




An improved black hole algorithm designed for K-means clustering method

Chenyang Gao¹ · Xin Yong² · Yue-lin Gao^{1,2}  · Teng Li¹

Received: 30 May 2023 / Accepted: 9 March 2024
© The Author(s) 2024

Abstract

Data clustering has attracted the interest of scholars in many fields. In recent years, using heuristic algorithms to solve data clustering problems has gradually become a tendency. The black hole algorithm (BHA) is one of the popular heuristic algorithms among researchers because of its simplicity and effectiveness. In this paper, an improved self-adaptive logarithmic spiral path black hole algorithm (SLBHA) is proposed. SLBHA innovatively introduces a logarithmic spiral path and random vector path to BHA. At the same time, a parameter is used to control the randomness, which enhances the local exploitation ability of the algorithm. Besides, SLBHA designs a replacement mechanism to improve the global exploration ability. Finally, a self-adaptive parameter is introduced to control the replacement mechanism and maintain the balance between exploration and exploitation of the algorithm. To verify the effectiveness of the proposed algorithm, comparison experiments are conducted on 13 datasets creatively using the evaluation criteria including the Jaccard coefficient as well as the Folkes and Mallows index. The proposed methods are compared with the selected algorithms such as the whale optimization algorithm (WOA), compound intensified exploration firefly algorithm (CIEFA), improved black hole algorithm (IBH), etc. The experimental results demonstrate that the proposed algorithm outperforms the compared algorithms on both external criteria and quantization error of the clustering problem.

Keywords Black hole algorithm · Data clustering · Logarithmic spiral path · Self-adaptive parameter

Introduction

Data clustering known as unsupervised algorithms plays an irreplaceable role in the field of machine learning [1]. Compared with supervised classification algorithms, data clustering algorithms can reveal the internal structure of unknown data regions and provide a new perspective on data. In recent years, data clustering algorithms have been widely applied in data mining [2], image segmentation [3], biomedicine [4], intelligent transportation [5], and many other fields, which shows the significant value of its research. Since the definition of clustering is not completely unified [1], a classic definition is shown here: (1) instances in the

same cluster are relatively close and similar; (2) instances in different clusters are relatively far apart and different; (3) measurements of similarity and dissimilarity should be clear and meaningful. The goal of clustering algorithms is to find such a set of clusters as described. As suggested by Fraley and Raftery in Ref. [6], clustering algorithms can be divided into two categories: hierarchical clustering and partitional clustering. Unlike hierarchical clustering, partitional clustering obtains the final clustering groups by first specifying the initial group and then repeatedly assigning instances to groups until it satisfies the convergence criterion [7]. The most popular partitional methods are the K-means algorithm and the K-centroids algorithm.

Due to its advantages of simple implementation, easy interpretation, and suitability for sparse data [8], the K-means algorithm has been widely studied and applied in many research areas. However, it is also noticed that the K-means algorithm is extremely sensitive to the setting of the initial centroids and does not perform well in some datasets [9]. This dependence on initialization leads to the condition that the performance of the K-means algorithm can be

✉ Yue-lin Gao
gaoyuelin@263.net

¹ The School of Computer Science and Engineering, Xidian University, Taibai Street, Xi'an 710071, Shannxi, China

² Ningxia Province Key Laboratory of Intelligent Information and Data Processing, North Minzu University, Wenchang North Street, Yinchuan 750021, Ningxia Province, China

improved by optimizing the selection of the initial centroids [10]. Since the heuristic algorithms were proposed, they have attracted researchers' attention and developed sufficiently due to their excellent optimization capabilities. These kinds of algorithms are able to search the solutions by following specific rules in a reasonable time, for the reason that they are designed to solve various optimization problems, especially NP-hard problems [11]. Considering such superiority, the heuristic algorithms are also introduced in addressing the problem of optimizing the selection of the initial centroids for the K-means algorithm [12–15].

BHA, proposed by Abdolreza [16] in 2013, is one of the effective swarm intelligence optimization algorithms. Its design mechanism simulates the attraction between a black hole and its surrounding stars in space. BHA has a simple structure and is easy to implement for application problems, meanwhile, it is also fast, efficient, and less affected by hyperparameters. Therefore, it has shown a great performance in many application fields, such as optimization problems [17, 18], feature selection [19], image segmentation [20], gene selection [21], clustering analysis [16, 22, 23], etc. In the past few years, plenty of improved BHAs have been proposed. For instance, Mohammed et al. [24] proposed the gravitational search—black hole algorithm (GSBHA) that combined GSA and BHA, which showed better performance than the original algorithms. In Ref. [25], Yaghoobi et al. modified the formula to search the area near the black hole as well as introduced mutation and crossover operators. After that, the local search ability of BHA was noticed and redesigned to improve its effectiveness [26]. Ibrahim et al. [27] also introduced the white hole local operator for the promotion of the exploitation ability.

Although the studies of the BHA effectively improve its performance, these algorithms also have limitations [28]. The simple design of the structure may cause the algorithm to be trapped in a worse local optimal value with a low probability of escaping in some application problems. In addition, the algorithm may be difficult to control and rely on randomness because there are almost no hyperparameters. Moreover, the quality of the black hole in the swarm has a great influence on the performance of the algorithm, and it is difficult to explore more space once the black hole almost manipulates the update process. At the same time, the trajectories of the stars toward the black hole are not elaborate enough to provide a sufficient exploitation of the solution space. The problems mentioned above may lead to a poor ability to control the balance between exploration and exploitation for the algorithm.

