795	0.0868	22.0702	0.4218	0.6822	0.5723	0.1465	0.8959
PFCM-CS	0.8795	0.0881	22.0658	0.4212	0.6834	0.5708	0.1464	0.8995
PFCM- ICS	**0.8795**	**0.0885**	**22.0612**	**0.4206**	**0.6838**	**0.5702**	**0.1464**	**0.9095**

Table 5. *Index assessment of algorithms FCM, PFCM, PFCM-PSO, PFCM-CS, and PFCM-ICS on Breast Cancer dataset.*

Method	DB-I	D-I	SSE	SC	PC-I	CE-I	S-I	Accuracy
FCM	1.1446	0.0838	217.5632	0.3794	0.6981	0.4707	0.4124	0.9232
PFCM	1.1446	0.0838	216.9461	0.3794	0.7022	0.4654	0.4031	0.9279
PFCM-PSO	1.1443	0.0838	216.4462	0.3845	0.7139	0.4608	0.3999	0.9279
PFCM-CS	**1.1407**	0.0838	216.4082	0.3829	**0.7157**	0.4611	0.3936	0.9279
PFCM- ICS	1.1446	**0.0838**	**216.3915**	**0.3894**	0.7152	**0.4605**	**0.3915**	**0.9332**

Table 6. *Fitness value of algorithms PFCM, PFCM-PSO, PFCM-CS, and PFCM-ICS on all datasets.*

Method	PFCM	PFCM-PSO	PFCM-CS	PFCM- ICS
IRIS	24.7676	24.7302	24.6799	**24.6652**
WINE	163.2433	163.1238	162.7228	**162.5749**
SEEDS	76.2298	76.2263	76.2138	**76.1864**
BREAST CANCER	479.6932	479.6231	479.4112	**479.4072**

From the clustering results of the four datasets which are shown in the tables from 2 to 6, according to the properties of datasets which are described in table 1 and Fig. 1, some conclusions are revealed as follows:

- The results summarized in table 2, table 3, table 4, table 5 show that the PFCM-ICS algorithm produces better quality clustering than those obtained when running other commonly encountered algorithms such as FCM, PFCM, PFCM-PSO, PFCM-CS. It is apparent that in terms of validity measures D-I, PC-I, DB-I, SSE, CE-I, SC and S-I, the performance of the proposed PFCM-ICS is better for most of the datasets.
- Performance of the proposed PFCM-ICS algorithm is also measured by the clustering accuracy *r*. Again, the proposed algorithm obtained the highest clustering accuracy score for all datasets. The clustering accuracy obtained on the dataset Seeds, Iris, Breast Cancer, Wine are 90.95%, 92.66%, 93.32%, 96.63%, respectively.
- Fig. 1 describes the detailed clustering accuracy of all algorithms on four datasets. These results exhibit that the PFCM-ICS produces a better clustering solution than the other algorithms such as FCM, PFCM, PFCM-PSO and PFCM-CS.

- In table 6, the fitness values of the PFCM, PFCM-PSO, PFCM-CS, PFCM-ICS algorithms were compared on four datasets. The results show that the PFCM-ICS algorithm achieved the best fitness values for all datasets. The fitness values obtained on the dataset Iris, Wine, Seeds, Breast Cancer are 24.6652, 162.5749, 76.1864, 479.4072, respectively.

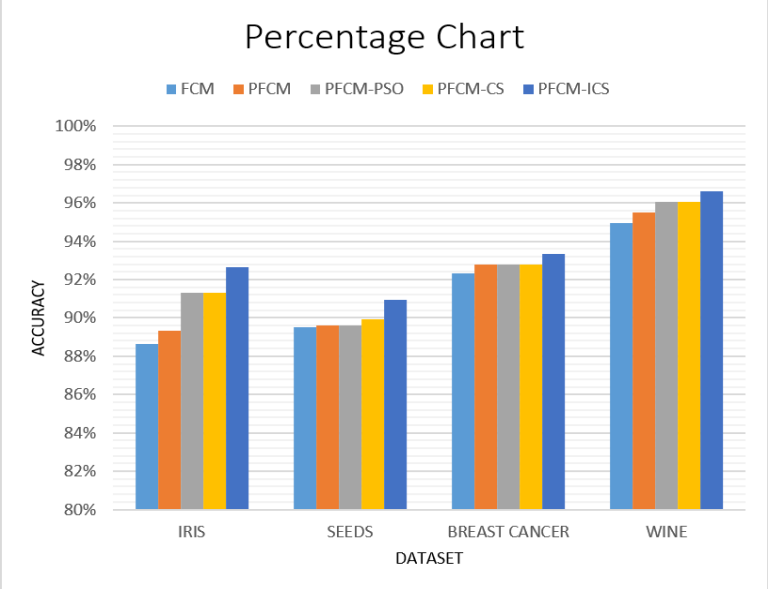

Figure 1. *The clustering accuracy of algorithms: FCM, PFCM, PFCM-PSO, PFCM-CS, and PFCM-ICS.*

