Neural Computing and Applications (2020) 32:16891–16899
https://doi.org/10.1007/s00521-018-03977-x

SMART DATA AGGREGATION INSPIRED PARADIGM & APPROACHES IN IOT APPLNS

An adaptive artificial-fish-swarm-inspired fuzzy C-means algorithm

Liang Xi[1] · Fengbin Zhang[1]

Received: 24 October 2018 / Accepted: 20 December 2018 / Published online: 1 January 2019

Abstract

Fuzzy C-means (FCM) is a classical algorithm of cluster analysis which has been applied to many fields including artificial intelligence, pattern recognition, data aggregation and their applications in software engineering, image processing, IoT, etc. However, it is sensitive to the initial value selection and prone to get local extremum. The classification effect is also unsatisfactory which limits its applications severely. Therefore, this paper introduces the artificial-fish-swarm algorithm (AFSA) which has strong global search ability and adds an adaptive mechanism to make it adaptively adjust the scope of visual value, improves its local and global optimization ability, and reduces the number of algorithm iterations. Then it is applied to the improved FCM which is based on the Mahalanobis distance, named as adaptive AFSA-inspired FCM(AAFSA-FCM). The optimal solution obtained by adaptive AFSA (AAFSA) is used for FCM cluster analysis to solve the problems mentioned above and improve clustering performance. Experiments show that the proposed algorithm has better clustering effect and classification performance with lower computing cost which can be better to apply to every relevant area, such as IoT, network analysis, and abnormal detection.

Keywords Cluster analysis · Data aggregation · Classifier · Global optimization

1 Introduction

Cluster analysis is one of the classic methods often used in artificial intelligence, network security, IoT technologies, etc. Its basic idea is to analyze some invisible relationship between data samples and then divide these samples into different clusters [1]. Fuzzy C-means clustering algorithm (FCM) is one of the most classic algorithms in unsupervised machine learning [2, 3]. It can associate a sample with two or more clusters using membership grades indicating the degrees of sample belonging to the corresponding clusters. Many emerging technologies also use the FCM as a classifier and have some ideal performances [4]. However, FCM is highly dependent on the initial value and prone to fall into local minimum values [5]. In order to solve these problems, people have adopted a variety of optimization measures to improve it, especially, in the aspect of reducing the dependence on the initial value to some extent [6]. However, those improvements which are mainly more complicated to implement have a mass of computational cost and more iterations [7].

Artificial-fish-swarm algorithm (AFSA) is an intelligent biological optimization algorithm designed to simulate the foraging behavior of fishes [8]. It is insensitive to the initial value selection, simple in calculation, and has strong robustness and preferably global convergence [9]. Therefore, applying AFSA to FCM can help solve the problem that is easily trapped in local extremum. However, the number of iterations of AFSA is difficult to control, which needs to be taken seriously.

From the above, this paper proposes the adaptive AFSA-inspired FCM algorithm (AAFSA-FCM), improves the AFSA with an adaptive mechanism to control the iterations, and then combines it with improved FCM with Mahalanobis distance. The basic process is as follows: Improve AFSA to adaptively change the visual field with continuous iteration until the optimal solution is obtained. And then, use the solution as the FCM initial clustering

✉ Fengbin Zhang
zhangfengbin@hrbust.edu.cn

[1] School of Computer Science and Technology, Harbin University of Science and Technology, Harbin, China

center for cluster analysis; the affinity calculation is based on the Mahalanobis distance.

The remaining structure of the article is as follows: Sect. 2 improves the FCM with Mahalanobis distance. Section 3 introduces AFSA and proposes the AAFSA and necessary analyzes in detail. And then, AAFSA-FCM is presented in Sect. 4, and the experiments are carried out in Sect. 5. Finally, Sect. 6 gives some concluding remarks by the experiments.

2 Improved fuzzy C-means algorithm with Mahalanobis distance

FCM's drawbacks mentioned before reduced its actual effects in many application areas. In consequence, people improved the traditional FCM with some other intelligent algorithms in some application background. Liu L proposed a new modified FCM. Its initialization and globally optimum cluster center were produced by chaotic quantum particle swarm optimization (CQPSO), which can effectively avoid the FCM's disadvantages above [10]. Xiao Mansheng et al. [11] proposed an FCM based on spatial correlation and membership smoothing, designed the influence value according to the spatial distribution characteristics of the data set to improve the clustering center and then to reduce the sensitivity to noise. Chen Haipeng et al. introduced an automatic partitioning method, which used different clustering numbers to perform multiple clustering analyses, and used the relevant membership information obtained by optimization to construct the correlation matrix to obtain the final result [12]. In this paper, the adaptive AFSA is introduced to reduce the effect of sensitive dependence on initial conditions and being trapped into the local optimum.

In the FCM, the Euclidean distance is often used to measure the difference between samples, which is more suitable for clustering analysis of the data set with spherical structure [13]. When the dimension of problem space is at a higher level, it takes a long time to converge. Conversely, the Mahalanobis distance is only related to the inverse of the covariance matrix, which cares about the number of samples rather than the dimension of the samples. Therefore, it is more suitable for clustering higher-dimensional data vectors, which has less computing time than the Euclidean distance. Ref. [14] adaptively adjusts the distribution of data set by Mahalanobis distance and apply it to fuzzy clustering to obtain better results. Therefore, before improving AFSA, this paper uses the Mahalanobis distance to measure the difference between samples.