The main goal of this article is to address the problem of selecting the initial centroids for improving the performance of the K-means algorithm. Therefore, an improved BHA, namely self-adaptive logarithmic spiral path black hole algorithm (SLBHA), is proposed here. Figure 1 shows how

the algorithm works, and the details of this algorithm will be presented in the other chapters. The main contribution of this paper is to use the new strategies for overcoming the mentioned drawbacks of the BHA. At the same time, the improved BHA shows an effective way to solve the clustering problem. First, the stars in SLBHA are updated by the logarithmic spiral path and random vector path. In this process, a parameter for controlling the randomness can help to adjust between the two paths, which contributes to the local searching around the black hole. Second, the greedy retention strategy is proposed to help retain better results. Third, SLBHA introduces a replacement mechanism for stars, which improves the diversity of the population, expands the search space of the population, and provides more possibilities for jumping out of the local optimal solution. Finally, an adaptive parameter is added to help control the balance between global and local search procedures by adjusting the replacement mechanism. Then the experimental results based on the standard datasets demonstrate the effectiveness of the proposed methods. The external criteria of the clustering problem are creatively employed in this paper for the analysis.

The rest of this paper is organized as follows. “[Related works](#)” briefly summarizes the heuristic algorithms applied to the clustering problems. “[Preliminaries](#)” presents the basic concepts of the classical BHA model and the conventional K-means algorithm. “[The proposed work](#)” details the proposed algorithm in this paper. “[Results and discussion](#)” evaluates the proposed model through experimental tests and compares it with other selected comparative algorithms. “[Conclusions and future work](#)” gives the conclusions and future research directions of this paper.

Related works

The effectiveness of heuristic algorithms in improving the K-means algorithm is proved in a great of research works. In this section, we give a brief overview of representative literature from the perspective of algorithms.

The genetic algorithm was utilized (GA) for clustering problems by Maulik et al. [29]. After that, the quantum-inspired genetic algorithm for K-means clustering (KMQGA) proposed by Xiao et al. [14] should be noticed. They introduced the Q-bit representation and the concept of quantum computing into their work and changed the length of a Q-bit in KMQGA as a variable quantity. Due to this mechanism, the searching space of this algorithm was extended, which verified its effectiveness on both the simulated datasets and the real datasets. Based on the advantages of the genetic algorithm, Fatahi et al. [30] proposed a combination of the pollination of flowers algorithm and the genetic algorithm

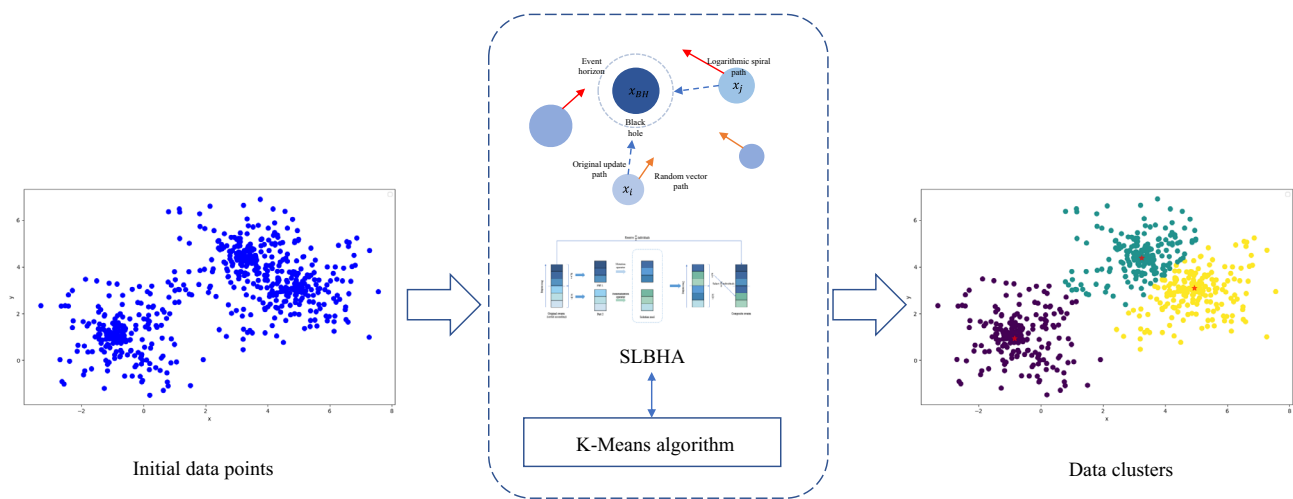


Fig. 1 The working process of the proposed algorithm

(FPAGA). The experimental results demonstrate its effectiveness with greater accuracy and better stability. However, these methods should pay attention to the diversity of the datasets as well as the exploration ability.

As one of the classical intelligent optimization algorithms, the particle swarm optimization (PSO) algorithm was also introduced in clustering problems. In Ref. [31], the author proposed two PSO methods for data clustering. The first algorithm showed how PSO helps to find the centroids of a specified number of clusters and the second one applied the K-means algorithm to seed the initial swarm. The fitness function of the proposed methods in Ref. [31] was novel at that time, but the design of the experiment could be more normalized. Hatamlou et al. [15] hybridized the PSO with a heuristic search algorithm (PSOHS). PSO was used to search for an initial solution to the clustering algorithm and then a heuristic search algorithm was applied to promote the quality of this solution. The superiority of this algorithm over other approaches has been shown in its experiment analysis. Li et al. [32] proposed the adaptive learning PSO to prevent the K-means clustering algorithm from depending on initial cluster centers. Then, the improved KM-ALPSO was proposed for customer segmentation and showed effectiveness and practicability in this task. PSO and its variants showed efficiency and robustness in solving this problem, but their ability to balance exploration and exploitation is questionable.

Ant colony optimization (ACO) methodology was applied to data clustering for data clustering in [33]. Niknam et al. [34] not only noticed the effectiveness of the ACO algorithm but were also interested in the simulated annealing (SA) algorithm. They combined these two algorithms and made use of the SA as a local search in ACO. The experimental results showed a better response and a quicker convergence than ordinary evolutionary methods. In addition, the authors

[35] also proposed a new hybrid evolutionary algorithm that combined the fuzzy adaptive particle swarm optimization (FAPSO), ACO, and K-means algorithms, which was called FAPSO-ACO-K. The performance of this algorithm was much better than the other algorithms for the partitioning clustering problem. The combination of ACO and the K-means algorithm has some different characters compared with the other heuristic algorithms, which should be researched and explored.