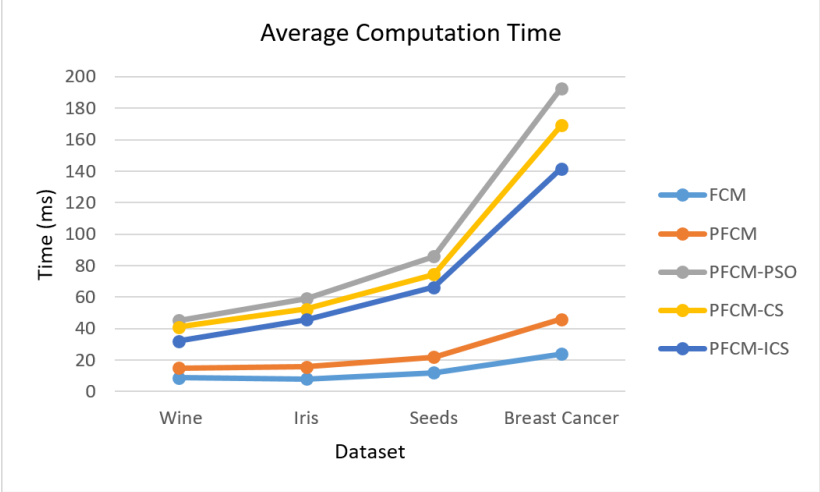

Figure 2. *Comparison of 30 running times between FCM, PFCM, PFCM-PSO, PFCM-CS, PFCM-ICS.*

Table 7 and Fig. 2 presents a comparison of the computation time it will take for FCM, PFCM, PFCM-PSO, PFCM-CS, PFCM-ICS algorithms for four datasets. The algorithms were executed 30 times to calculate the averaging time. Compared with hybrid algorithms, PFCM-ICS gives better computation time than PFCM-PSO, PFCM-CS and it is clearly shown by the observation of Fig. 2.

Table 7. *Average computation time for algorithms in millisecond.*

Method	FCM	PFCM	PFCM-PSO	PFCM-CS	PFCM-ICS
Wine	8.771	14.952	45.222	41.084	32.028
Iris	7.987	15.582	59.044	52.453	45.764
Seeds	11.983	21.979	85.746	74.386	66.365
Breast Cancer	23.968	45.854	192.659	169.065	141.654

5. CONCLUSIONS

In this paper, a PFCM clustering algorithm based on an improved cuckoo search is proposed. The experimental results show that the proposed method can achieve higher accuracy and obtain better fitness values than some recent well-known clustering algorithms. According to the clustering results, when using some indicators to assess cluster quality, the PFCM-ICS algorithm achieves the best results in most cases. In general, the PFCM-ICS algorithm shows that it is a trustful, stable, accurate clustering algorithm and outperforms FCM, PFCM, PFCM-PSO and PFCM-CS. In order to improve the obtained results and as future work, we wish to develop a kernel method based on PFCM and parallel models to solve the complex problem of big data and accelerate computation.

REFERENCES

[1]. Jain, K., Murthy, M.N., Flynn, P.J.: *"Data Clustering: A Review"*. ACM Computing Surveys 31(3), 264–323 (1999)

[2]. Xu, R., Wunsch, D.C.: *"Clustering"*, 2nd edn., pp. 1–13. IEEE Press, John Wiley and Sons, Inc. (2009)

[3]. I. H. Witten and E. Frank, *"Data Mining-Pratical Machine Learning Tools and Techniques"*, 3rd ed. Morgan Kaufmann Publishers, Inc., (2011).

[4]. S. Mitra and T. Acharya, *"Data Mining: Multimedia"*, Soft Computing, and Bioinformatics. Wiley, (2003).

[5]. Anil K. Jain. *"Data clustering: 50 years beyond K-means"*. Pattern Recognition Letters; 31(8):651-666, (2010).

[6]. J. C. Bezdek, *"Pattern Recognition with Fuzzy Objective Function Algorithms"*. Plenum Press, New York, (1981).