(1) Mahalanobis distance

Let $\boldsymbol{A}$ be an $n \times l$ input matrix containing n samples, $\boldsymbol{A} = (\boldsymbol{n}_i)(i = 1, 2, \ldots, n)$, The Mahalanobis distance between the sample mean of $\boldsymbol{n}_i$ to $\boldsymbol{A}$ can be defined as

$$d_{\mathrm{M}} = \sqrt{(\boldsymbol{n}_i - \bar{\boldsymbol{n}})^{\mathrm{T}} \boldsymbol{C}^{-1} (\boldsymbol{n}_i - \bar{\boldsymbol{n}})}. \tag{1}$$

where $\bar{\boldsymbol{n}}$ is the samples' mean and $\boldsymbol{C}$ is the covariance matrix expressed as:

$$\boldsymbol{C} = \frac{1}{n} \sum_{i=1}^{n} (\boldsymbol{n}_i - \bar{\boldsymbol{n}})(\boldsymbol{n}_i - \bar{\boldsymbol{n}})^{\mathrm{T}}. \tag{2}$$

(2) Improved FCM with Mahalanobis distance

Its objective function can be expressed as:

$$J(\boldsymbol{U}, \boldsymbol{V}, \boldsymbol{C}) = \sum_{i=1}^{c} \sum_{j=1}^{n} u_{ij}^{m} (\boldsymbol{n}_j - \bar{\boldsymbol{n}}_i)^{\mathrm{T}} \boldsymbol{C}^{-1} (\boldsymbol{n}_j - \bar{\boldsymbol{n}}_i). \tag{3}$$

where $\boldsymbol{U}$ is the membership matrix and $\boldsymbol{V}$ is the cluster center matrix. The goal of the FCM with Mahalanobis distance is to get the minimum of this formula. Its constraints:

$$\sum_{i=1}^{c} u_{ij} = 1,\ u_{ij} \in [0, 1] \tag{4}$$

is the degree of membership. Then this algorithm uses the Lagrange multiplier method to get the following formulas:

$$\bar{\boldsymbol{n}}_i = \frac{\sum_{j=1}^{n} u_{ij}^{m} \boldsymbol{n}_j}{\sum_{j=1}^{n} u_{ij}^{m}}, \tag{5}$$

$$\boldsymbol{T}_i = \frac{\sum_{j=1}^{n} u_{ij}^{m} (\boldsymbol{n}_j - \bar{\boldsymbol{n}}_i)(\boldsymbol{n}_j - \bar{\boldsymbol{n}}_i)^{\mathrm{T}}}{\sum_{j=1}^{n} u_{ij}^{m}}, \tag{6}$$

$$d_{ij}^{2} = (\boldsymbol{n}_j - \overline{\boldsymbol{n}_i})^{\mathrm{T}} [\overline{\boldsymbol{n}_i} \det(\boldsymbol{T}_i)^{1/l} \boldsymbol{T}_i^{-1}] (\boldsymbol{n}_j - \overline{\boldsymbol{n}_i}), \tag{7}$$

$$u_{ij} = 1 \Big/ \sum_{k=1}^{c} (d_{ij}/d_{kj})^{2/m-1}. \tag{8}$$

where $1 \leq i \leq c$, $1 \leq j \leq n$, c is the number of cluster center.

This algorithm need set three parameters: c, iterative termination error (ε) and fuzzy weighted exponent (m). Here c is given in advance; and $m > 1$, obtained by Ref. [15], generally is set with the median of [1.5,2.5]; ε is usually set with the value of 10^{-5}. The algorithm processes are the same as the traditional FCM on the whole.

(3) Algorithm analyses

I. The time complexity of initializing $\boldsymbol{U}$ is $O(cn)$, The time complexity for initializing other parameters is $O(1)$.

II. The time complexity of calculating cluster centers is $O(nlc)$. The time complexity of calculating the objective function is also $O(nlc)$.

III. The algorithm takes f iterations, so the total time complexity of the algorithm is $O(nlcf)$.

3 Adaptive artificial-fish-swarm algorithm (AAFSA)

3.1 Artificial-fish-swarm algorithm (AFSA) and its problem analysis

AFSA is one of the intelligent global optimization algorithms based on the free swimming characteristics of fishes. Its basic idea is that, the more fish are gathered, the more abundant food in the area. According to this feature, artificial fishes are constructed to imitate the prey, living and chase behaviors of fishes in the natural environments [16]. At present, the relevant publications have described the effects of AFSA in different application backgrounds. For example, Ref. [17] proposed a novel attribute reduction algorithm based on AFSA and rough set which can search the attribute reduction set effectively, have low time complexity and excellent global search ability. In Ref. [18], a feature selection method based on AFSA optimization was proposed, which was applied to big data processing to achieve better performance on reducing database dimensions. Ref. [19] exploits the optimization capability of AFSA using error diffusion algorithm to obtain better images after quantization.

In AFSA, each artificial fish represents a solution to the problem. The problem space corresponds to the living environments of the artificial fishes. The value of the objective function corresponds to the food concentration. The algorithm searches for the optimal solution by continuous iterative calculation in the problem space. The preset number of iteration is the termination condition. AFSA mainly includes the following four major operations: foraging behavior, cluster behavior, rear-end behavior, and random behavior, where foraging behavior is the central operation. And it uses a bulletin board to record the optimal value of artificial fishes in each iteration of the global search process.

In foraging behavior, the artificial fish: k randomly selects a position according to Eq. (9) in the *Visual* neighborhood. The position $\boldsymbol{X}_{k_{\text{next}}}$ is

$$\boldsymbol{X}_{k_{\text{next}}} = \boldsymbol{X}_k + Visual\,\boldsymbol{R}. \tag{9}$$

where $\boldsymbol{R}$ is a random vector between [0,1] generated by the *Rand*() function, and *Visual* represents the field of view. If the food concentration: $Y_k > Y_{k_{\text{next}}}$ at this position [calculated by Eq. (3)], move one step by Eq. (10) to the orientation:

$$\boldsymbol{X}_k^{t+1} = \boldsymbol{X}_k^t + \frac{\boldsymbol{X}_{k_{\text{next}}} - \boldsymbol{X}_k^t}{||\boldsymbol{X}_{k_{next}} - \boldsymbol{X}_k^t||} \cdot Step \cdot \boldsymbol{R}. \tag{10}$$

where *Step* is the step length. Conversely, select another location according to Eq. (11) to determine whether to proceed. If the advance condition cannot be satisfied after a certain number of attempts, proceed to random behavior.

$$\boldsymbol{X}_k^{t+1} = \boldsymbol{X}_k^t + Visual \cdot \boldsymbol{R}. \tag{11}$$

foraging behavior is the basis of convergence, *Visual* selection is appropriate or not directly affects the final result. Experiments show that the reason for the increasing iterations of the algorithm is that the *Visual* value is fixed. In the search process of the optimal solution, there will be a small number of artificial fishes different from the best recorded on the bulletin board. If the value of the given *Visual* is too large, the initial convergence will be fast, but the iterations are increased sharply when the optimal solution is approached, and the number of trials: *TryNumber* will also increase sharply, which affects the execution efficiency of the algorithm. On the contrary, if the *Visual* parameter is set too small, the convergence speed will be slower, the calculation amount will be increased. Moreover, the algorithm will easily fall into the local minimum value and the global optimal solution cannot be obtained. The improved AFSA in this paper is designed by adaptive adjusted *Visual* to deal with the problems.