The gravitational search algorithm (GSA) is an effective method for searching problem space for the optimal solution and it was combined with the K-means algorithm in the hybrid method proposed by Hatamlou et al. [36]. Their hybrid algorithm, named GSA-KM, helped the K-means algorithm to escape from local optima and increased the convergence speed of GSA. Different from the classical GSA, Dowlatshahi et al. [37] adapted the structure of GSA by a special encoding scheme and presented the grouping GSA (GGSA). The simulation experimental results indicated that this method can effectively be applied to multivariate data clustering. Han et al. [38] introduced a new mechanism that is inspired by the collective response behavior of birds into GSA to add diversity, which was called bird flock GSA (BFGSA). Since the collective response mechanism helped the algorithm explore a wider range of the search space, the performance of BFGSA was much better than the other algorithms. The proposed GSA-based methods mentioned above concentrated on overcoming the drawbacks of the traditional GSA and achieved great results.

In addition, there are still many other heuristic algorithms applied to this problem. For example, Senthilnath et al. [39] used the firefly algorithm (FA) for data clustering and compared it with the other two algorithms. Xie et al. [9] proposed two variants of FA, namely inward intensified exploration FA (IIEFA) and compound intensified exploration FA (CIEFA).

The dispersing mechanism introduced in CIEFA ensured sufficient variance between fireflies for the purpose of increasing search efficiency. However, the time complexity of the FA-based methods is not good enough. A modified bee colony optimization (MBCO) was presented in Ref. [40] and the hybrid algorithms performed better than the compared algorithms. To tackle the cuckoo search (CS) clustering problem, Boushaki et al. [11] extended the CS capabilities using nonhomogeneous update which is inspired by the quantum theory. Zhou et al. [41] used a recently proposed meta-heuristic optimization algorithm, called symbiotic organism search (SOS), to solve the clustering problems. Tawhid et al. [42] proposed a new hybrid swarm intelligence optimization algorithm called monarch butterfly optimization (MBO) algorithm with cuckoo search (CS) algorithm and applied it to the clustering problem. An enhanced whale optimization algorithm (EWOA) is introduced in Ref. [43], while the experiments demonstrated the applicability and feasibility of the enhancements. Almotairi et al. [44] proposed a method named HRSA for the clustering problem, which combined the original Reptile Search Algorithm (RSA) and Remora Optimization Algorithm (ROA).

For all the algorithms mentioned above, it should be emphasized that there is no algorithm that can obtain satisfactory solutions for any application problems and outperform any other algorithms. A useful algorithm should strike a balance between exploitation and exploration abilities and converge to the optimal solution as required. In this paper, we focus on designing the improved BHA for solving the initial optimization problem of the K-means algorithm. The logarithmic spiral path and a replacement mechanism for stars are introduced to improve the searching ability of the algorithm. At the same time, we innovatively design an adaptive parameter to help control the balance between global and local search procedures through adjusting the replacement mechanism. The experiments show that the proposed algorithm is able to converge to the optimal solution mostly and outperform the compared algorithms.

Preliminaries

This section describes briefly the main concepts utilized in the proposed approach, which are the classic BHA and the K-means algorithm.

The classical BHA

The concept of the black hole was first identified by John Michell and Pierre Laplace in the eighteenth century and named by John Wheeler in 1967 [16]. BHA is a population-based intelligent optimization algorithm inspired by the behaviors of black hole and stars. The black holes are formed

when massive stars gravitationally collapse and then create a gravitational field that is so powerful that even light cannot escape. The space around a black hole called the event horizon is the limit that the matter can reach because nothing enters the scope of the event horizon can escape. The radius of the event horizon is called the Schwarzschild radius, which is calculated by the following equation:

$$R = \frac{2GM}{C_l}, \quad (1)$$

where G is the gravitational constant, M is the mass of the black hole, and C_l is the speed of light. Inspired by the above concepts, the BHA is proposed as a novel heuristic algorithm. In the process of searching for the optimal solution, the best agent is set as the black hole while the others are regarded as stars. Then the locations of stars change as they move toward the black hole following the specific tail. Once a better solution is found, the black hole will be replaced with it. In addition, those who are beyond the event horizon of the black hole will be swallowed by it and new stars will be generated to keep the population constant.

Suppose the number of the population is N and the dimension of the optimization problem is D . At the beginning, the agents are randomly initialized in the solution space then the fitness value of each agent is calculated [28]. Take solving the minimum value problem as an example, the agent of the minimum fitness value is set as the black hole. X_i^t represents the i th agent at the t th iteration and X_{BH}^t is the black hole. Then the movements of agents can be formulated by the following equation:

$$X_i^{t+1} = X_i^t + rand \times (X_{BH}^t - X_i^t), \quad (2)$$

where $rand$ represents a random number in the interval $[0,1]$. The formula indicates that the stars in the population are attracted by the black hole and move in the direction of the black hole while the distances of movements are decided by the random number $rand$. However, there exists an event horizon around the black hole. Once a star approaches or exceeds the radius of the event horizon, it will be absorbed by the black hole, and the algorithm will replace it with a new star in the population. The event radius of a black hole here is different with the Eq. (1), which is given by the following equation:

$$R = \frac{f_{BH}}{\sum_i^N f_i}, \quad (3)$$

where f_i and f_{BH} represent the fitness value of the i th agent and the black hole, respectively. At each iteration of the algorithm, the agents in the population are re-evaluated and then compared to the black hole based on the fitness value for

checking whether the black hole needs to be replaced. The process of the BHA will not stop until the convergence condition is met, where the optimal solution is found.