[7]. R. Krishnapuram and J. Keller, *"A possibilistic approach to clustering,"* IEEE Trans. Fuzzy Systems, vol. 1. no. 2, pp. 98–110, (1993).

[8]. N. R. Pal, K. Pal, and J. C. Bezdek, *"A possibilistic fuzzy c-means clustering algorithm,"* IEEE Trans. Fuzzy Systems, vol. 13, no. 4, pp. 517–530, (2005).

[9]. Colanzi, T.E., Assunção, W.K.K.G., Pozo, A.T.R., Vendramin, A.C.B.K., Pereira, D.A.B., Zorzo, C.A., de Paula Filho, P.L.: *"Application of Bio inspired Metaheuristics in the Data Clustering Problem"*. Clei Electronic Journal 14(3) (2011)

[10]. Yang, X.-S., Deb, S.: *"Cuckoo Search via Levy Flights"*. In: Proc. of World Congress on Nature and Biologically Inspired Computing (NaBIC 2009), India, pp. 210–214. IEEE Publications, USA (2009)

[11]. Yang, X.-S., Deb, S.: *"Engineering Optimisation by Cuckoo Search"*. International Journal of Mathematical Modelling and Numerical Optimisation 1(4-30), 330–343 (2010)

[12]. M. Jamil, H.J. Zepernick, X.S. Yang. *"Levy Flight Based Cuckoo Search Algorithm for Synthesizing Cross-Ambiguity Functions"*. IEEE Military Communications Conference (Milcom), San Diego, CA; 823–828, (2013).

[13]. X.-S. Yang, *"Nature-inspired Optimization Algorithm"*, first ed. Elsevier, MA, USA, (2014).

[14]. Jothi, R., Vigneshwaran, A.: *"An Optimal Job Scheduling in Grid Using Cuckoo Algorithm"*. International Journal of Computer Science and Telecommunications 3(2), 65–69 (2012).

[15]. Noghrehabadi, A., Ghalambaz, M., Ghalambaz, M., Vosough, A.: *"A hybrid Power Series –Cuckoo Search Optimization Algorithmto Electrostatic Deflection of Micro Fixed-fixed Actuators"*. International Journal of Multidisciplinary Sciences and Engineering 2(4), 22–26 (2011).

[16]. L.D. Coelho, C.E. Klein, S.L. Sabat, V.C. Mariani. *"Optimal chiller loading for energy conservation using a new differential cuckoo search approach"*. Energy. 2014; 75 (1) :237–243.

[17]. A. Natarajan, S. Subramanian, K. Premalatha. *"A comparative study of cuckoo search and bat algorithm for Bloom filter optimisation in spam filtering"*. Int. J. Bio-Inspir. Comp; 4 (2): 89–99, (2012).

[18]. Liyu, Mliang. *"New Meta-heuristic Cuckoo Search Algorithm"*. Systems engineering. (08),64-69, (2012).

[19]. Yang X-S, Deb S. *"Cuckoo search:recent advances and applications"*, Neural Computing and Applications, 24(1):169-174, (2014).

[20]. Huynh Thi Thanh Binh, Nguyen Thi Hanh, La Van Quan, Nilanjan Dey, *"Improved Cuckoo Search and Chaotic Flower Pollination optimization algorithm for maximizing area coverage in Wireless Sensor Networks"*, Neural Comput & Applic, (2016).

[21]. U. Maulik, S. Bandyopadhyay, *"Performance evaluation of some clustering algorithms and validity indices"*, IEEE Trans. Pattern Anal. Mach. Intell. 24 (12), 1650–1654, (2002).

[22]. J.C. Bezdek, N. Pal, *"Some new indexes of cluster validity"*, IEEE Trans. Syst. Man Cybern. 28 (3), 301–315, (1998).

[23]. C.H. Chou, M.C. Su, E. Lai, *"A new cluster validity measure and its application to image compression"*, Pattern Anal. Appl. 7 (2), 205–220, (2004).

[24]. Z. Wang, A.C. Bovik, *"Mean squared error: love it or leave it? A new look at signal fidelity measures"*, IEEE Signal Process. Mag. 98–117, (2009).

[25]. J. Cao, Z. Wu, J. Wu, and H. Xiong, *"SAIL: Summationbased incremental learning for informationtheoretic text clustering,"* IEEE Transactions on Cybernetics, vol. 43, no. 2, pp. 570–584, (2013).