3.2 Adaptive artificial-fish-swarm algorithm (AAFSA)

3.2.1 Algorithm process

In order to solve the above problems, this paper improves the *Visual* selection of foraging behavior, makes it adaptively change according to the current iteration state, and then proposes adaptive AFSA (AAFSA). The basic idea is that *Visual* is adaptively reduced as the iterations increases. At the same time, set a lower threshold to avoid the later *Visual* being too small to affect the final result. When *Visual* is less than the threshold, it stops decreasing. The set of this threshold needs to be based on the specific problem. The *Visual* is calculated as:

$$\begin{cases} Visual_{p+1} = \lambda Visual_p, & \lambda_p \geq \gamma. \\ Visual_{p+1} = \gamma Visual_0, & \lambda_p < \gamma. \end{cases} \tag{12}$$

where $0 < \lambda < 1$, is the attenuation factor. p is the iterative counter (the maximum is p). $\gamma \in (0,1)$ is the lower threshold. AAFSA algorithm process in each iteration is shown below:

Step 1. Foraging behavior is performed as before [Eq. (9–11)] and updated *Visual* by Eq. (12).
Step 2. Cluster behavior. The state of the *k*th artificial fish is $\boldsymbol{X_k}$, looking for the companions in its *Visual* range. The number of companions is *num*, and the scale of artificial fish is *N*. If *num*/*N* < δ ($0 < \delta < 1$), it means that the nutrition around the companions is rich and the number of fish is appropriate, where δ is the congestion factor. If $Y_k > Y_c$ is satisfied at the same time, move one step by Eq. (13) to the center position $\boldsymbol{X_c}$:

$$\boldsymbol{X}_k^{t+1} = \boldsymbol{X}_k^t + \frac{\boldsymbol{X}_c - \boldsymbol{X}_k^t}{||\boldsymbol{X}_c - \boldsymbol{X}_k^t||} \cdot Step \cdot \boldsymbol{R}. \tag{13}$$

Otherwise, perform Step 1.
Step 3. Rear-end behavior. The state of the *k*th artificial fish is $\boldsymbol{X}_k$, find the best companion $\boldsymbol{X}_{\max}$ in its *Visual* range. Let $Y_k > Y_{\max}$; the number of companions: *num* in the *Visual* range of $\boldsymbol{X}_{\max}$ satisfies the condition of $num/N < \delta$, $(0 < \delta < 1)$. δ is the crowding factor. It indicates that the nutrient is rich around $\boldsymbol{X}_{\max}$ and the number of fish is appropriate; then move one step by Eq. (14) to the orientation of $\boldsymbol{X}_{\max}$:

$$\boldsymbol{X}_k^{t+1} = \boldsymbol{X}_k^t + \frac{\boldsymbol{X}_{\max} - \boldsymbol{X}_k^t}{||\boldsymbol{X}_{\max} - \boldsymbol{X}_k^t||} \cdot Step \cdot \boldsymbol{R}. \tag{14}$$

Otherwise, perform Step 1.

Step 4. Random behavior. The artificial fish executes Eq. (11) in the *Visual* range, selects the next position, and moves to that position.
Step 5. Updating records is performed on the bulletin board, *p*- -.

3.2.2 The analyses of AAFSA

(1) The time cost analyses

I. Time complexity for initializing artificial fishes (amount is *N*) is *O*(*N*), time complexity for initializing other parameters is *O*(1).

II. The bulletin board needs to be initialized once, updated $N - 1$ times. Therefore, the time complexity is *O(N)*;

III. Clustering behavior needs calculating δ needs *N* times, judge once, move once. Artificial fishes are gathered *N* times. Therefore, the time complexity is $O(N^2 + 2N)$;

IV. Rear-end behavior takes *N* times to calculate δ, and *N* times to find the optimum. Moreover, Artificial fishes need to chase for *N* times. Therefore, the time complexity is $O(2N^2 + 2N)$;

V. Foraging behavior needs to test *TryNumber* times, at least once. And artificial fishes need to feed for *N* times. Therefore, the time complexity is $O(N \cdot TryNumber)$;

In a word, since the maximum number of iterations passed by AAFSA is *P*, the time complexity of the algorithm is $O(P \times (3N^2 + N \cdot TryNumber + 6N)) \approx O(PN^2)$. This time cost is acceptable.

(2) The analyses of convergence

AAFSA is convergent to the global optimal solution which can be proved by establishing an absorbing Markov process model.

Definition 1 $\{\boldsymbol{X}(t)\}_{t=0}^{\infty}$ is considered as a random process where $t_1 < t_2 < \ldots < t_n$, $\boldsymbol{X}_1, \boldsymbol{X}_2, .., \boldsymbol{X}_n \in \boldsymbol{E}$, $\boldsymbol{E}$ is the state space. If

$$\begin{aligned} P(\boldsymbol{X}(t_n) = \boldsymbol{X}_n | \boldsymbol{X}(t_{n-1}) = \boldsymbol{X}_{n-1}, \ldots, \boldsymbol{X}(t_1) = \boldsymbol{X}_1) \\ = P(\boldsymbol{X}(t_n) \leq \boldsymbol{X}_n | \boldsymbol{X}(t_{n-1}) = \boldsymbol{X}_{n-1}), \end{aligned}$$

$\{\boldsymbol{X}(t)\}_{t=0}^{\infty}$ is called the Markov process.