The K-means algorithm

According to the related articles [7], the K-means algorithm starts from a randomly set of centroids, assigns instances to the clusters by the comparison of the distances with the centroids, then recalculates the centroids and iterates until the termination condition is satisfied. The main similarity metric utilized by the K-means clustering algorithm is Euclidean distance [8], that is, the data points in the same cluster are closer and the distances within different clusters are relatively farther based on the Euclidean distance. Specifying the number of clusters that the algorithm needs to split into and the location of the initial centroids are essential for the K-means algorithm. For the next step, data points are assigned to different clusters by comparing the distances to the initial centroids. The resulting clusters may still have large errors at this point, so the centroids should be recalculated and the data points should be reallocated as well. The above process is completed iteratively until the stop condition of the algorithm is met. The selection of the initial centroids has an important influence on the results of the clustering, which means the better initial centroids can greatly improve the performance of the algorithm.

Given k initial centroids, each centroid represents a cluster. There are m instances in the dataset, and the dimension of each instance is d , then the objective function is defined as:

$$J = \sum_{i=1}^k \sum_{j=1}^{|C_i|} \text{distance}(x_j^i, \mu^i), \tag{4}$$

where C_i represents the i th cluster and μ^i is the centroid of it. $|C_i|$ is the total number of the cluster C_i . x_j^i represents the j th instance of the i th cluster and $\text{distance}(x_j^i, \mu^i)$ indicates the distance between x_j^i and μ^i which can be defined by the following equation:

$$\text{distance}(x_j^i, \mu^i) = \sqrt{\sum_{p=1}^d [x_j^i(p) - \mu^i(p)]^2}. \tag{5}$$

After assigning the data points for the first time, the centroids in the algorithm are recalculated using the mean value of the instances belonging to the cluster, which is formulated by the following equation:

$$\mu^i = \frac{1}{|C_i|} \sum_{j=1}^{|C_i|} x_j^i. \tag{6}$$

The stopping condition of the algorithm may be that the objective function value no longer changes with the re-clustering of instances, or the algorithm reaches the number of iterations. The time complexity of the K-means algorithm is proved to be linear, namely $O(Tkmd)$ [8], where T is the number of iterations of the algorithm. It is the linear time complexity that makes K-means a popular and competitive method. Even if the number of instances in the dataset is relatively large, the K-means algorithm has certain advantages compared with other clustering algorithms.

The proposed work

The classical BHA has demonstrated its feasibility and superiority in the data clustering problem at the beginning of its proposal [16]. However, the problem that the algorithm is easy to fall into local optimization limits its application and development. Therefore, this paper designs an improved black hole algorithm, namely SLBHA. The flow chart of the algorithm is shown in Fig. 2. To overcome the drawback of trapping in the local optimum, SLBHA introduces the logarithmic spiral path for improving the local exploitation ability into the BHA and adds a greedy retention strategy. Moreover, a new improved mechanism of global exploration is designed in SLBHA, which greatly expands the search scope of the algorithm and refines the search process. The proposed algorithm will be described in the following sections. To be clearer, the pseudo-code of the algorithm is shown in Algorithm 1.

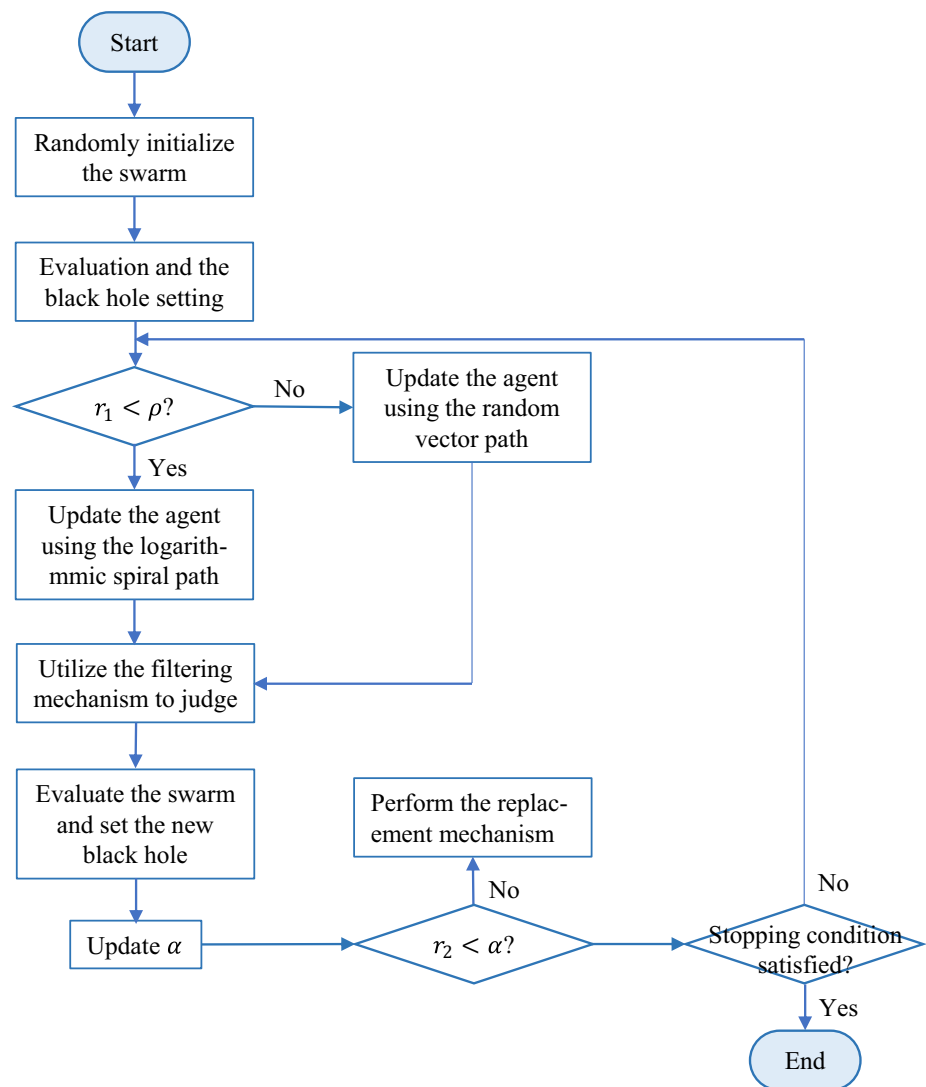
Initialization and representation of agents