[26]. Dinh Sinh Mai, Long Thanh Ngo, Le Hung Trinh, Hani Hagras, *"A hybrid interval type-2 semi-supervised possibilistic fuzzy c-means clustering and particle swarm optimization for satellite image analysis"*, Information Sciences, (2020).

[27]. Z. Huang, M.K. Ng, *"A fuzzy k-modes algorithm for clustering categorical data"*, IEEE Trans. Fuzzy Syst. 7 (4), 446–452, (1999).

[28]. M. Mareli, B. Twala, *"An adaptive Cuckoo search algorithm for optimisation"*, Applied Computing and Informatics, 2210-8327 / (2017).

[29]. Y. He, K. Zhang, Z. Sun, *"A possibilistic fuzzy c-means clustering algorithm based on improved particle swarm optimization"*, Journal of Computational Information Systems 10(18):7845-7857, (2014).

[30]. J. Kennedy, R. Eberhart, *"Particle swarm optimization"*, in: IEEE International Conference on Neural Networks, pp. 1942–1948, (1995).

[31]. Artur Starczewski, Adam Krzyżak, *"Performance Evaluation of the Silhouette Index"*, ICAISC 2015: Artificial Intelligence and Soft Computing, pp 49-58, (2015).

[32]. Dae-WonKim, Kwang H.Lee, DoheonLee, *"On cluster validity index for estimation of the optimal number of fuzzy clusters"*, Pattern Recognition, Volume 37, Issue 10, Pages 2009-2025, (2004).

[33]. Dinh Sinh Mai, Long Thanh Ngo, Hung Le Trinh, Hani Hagras: *"A hybrid interval type-2 semi-supervised possibilistic fuzzy c-means clustering and particle swarm optimization for satellite image analysis"*. Inf. Sci. 548: 398-422 (2021).

[34]. Dinh Sinh Mai, Trong Hop Dang: *"An improvement of collaborative fuzzy clustering based on active semi-supervised learning"*. FUZZ-IEEE 2022: 1-6, (2022).

[35]. Tran Manh Tuan, Dinh Sinh Mai, Tran Dinh Khang, Phung The Huan, Tran Thi Ngan, Long Giang Nguyen, Vu Duc Thai: *"A New Approach for Semi-supervised Fuzzy Clustering with Multiple Fuzzifiers"*. Int. J. Fuzzy Syst. 24(8): 3688-3701 (2022).

[36]. Abdullah Alghamdi: *"A Hybrid Method for Big Data Analysis Using Fuzzy Clustering, Feature Selection and Adaptive Neuro-Fuzzy Inferences System Techniques: Case of Mecca and Medina Hotels in Saudi Arabia"*. Arabian Journal for Science and Engineering (2022).

TÓM TẮT

Thuật toán phân cụm mờ xác xuất C-mean dựa trên cải tiến của thuật toán tìm kiếm Cuckoo cho bài toán phân cụm dữ liệu

Thuật toán phân cụm mờ xác xuất C-mean (PFCM) là một thuật toán phân cụm mạnh mẽ. Nó là sự kết hợp của hai thuật toán phân cụm mờ C-mean (FCM) và phân cụm xác xuất C-mean (PCM). Thuật toán PFCM giải quyết các điểm yếu của FCM trong việc xử lý với dữ liệu có nhiều nhiễu và các điểm yếu của PCM trong trường hợp các cụm chồng lấp. Tuy nhiên, PFCM vẫn có một điểm yếu chung là thuật toán phân cụm dễ rơi vào tối ưu cục bộ. Cuckoo search (CS) là một thuật toán tiến hóa mới, đã được thử nghiệm trên một số bài toán tối ưu và tỏ ra ổn định, hiệu quả cao. Trong nghiên cứu này, chúng tôi đề xuất một phương pháp kết hợp bao gồm PFCM và tìm kiếm Cuckoo được cải tiến để tạo thành thuật toán PFCM-ICS. Phương pháp đề xuất đã được đánh giá trên 4 bộ dữ liệu được phát hành từ kho dữ liệu UCI và được so sánh với các thuật toán phân cụm gần đây như FCM, PFCM, PFCM dựa trên tối ưu hóa bầy đàn (PSO), PFCM dựa trên CS. Kết quả thực nghiệm cho thấy phương pháp đề xuất cho chất lượng phân cụm tốt hơn và độ chính xác cao hơn so với các thuật toán khác.

Từ khóa: Phân cụm mờ xác xuất C-mean; Tìm kiếm Cuckoo; Tìm kiếm Cuckoo được cải tiến; Phân cụm mờ.