Definition 2 Any given some region of Markov process $\{\boldsymbol{X}(t)\}_{t=0}^{\infty}$ and the optimal state space of the region $\boldsymbol{Y}^* \subset \boldsymbol{Y}$. If it satisfies:

$$P(\boldsymbol{X}(t+1) \notin \boldsymbol{Y}^* | \boldsymbol{X}(t) \in \boldsymbol{Y}^*) = 0,$$

$\{\boldsymbol{X}(t)\}_{t=0}^{\infty}$ is called an absorbing Markov process.

Lemma 1 Let AAFSA's optimization process be $\{\boldsymbol{X}(t)\}_{t=0}^{\infty}$, then $\{\boldsymbol{X}(t)\}_{t=0}^{\infty}$ is an absorbing Markov process.

Proving According to the AAFSA optimization process, $\{\boldsymbol{X}(t)\}_{t=0}^{\infty}$ is a discrete-time stochastic process. Because the state $\boldsymbol{X}(t)$ of the artificial fishes in the current iteration is only determined by $\boldsymbol{X}(t-1)$, $\boldsymbol{X}(0)$ can be randomly selected at initialization, so:

$$P(\boldsymbol{X}(t) | \boldsymbol{X}(t-1), \boldsymbol{X}(t-2), \ldots, \boldsymbol{X}(0)) = P(\boldsymbol{X}(t) | \boldsymbol{X}(t-1)),$$

That is, $\{\boldsymbol{X}(t)\}_{t=0}^{\infty}$ has Markov property.

It can be known from the AAFSA clustering, rear end and foraging behavior that when $\boldsymbol{X}(t) \in \boldsymbol{Y}^*$ is the optimal solution space: $\boldsymbol{X}(t+1) \in \boldsymbol{Y}^*$, therefore:

$$P(\boldsymbol{X}(t+1) \notin \boldsymbol{Y}^* | \boldsymbol{X}(t) \in \boldsymbol{Y}^*) = 0,$$

That is, $\{\boldsymbol{X}(t)\}_{t=0}^{\infty}$ is an absorbing Markov process.

Definition 3 Given the absorbing Markov process $\{\boldsymbol{X}(t)\}_{t=0}^{\infty} (\forall \boldsymbol{X}(t) \in \boldsymbol{Y})$ and the optimal state space $\boldsymbol{Y}^* \subset \boldsymbol{Y}$ in a certain region, $\mu(t) = P(\boldsymbol{X}(t) \in \boldsymbol{Y}^*)$ is the probability that it reaches an optimal state in a certain region at a certain time: t. If $\lim_{t->\infty}\mu(t)=1$, then $\{\boldsymbol{X}(t)\}_{t=0}^{\infty}$ has convergence.

Lemma 2 Given AAFSA corresponds to an absorbing Markov process $\{\boldsymbol{X}(t)\}_{t=0}^{\infty}(\forall \boldsymbol{X}(t)\in \boldsymbol{Y})$ and optimal state space $\boldsymbol{Y}^{*}\subset \boldsymbol{Y}$ in a certain region. If:

$$P(\boldsymbol{X}(t)\in \boldsymbol{Y}^{*}|\boldsymbol{X}(t-1)\notin \boldsymbol{Y}^{*})\geq d(t-1)\geq 0$$

And

$$\lim_{t->\infty}\mathop{\varPi}_{i=0}^{t}[1-d(i)]=0,$$

then $\lim_{t->\infty}\mu(t)=1$, where $\mu(t){=}P(\boldsymbol{X}(t)\in \boldsymbol{Y}^{*})$.

Proving According to the full probability formula,

$$\begin{aligned}\mu(t){=}\ [1{-}\mu(t-1)]P(\boldsymbol{X}(t)\in \boldsymbol{Y}^{*}|\boldsymbol{X}(t-1)\ /\\ \in \boldsymbol{Y}^{*})+\mu(t-1)P(\boldsymbol{X}(t)\\ \in Y^{*}|\boldsymbol{X}(t-1)\in \boldsymbol{Y}^{*}),\ (\forall t\\ =1,2,\ldots).\end{aligned}$$

Known by Lemma 1, $\{\boldsymbol{X}(t)\}_{t=0}^{\infty}$ is a Markov process, then:

$$P(\boldsymbol{X}(t)\in \boldsymbol{Y}^{*}|\boldsymbol{X}(t-1)\in \boldsymbol{Y}^{*})=1,$$

that is:

$$\begin{aligned}&\mu(t){=}\ [1{-}\mu(t-1)]P(\boldsymbol{X}(t)\in \boldsymbol{Y}^{*}|\boldsymbol{X}(t-1)\notin \boldsymbol{Y}^{*})+\mu(t-1)\\ &=\ >1{-}\mu(t){=}\ [1{-}P(\boldsymbol{X}(t)\in \boldsymbol{Y}^{*}|\boldsymbol{X}(t-1)\notin \boldsymbol{Y}^{*})]\cdot[1{-}\mu(t-1)]\\ &=\ >1{-}\mu(t)\leq[1-d(t-1)][1{-}\mu(t-1)]=\left[1{-}\mu(0)\mathop{\varPi}_{i=0}^{t-1}[1-d(i)\right]\\ &=\ >\lim_{t->\infty}\mu(t)\geq 1-[1-\mu(0)]\lim_{t->\infty}\mathop{\varPi}_{i=0}^{t-1}[1-d(i)]\\ &\geq 1-[1-\mu(0)]\lim_{t->\infty}\mathop{\varPi}_{i=0}^{t}[1-d(i)]\\ &=\ >\lim_{t->\infty}\mu(t)\geq 1,\end{aligned}$$

Because of the probability $\mu(t)\leq 1$ definitely, so $\lim_{t->\infty}\mu(t){=}1$.

Theorem 1 AAFSA converges to the global optimal solution with probability 1.