Similar to other population-based intelligent optimization algorithms, each individual in the population of the BHA represents a feasible solution that is a set of centroids for the data clustering problem. The agents can be defined as follows:

$$X_i = (c_{i,1}, c_{i,2}, \dots, c_{i,k})^T, \tag{7}$$

where $c_{i,j}$ ($j = 1, 2, \dots, k$) represents the j th centroid in i th agent, it is a d -dimensional vector. k is the number of clusters which is defined before the process of the algorithm. It is obvious that X_i is a matrix of $k \times d$ and a swarm of N agents is a set of N matrices. For initialization, k points for each agent as the set of centroids are randomly selected from the dataset in the proposed methods.

Fig. 2 The flow chart of SLBHA



The proposed improved self-adaptive logarithmic BHA (SLBHA)

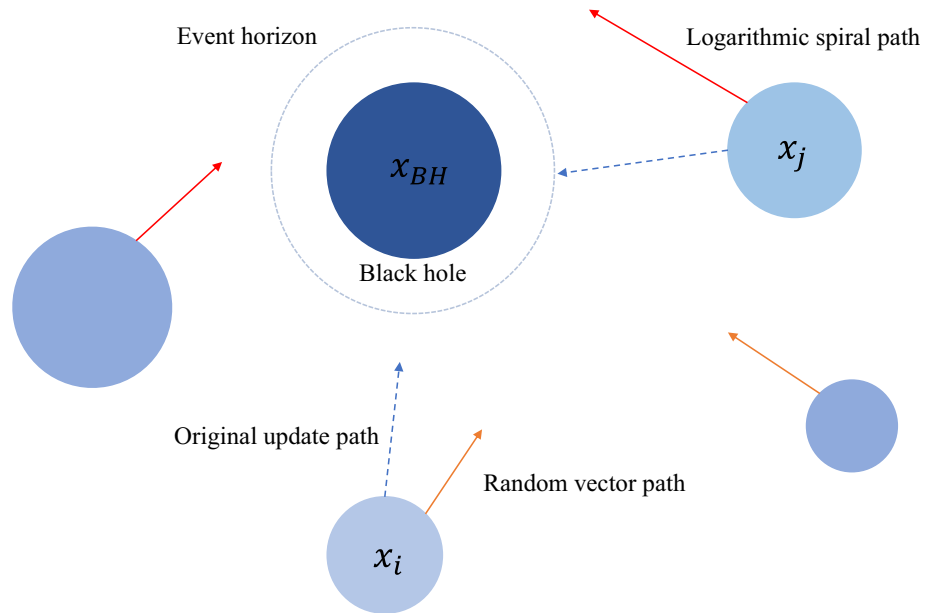
In the classical BHA, the stars are attracted by the black hole and move toward it at a certain distance. It can be seen that the method of deciding the trajectory of stars is usually single and depends a lot on randomness. That is to say, when the black hole is located in the local optimal solution, the stars may be attracted and moved to it with a too-fast convergence at the beginning, which may cause the algorithm to fall into the local optimal solution and be unable to jump out of it. After setting the position of the black hole, the movements of the stars are equivalent to searching the space around the black hole. No matter whether the phase of the algorithm is global exploration or local exploitation, the BHA deals with it with the same method, which means its balance between the two searching modes may not be satisfactory.

Logarithmic spiral path with parameter

The logarithmic spiral path was proposed in the whale optimization algorithm [45] to simulate the helix-shaped movement of humpback whales, which is an effective search method. Used in the firefly algorithm in the literature [46], this path improved the local exploitation ability of the firefly algorithm and verified its effectiveness in experiments. Sharma et al. [47] introduced a logarithmic spiral-based local search strategy and incorporated it with the ABC algorithm to build a new viable algorithm. To the best of our knowledge, there has been no such research that combines this strategy with the BHA until now. In this paper, the logarithmic spiral path is introduced into the BHA and the improved updating formula is given in the following equation:

$$\widehat{X}_i^{t+1} = \begin{cases} X_i^t + (X_{BH}^t - X_i^t) \cdot e^{bl} \cdot \cos(2\pi l), & r_1 < \rho \\ X_i^t + R \cdot (X_{BH}^t - X_i^t), & r_1 \geq \rho \end{cases}, \quad (8)$$

Fig. 3 The movement of the stars



where \widehat{X}_i^{t+1} represents the calculated result of the next generation that may be used for updating. The symbol \cdot is the element-by-element multiplication operation. b is a constant related to the shape of the logarithmic spiral path and l is a random number in $[-1,1]$. r_1 is another random number in $[0,1]$ and is generated in each updating process. ρ is the hyperparameter that controls the update paths of agents, which is in the range $[0,1]$. When the generated random number r_1 is less than ρ , the agent will be updated according to the logarithmic spiral path, otherwise, the other path is chosen. R is a random matrix of size $k \times 1$. The idea of replacing the random number with the random vector to expand the agent search solution space was first proposed by Yaghoobi and Mojallali [25], and then Deeb et al. introduced it in Ref. [22] and made improvements. Inspired by the above literature, this paper also takes use of the random vector as one of the paths. Deeb et al. [22] suggest that the elements in the random vector should be valued within the range of $[0,1.5]$ and the dimension of the vector is d . However, to avoid excessive deviation from the original optimization path, the randomness is reduced using a $k \times 1$ random vector in this paper. The generated numbers of vectors are also adjusted in $[0,1]$. In this way, the algorithm not only expands the search space for the agents but also keeps the degree of randomness at a balanced level. The improved star trajectory selection mechanism is shown in Fig. 3. The path of traditional BHA is represented by the dotted line, which is only for illustration here and not actually used in the algorithm.