Proving Suppose the solution space for the problem is $\boldsymbol{\Omega}$. Let it have m local optimum regions: $\boldsymbol{\Omega}_1$,$\boldsymbol{\Omega}_2$,…,$\boldsymbol{\Omega}_m$. During the AAFSA iteration process, there must be an artificial fish finding a local optimal region. Let it be $\boldsymbol{X}_i$, and the local optimal region is $\boldsymbol{\Omega}_1$. From Lemma 2, $\boldsymbol{X}_i$ converges to the optimal value Y_1 in $\boldsymbol{\Omega}_1$.

When AAFSA conducts clustering and rear-end collision, δ controls the density of the artificial fishes at the target position. When the density exceeds δ, the artificial fishes will move to other areas. Therefore, when $\boldsymbol{\Omega}_1$ is fully loaded, other artificial fishes will search the remaining area to obtain a second local convergence area, set to $\boldsymbol{\Omega}_2$. At the same time, the artificial fishes will also converge to the optimal value Y_2 of $\boldsymbol{\Omega}_2$.

Similarly, the remaining $\boldsymbol{\Omega}_3$, $\boldsymbol{\Omega}_4$,…,$\boldsymbol{\Omega}_m$ have the local optimal solutions: Y_3, Y_4,…,Y_m. In addition, in AAFSA, the *Visual* value in each iteration is computable and does not affect the iterative trend. The bulletin board records the above optimal solution. When the algorithm meets the end condition, the bulletin board records are output, that is $Y_{\text{optimum}}{=}\max\{Y_1,Y_2,\ldots,Y_m\}$.

In summary, AAFSA converges to the global optimal solution with probability 1.

4 Adaptive artificial-fish-swarm-inspired fuzzy C-means algorithm (AAFSA-FCM)

AAFSA-FCM can make good use of the advantages of AAFSA to solve the problem that the FCM is too dependent on the initial value and easy to fall into the local extremum. The algorithm's main idea is: use AAFSA to iteratively calculate the optimal solution of the problem. Then use the solution as the initial clustering center of FCM to perform iterative clustering analysis, divide different types of samples into different cluster classes. The implementation procedure of this algorithm is shown as follows:

Step 1. Some parameters should be set: the number of artificial fishes: N, the moving step: *Step*, the number of trials: *TryNumber*, the crowding factor: δ, the search range: *Visual*, the number of cluster division: c, the fuzzy index: m and other parameters;
Step 2. According to the position of the artificial fishes, comparing the current result with the records on the bulletin board, the superior values are selected to update the information on the bulletin board;
Step 3. The initial values of the samples' cluster center and the objective function are calculated according to Eqs. (6) and (3), and then, fitness evaluation with different samples are performed;
Step 4. Perform AAFSA foraging, clustering, and rear-end behavior are operated in each iteration;
Step 5. The state of the artificial fishes is updated by Eqs. (9)–(11), and *Visual* is adjusted adaptively according to Eq. (12);
Step 6. If the end condition is satisfied, the result is taken as the initial value of the FCM, and the process turns to Step 7 to continue the improved FCM cluster analysis; if not, go to Step 2 to continue the AAFSA, p- -.
Step 7. Using the improved FCM, iterative calculation is performed until the constraints are met, and then the final result will be got.

Combined with the previous algorithm analysis, the total time complexity of AAFSA-FCM is $O(PN^2) + O(nlcf)$, and it is acceptable.

5 Experiments and analyses

5.1 Contrast experiments between FCM and AAFSA-FCM

The experiments selected UCI seeds dataset and Iris dataset for cluster analysis. The samples of UCI seeds dataset have eight-dimensional properties and can be divided into three categories. The sample's amount is 210, and each type contains 70 samples; The Iris dataset has four-dimensional properties, 150 samples, including three types, and each type contains 50 records.

The main parameters involved in AAFSA-FCM should be confirmed firstly. According to Ref. [20, 21], several experiments were carried out with the selected test data set, the optimal values are selected: in FCM, m takes 2, c takes 3; in AAFSA, δ is 0.8, γ is 0.5. As for the λ, which is the key parameter of this algorithm, through the analyses of the relationship between it and iteration times by the experiment results of mean and standard deviation shown in Fig. 1, take it as 0.5.

The clustering results of the UCI seeds dataset are shown in Figs. 2 and 3. The clustering results of the Iris dataset are shown in Figs. 4 and 5. It can be seen that the division effect of FCM is relatively vague and loose, the position of each sample is far away from the cluster center. There is no obvious division boundary between classes, and the correct rate of clustering is relatively lower. This shows that the FCM clustering was very likely not select the optimal clustering center. While the AAFSA-FCM's