In addition, the datasets have been standardized before used in the experiments of this paper, so the value of each dimension of the agent should be between 0 and 1. In the process of star movement, a module to check whether the

star exceeds the value range is necessary. When the value of the position exceeds 1 or is less than 0, it is set as 1 or 0 for calculation.

Greedy retention strategy

In this paper, a cautious greedy retention strategy is used to retain the optimal solution of the agent during its movement. If the position that the star is moving causes the quality of the solution to decline, it will remain stationary, and only when the movement gains more promotion can the star move. In this way, it is possible to ensure that the update process of the stars is always in a progressive state and the enhancement of the algorithm solution is stable. Its mathematical expression is shown in the following equation:

$$X_i^{t+1} = \begin{cases} \widehat{X}_i^{t+1}, & \text{if } f(\widehat{X}_i^{t+1}) < f(X_i^t) \\ X_i^t, & \text{else} \end{cases} \quad (9)$$

With the above measures, the optimization ability of the BHA has been improved to some extent. However, it is insufficient because such a greedy strategy may lead the algorithm to be trapped in a local optimum. Improving the global search ability of the algorithm, especially in the early stage of the algorithm, can help to get rid of the local optimal solution. Therefore, the following strategies are designed based on this idea.

Replacement mechanism of stars

In the classical BHA, the stars are attracted to search only around the black hole and such an approach may lead to

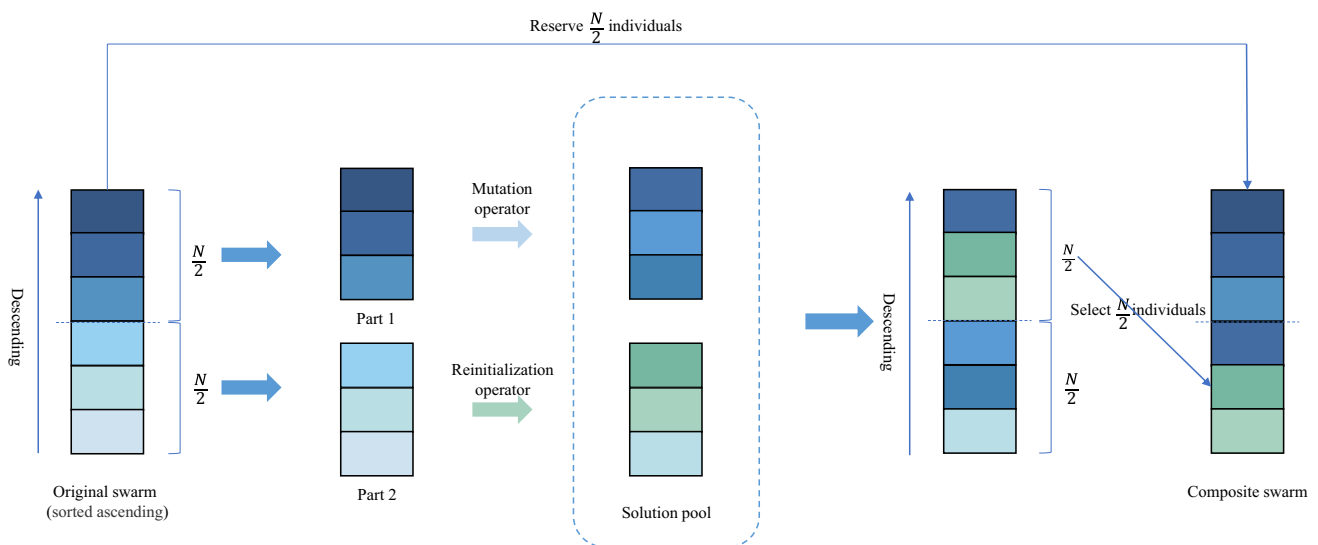


Fig. 4 Diagram of replacement mechanism

the lack of global exploration capability of the algorithm. To further improve the global search capability of the BHA and increase the diversity of the population, a replacement mechanism for stars is proposed in this paper. The basic idea of this mechanism is that the locations of good solutions should be preserved and the worse solutions should be replaced to search more solution space. Figure 4 shows the process of the replacement mechanism. The steps of the operator are given as follows:

Step 1: Sort all the stars (including the black hole) in descending order according to the fitness value. It can be concluded that the top-ranked stars represent the worse quality solutions. Half of the better agents is Part 1 and the other half is Part 2.

Step 2: Choose the stars of Part 1 and execute the mutation operator one by one. The results of mutation are put into the solution pool which can be seen as a set of the solutions that are waiting for selection.

Step 3: The stars in Part 2 are reinitialized and thrown into the solution pool, too.

Step 4: All stars in the solution pool are sorted in descending order. The latter half (the best part) of the solution pool is utilized to replace Part 2 of the original swarm.

The mutation operator in Step 2 can be defined as: (1) Randomly select a position of mutation z (integer in $[1, k]$) of X_{mu} ; (2) Select a data point x_z at random in the dataset; (3) Replace the clustering center point of $X_{mu}[z]$ with x_z to generate a new agent. The mutation operator makes relatively minor changes, which means the mutation of superior agents in the population may be beneficial to obtain better results with less cost. The reinitialization operation is equivalent to

completely replacing agents, and the generation of new individuals in half of the population is conducive to promoting the diversity of the population and expanding its search space.

Self-adaptive parameter

It is not enough to only improve the exploration and exploitation capabilities of the algorithm, because it is more important to maintain the balance between the two search modes. The algorithm needs to update the population at the right time to expand the searching space, so as to avoid falling into the local optimal region. In the earlier iterations of the algorithm, more global searching is necessary, while more attention should be paid to local search in the later iterations. Therefore, an adaptive parameter is introduced in this paper to choose the appropriate timing and maintain the balance. The formula of the self-adaptive parameter α is shown as follows:

$$\alpha = e^{\frac{-2t}{T+1}}, \quad (10)$$

where t is the current iteration number and T is the total number of iterations. Figure 5 is the functional graph of α . It can be seen from the figure that the value of α gradually decreases from 1 to close to 0.1 with the increase of iterations. In each iteration, a random number r_2 is generated in $[0, 1]$. If $r_2 < \alpha$, the replacement mechanism will be performed in this iteration. It is apparent that the probability of performing this operation is higher at the early stage of the algorithm. However, with the increase of the number of iterations, the algorithm needs more local exploration and excessive global searching may affect the convergence speed of the algorithm. Therefore, the probability of performing this operation is relatively reduced.

Algorithm 1 Pseudo-code of SLBHA

Input: Dataset of size $m \times d$, the maximum number of iterations T , the population size N , the value of ρ and the number of clusters k

Output: The optimal solution that the algorithm found

Initialize the population randomly.

Evaluate the swarm by the fitness function and set the best solution as the black hole.

For $t \leftarrow 1$ to T do

For $i \leftarrow 1$ to $(N - 1)$ do

Generate a random number r_1 .

If $r_1 < \rho$ then

Update the agent using the logarithmic spiral path, get \hat{X}_i^{t+1} .

Else

Update the agent using the random vector path, get \hat{X}_i^{t+1} .

Evaluate the new position of the agent, $f(\hat{X}_i^{t+1})$.

If $f(\hat{X}_i^{t+1}) < f(X_i^t)$ then

Replace the current agent X_i^t with \hat{X}_i^{t+1} , i.e., $X_i^{t+1} = \hat{X}_i^{t+1}$.

Else

The current agent doesn't move.

If $f(X_i^{t+1}) < f(X_{BH}^t)$ then

$X_{BH}^t = X_i^{t+1}$

Else

Calculate the event radius of the black hole R .

If $distance(X_i^{t+1}, X_{BH}^t) < R$ then

Replace the current agent with a new star, i.e., $X_i^{t+1} = X_{new}$

Update the parameter α and define a list as the solution pool L_{pool} .

Generate a random number r_2 .

If $r_2 < \alpha$ then

Sort all the agents in population in descending order according to the fitness value.

For $i \leftarrow 1$ to $(N/2)$ do

Randomly initialize a new agent X_{newi}^t , add it to L_{pool} .

For $i \leftarrow (N/2)$ to N do

Perform the mutation operator on the X_i^t , add the generated X_{mui}^t to L_{pool} .

Evaluate all the new agents and sort the agents in L_{pool} in descending order according to the fitness value.

For $i \leftarrow 1$ to $(N/2)$ do

$population[i] = L_{pool}[i]$

Results and discussion

To objectively and comprehensively evaluate and verify the effectiveness of the algorithms proposed in this paper, 13 datasets are selected for experiments. The comparison algorithms used here are listed as follows: K-means [7], K-means + + [48], FC-Kmeans [49], PSO [31], ABC [50, 51], FA

[38], BHA [16], WOA [52, 53], SOS [40], CIEFA [9], IBH [22]. The experiments mentioned above are conducted on an Intel(R) Core (TM) i7-10,700 CPU with 16 GB RAM. The content related to the experiments will be introduced in this section.

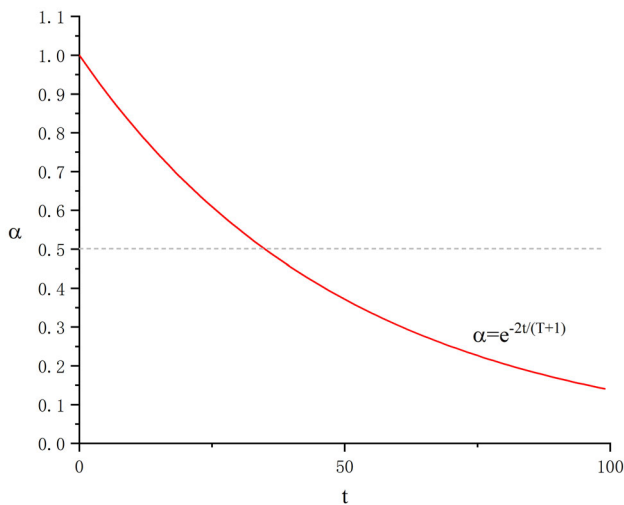


Fig. 5 The curve where the value of α changes with the number of iterations

Table 1 Datasets used in the experiments

Datasets	Features number	Instances number
Contraceptive Method Choice	9	1473
Housing Prices	12	545
Stroke Prediction	12	5110
Heart failure clinical records	13	299
Shill Bidding	13	6321
Early Stage Diabetes Risk Prediction	17	520
Lymphography	18	148
Mobile Price	20	2000
Gender Gap in Spanish WP	21	4746
Steel Plates Faults	32	1941
Ionosphere	34	351
QSAR biodegradation	41	1055
Arcene	10,000	900

Datasets description

Table 1 gives a detailed introduction to the datasets used in this paper, including the number of features and the number of instances of the datasets. The datasets of Stroke prediction, Early-stage diabetes risk prediction, Mobile price, and Housing Prices are selected from the Kaggle website, and the rest are from the UCI data repository. For more details, please access the website of Kaggle (<https://www.kaggle.com/>) and the UCI Repository (<http://archive.ics.uci.edu/ml/index.php>) [54].

The datasets used in this paper have different dimensions and instances, which presents various challenges to the

data clustering algorithm. Usually, the preprocessing process including feature coding and data standardization is required before using the datasets. The main theme in this paper is the data clustering problem, so distance measurement is an important factor [1]. Therefore, the non-numeric data need to be converted to numeric data to facilitate calculation. In addition, the data need to be normalized to eliminate the influence of different scales, which can also indirectly avoid the influence of noise and outliers. Here, the datasets are pre-processed by the methods mentioned above including feature coding, data standardization, and normalization to facilitate the comparison of algorithms, and this step is also essential.