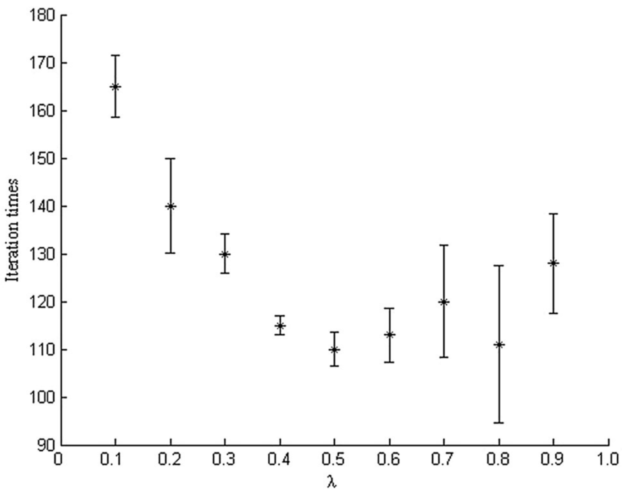

Fig. 1 Relationship between λ and iteration times

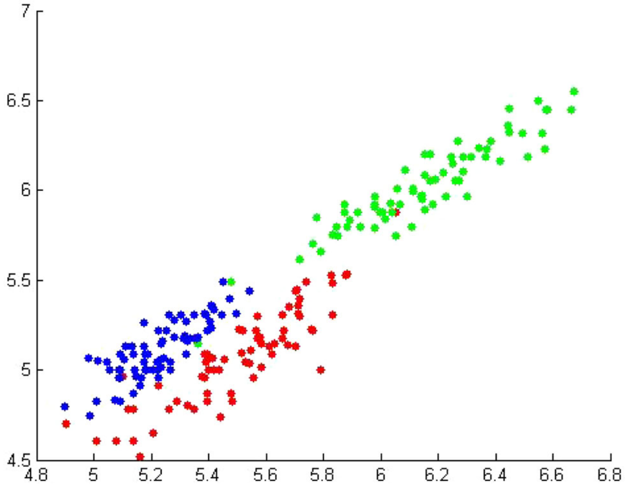

Fig. 2 FCM's results with UCI seeds data set

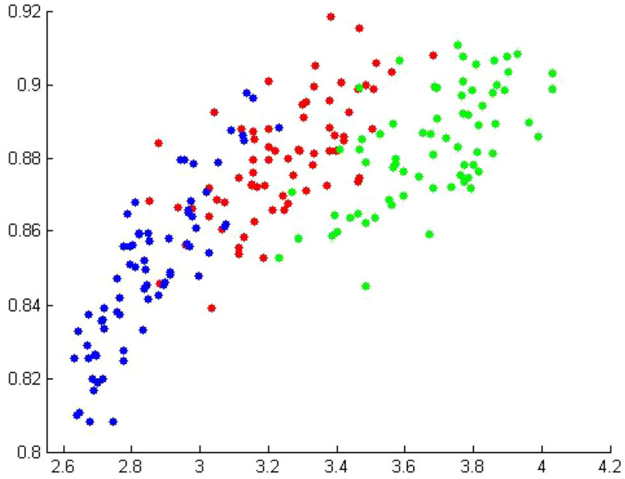

Fig. 3 AAFSA-FCM's results with UCI seeds data set

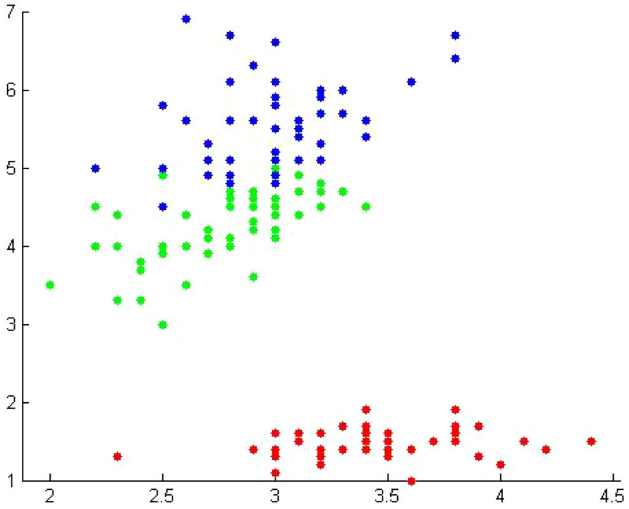

Fig. 4 FCM's results with Iris data set

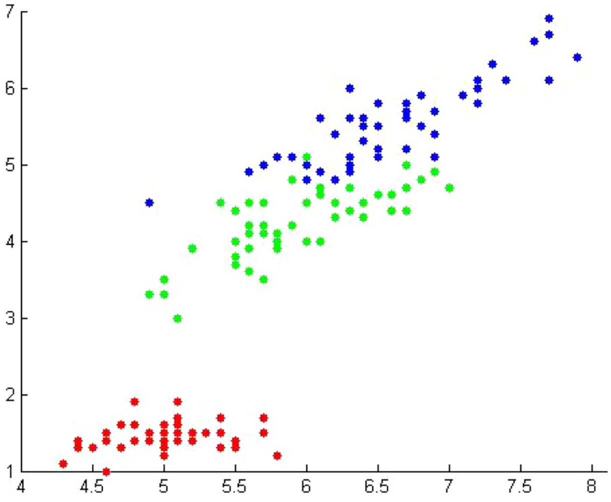

Fig. 5 AAFSA-FCM's results with Iris data set

classification is relatively better. Each class is relatively compact, the division between classes is relatively obvious, and the clustering accuracy is significantly higher than FCM. This indicates that only if the clustering center sought by AAFSA is relatively superior, would the FCM's clustering effect be better.

5.2 AAFSA-FCM's experiments with KDD CUP 1999 data set

In order to further verify the performances of the algorithm with multidimensional data, these experiments used the KDD CUP 1999 data set which is divided into two parts: the first part is the stability experiment based on the AAFSA-FCM, The next is the comparative experiment based on FCM, AFSA-FCM and AAFSA-FCM. Each algorithm was designed as a classifier.

First, data preprocessing is required. Convert a discrete attribute into a continuous attribute, such as a protocol attribute. The transformation rule to be: $TCP \rightarrow 1$, $UDP \rightarrow 2$, $ICMP \rightarrow 3$, etc. Then, randomly selected six groups of samples for experiments: Set one of them to be the training data set with 30,000 records, of which 363 are abnormal records. The remaining five groups are used as test data sets; each group contains 10,000 records, of which 120 are abnormal records. Furthermore, parameter setting process is the same as the previous experiments, and the parameters are set as follows according to the literature [13]: δ, λ, γ are all taken as 0.5, m in the FCM is 2, and c is taken as 5.

In order to evaluate the advantages and disadvantages of these algorithms, the accuracy rate (AR) and the error rate (ER) are defined as follows:

$$\mathrm{DR} = \frac{\text{Number of abnormal samples catched}}{\text{Number of total abnormal samples}}$$

$$\mathrm{FR} = \frac{\text{number of normal samples mistaken for abnormal}}{\text{number of total normal samples}}$$

5.2.1 Stability experiment of AAFSA-FCM

The algorithm stability is very important, and it can be measured by multiple experiments to calculate the mean and standard deviation. In this experiment, six sets of data are selected randomly from the test data set, each group containing 1000 samples. To form $\boldsymbol{X}_{n\times l}$ matrix, repeated five experiments on each data set.

The results are shown in Figs. 6 and 7. As can be seen from the figures, the algorithm gets different classifying performance with different test data, but the fluctuations are within the acceptable range. The main reason for fluctuations is that the algorithm has different cognitive abilities for different categories of data. Moreover, it can be seen that the means of AR and ER of different test data are different under the same parameters of AAFSA-FCM, but the results are satisfactory, and the standard deviation is within reasonable range. This is because the training data and test data selected in this experiment are randomly sampled, the sample's number is limited, and the characteristics of the included abnormal samples are diverse. Overall, the AAFSA-FCM has better classifying performance, especially in terms of stability.

5.2.2 Contrast experiments based on FCM, AFSA-FCM, and AAFSA-FCM

In order to verify the performances of the FCM algorithm after adding AAFSA, the above five sets of test data were tested using the basic FCM, AFSA-FCM, and AAFSA-

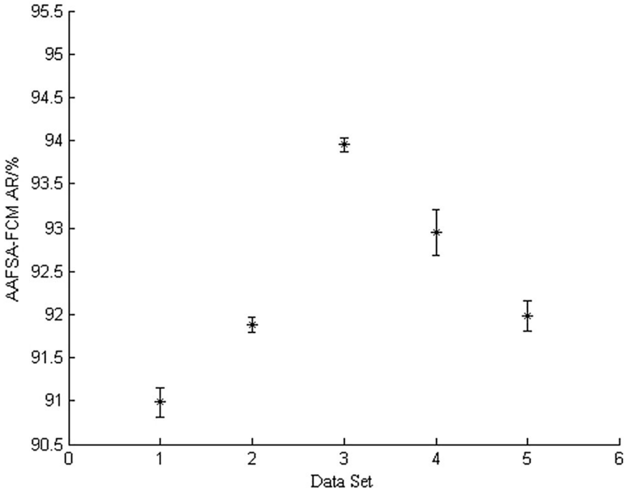

Fig. 6 AAFSA-FCM's AR with different test matrixs

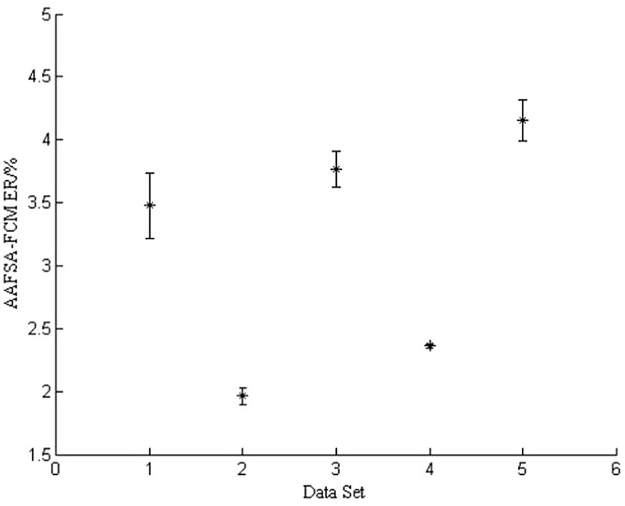

Fig. 7 AAFSA-FCM's ER with different test matrixs

FCM, and the parameters used were the same. The results are shown in Figs. 8, 9, and 10.

Figures 8 and 9 show the comparison of the classifying performances of the three algorithms. It can be seen that under different test data, the AR of AAFSA-FCM is higher than that of FCM and AFSA-FCM, and its ER is lower than the other two algorithms.

Figure 10 shows the comparison of the AR of the three algorithms in one of the test data in different iterations. It can be seen that the AAFSA-FCM can achieve the optimal value when the number of iteration is 85. The other two algorithms require 100–110 iterations to achieve the ideal result, but the results are also worse than the AAFSA-FCM's. That is because AAFSA optimizes AFSA, adaptively adjusts the value of *Visual*, improves the local and global optimization ability, reduces the iterations time, and makes it difficult to fall into local extremum, and thus can obtain the global optimal solution. Then, combining AAFSA with FCM can solve the problems that FCM is too dependent on the initial value and is easy to fall into local optimum.

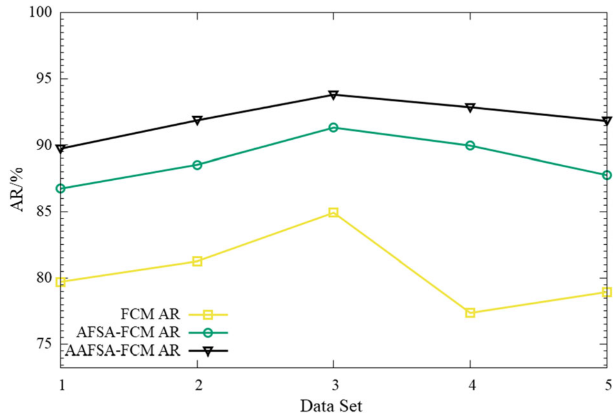

Fig. 8 AR comparison with three algorithms

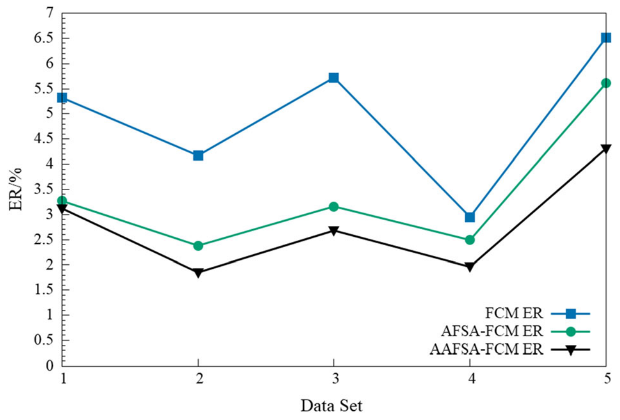

Fig. 9 ER comparison with three algorithms

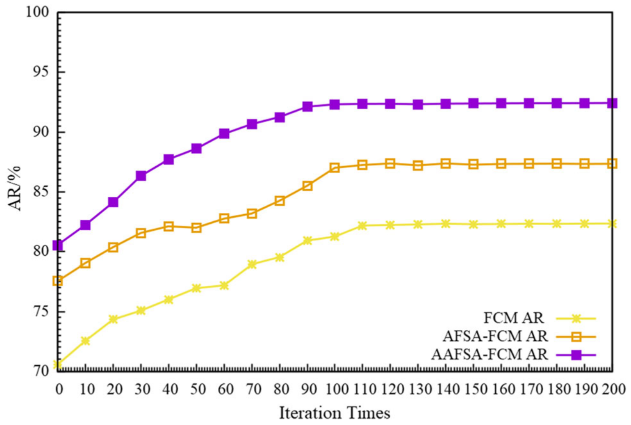

Fig. 10 DR comparison with three algorithms in different iteration cycles

Overall, the above experiments have tested the AAFSA and AAFSA-FCM with different performance evaluations from different perspectives and obtained better test results. It shows that it is feasible and effective to improve AFSA and act on FCM clustering analysis.

6 Conclusion

FCM is a classical clustering analysis algorithm. Meanwhile, AFSA is an effective intelligent optimization algorithm. While both have some defaults need be perfected to adapt to the real complex application environments. In order to solve the problem that AFSA is easy to fall into local optimum when approaching the optimal solution, this paper introduces an adaptive mechanism to AFSA, and adaptively changes the value of *Visual* as iterations of the algorithm, to improve the local and global search ability of AAFSA. Then, because of its intelligent optimization and better robustness, applying it to the FCM to solve the drawbacks that FCM is too dependent on the initial value and easy to fall into the local optimum. Experiments show that the AAFSA-FCM has better classifying performance in clustering and data mining and aggregation. However, the parameter setting needs further mathematical analysis. And the algorithm's effectiveness of practical application needs to be verified by more experiments, especially, in some real environments, such as data mining, deep learning, and abnormal detection in IoT or image processing.

Acknowledgments This work was sponsored by the Natural Science Foundation of Heilongjiang Province (F2018019).

Compliance with ethical standards

Conflict of interest We declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with the work submitted.

References

1. Amirkhani A, Mosavi MR, Mohammadi K et al (2018) A novel hybrid method based on fuzzy cognitive maps and fuzzy clustering algorithms for grading celiac disease. Neural Comput Appl 30(5):1573–1588
2. Yu X, Chu Y, Jiang F et al (2018) SVMs classification based two-side cross domain collaborative filtering by inferring intrinsic user and item features. Knowl-Based Syst 141:80–91
3. Katarya R, Verma OP (2018) Recommender system with grey wolf optimizer and FCM. Neural Comput Appl 30(5):1679–1687
4. Demircan S, Kahramanli H (2018) Application of fuzzy C-means clustering algorithm to spectral features for emotion classification from speech. Neural Comput Appl 29(8):59–66
5. Bharill N, Patel OP, Tiwari A (2018) Quantum-inspired evolutionary approach for selection of optimal parameters of fuzzy clustering. Int J Syst Assur Eng Manag 9(4):875–887
6. Kowkabi F, Keshavarz Ahmad Ghassemian H (2017) Hybrid preprocessing algorithm for endmember extraction using clustering, over-segmentation, and local entropy criterion. IEEE J Sel Top Appl Earth Obs Remote Sens 10(6):2940–2949
7. Zainuddin Z, Ong P (2013) Design of wavelet neural networks based on symmetry fuzzy C-means for function approximation. Neural Comput Appl 23:S247–S259
8. Sengottuvelan P, Prasath N (2017) BAFSA: breeding artificial fish swarm algorithm for optimal cluster head selection in wireless sensor networks. Wireless Pers Commun 94(4):1979–1991
9. Kumar KP, Saravanan B, Swarup KS (2016) Optimization of renewable energy sources in a microgrid using artificial fish swarm algorithm. Energy Proced 90:107–113
10. Liu L, Sun SZ, Yu H et al (2016) A modified fuzzy C-means (FCM) clustering algorithm and its application on carbonate fluid identification. J Appl Geophys 129:28–35
11. Xiao MS, Xiao Z, Wen ZC et al (2017) Improved FCM clustering algorithm based on spatial correlation and membership smoothing. J Electron Inf Technol 39(5):1123–1129
12. Chen HP, Shen XJ, Long JW et al (2017) Fuzzy clustering algorithm for automatic identification of clusters. Acta Electron Sin 45(3):687–694
13. Shanthi I, Valarmathi ML (2013) SAR image despeckling using possibilistic fuzzy C-means clustering and edge detection in bandelet domain. Neural Comput Appl 23:S279–S291
14. Johnson DM, Xiong CM, Corso JJ (2016) Semi-supervised nonlinear distance metric learning via forests of max-margin cluster hierarchies. IEEE Trans Knowl Data Eng 28(4):1035–1046
15. Kannan SR (2013) Effective FCM noise clustering algorithms in medical images. Comput Biol Med 43(2):73–83
16. Azad MAK, Rocha AMAC, Fernandes EMGP (2014) Improved binary artificial fish swarm algorithm for the 0-1 multidimensional knapsack problems. Swarm Evolut Comput 14:66–75
17. Luan XY, Li ZP, Liu TZ (2016) A novel attribute reduction algorithm based on rough set and improved artificial fish swarm algorithm. Neurocomputing 174:522–529
18. Manikandan RPS, Kalpana AM (2017) Feature selection using fish swarm optimization in big data. Cluster Computing. https://doi.org/10.1007/s10586-017-1182-z
19. El-said SA (2015) Image quantization using improved artificial fish swarm algorithm. Soft Comput 19(9):2667–2679
20. Wang LG, Shi QH (2010) Parameters analysis of artificial fish swarm algorithm. Comput Eng 36(24):169–171
21. Ma XM, Liu N (2014) Improved artificial fish-swarm algorithm based on adaptive vision for solving the shortest path problem. J Commun 35(1):1–6