Evaluation criteria

Clustering classifies a dataset with undefined classes according to some specific method, so its evaluation methods are defined differently from that of classification algorithms. The literature [55] gives three methods to evaluate the validity of clustering: external criteria, internal criteria, and relative criteria. External criteria are mainly evaluated by imposing the results of clustering algorithms on a pre-specified dataset structure to validate the clustering solutions. Internal criteria evaluate the internal structure generated by the clustering algorithm. As for relative criteria, they evaluate a structure by comparing it with other methods. External criteria are based on some prior information of the datasets while internal criteria are not dependent on external information [56]. The evaluation criteria used in the experiments of this paper include external criteria and quantization error.

External criteria

Suppose $C = \{C_1, C_2, \dots, C_k\}$ is the set of clusters that the data clustering algorithm generated and $P = \{P_1, P_2, \dots, P_s\}$ is defined structure of the dataset. Consider a pair of data points (x_a, x_b) randomly selected from the dataset, the following terms will be measured:

SS: if x_a and x_b belong to the same cluster of C and the same partition of P .

SD: if x_a and x_b are in the same cluster of C , but in the different partitions of P .

DS: if x_a and x_b are in the different clusters of C , but in the same partition of P .

DD: if x_a and x_b belong to the different clusters of C and the different partitions of P .

Here, a , b , c , and d are utilized to represent the number of SS, SD, DS, and DD. The total number of all pairs of data points in the dataset is M , which means $M = a + b + c + d$. It can be deduced that $M = m(m - 1)/2$, where m is the total number of the dataset mentioned before. Then the indices that measure the similarity between C and P can be defined

as follows:

$$J = \frac{a}{(a + b + c)}, \tag{11}$$

$$FM = \frac{a}{\sqrt{m_1 m_2}} = \sqrt{\frac{a}{a + b} \cdot \frac{a}{a + c}}, \tag{12}$$

where J is the Jaccard coefficient while FM represents Folkes and Mallows index. $m_1 = a/(a + b)$, $m_2 = a/(a + c)$. For the two indices, the higher value indicates the more similar degree of C and P . Here, these two indices are utilized to evaluate the similarity between the obtained clustering results and the original labels of the dataset, so as to compare the effectiveness of the algorithms. To the best of our knowledge, it is the first work that introduces these two criteria into the related studies.

Quantization error

The fitness function is essential for intelligent optimization algorithms. A suitable fitness function can improve the efficiency and performance of the algorithm. Inspired by the literature [31], this paper selects quantization error as the fitness function, which can be formulated as follows:

$$\begin{aligned} \text{fitness} &= f(X_p^t) = f[(c_{p,1}, c_{p,2}, \dots, c_{p,k})] \\ &= \frac{\sum_{i=1}^k \left[\sum_{j=1}^{|C_i|} \frac{\text{distance}(x_j^t, c_{p,i})}{|C_i|} \right]}{k}, \end{aligned} \tag{13}$$

where X_p^t is the p th agent in t th iteration and $c_{p,k}$ is its k th element. The quantization error is also evaluated based on the structure of the clusters to some extent, which can be regarded as an internal criterion. The smaller the fitness value is, the better the solutions searched by the agent are. The indicators used in this paper also include the average fitness value, the best fitness value, the worst fitness value, and the standard deviation.

Experiment settings

In order for each algorithm to exhibit the best performance, the state-of-art algorithms selected for comparison all use the parameters suggested in the original articles, which is shown in Table 2. Among them, algorithms not mentioned in the table generally require no parameters. The parameters of SLBHA are tested and discussed in “Parameter experiment”. For the sake of fairness, the experiments are conducted by running the algorithms 30 times. The maximum number of iterations and agents are set as 100 and 20, and the fitness function of all the algorithms is set as the same one, i.e., the

Table 2 Parameter settings of the experiments

Algorithms	Parameters
PSO	$w = 0.72, c_1 = c_2 = 1.49$
ABC	Limit = 1000
FA	$\beta_0 = 1, \gamma = 1$
CIEFA	$\tau = (1 - t/T_{\text{total}}) \times (1 + \mu), \beta_0 = 1, \gamma = 1, \alpha = 0.2$
SLBHA	$\rho = 0.5, \alpha = e^{-2t/T+1}$

quantization error used in this paper. Since they run on the same standardized datasets, the values of each dimension of agents in every algorithm are in the range of [0,1].

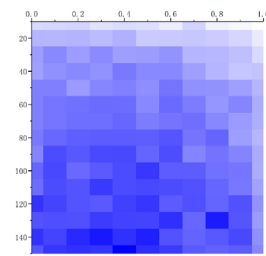
Parameter experiment

In the proposed algorithm, there is a parameter ρ , which controls the selection of the star movement path, and its value may have an important impact on the algorithm’s exploration capability. Therefore, an experiment is designed to analyze which value of this parameter is appropriate for the algorithms. In addition to the introduced parameter ρ , for the swarm intelligence algorithm, the population size N and the number of iterations T also exercise a greater influence on the performance of the algorithm. Therefore, it is necessary to design experiments to explore the values of these three parameters for this paper.

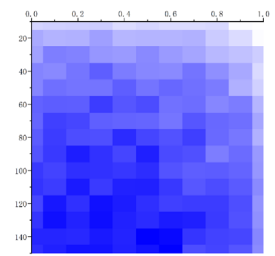
For the purpose of exploring the appropriate value of the parameter ρ while observing its influence on the algorithm and the improvement effect, the first part of the parameter experiments will discuss the settings of ρ and T . In this experiment, ρ is taken within [0, 1], then the experiment selects the values of ρ by 0.1 steps. In the meanwhile, the number of iterations T is set in [100, 150], and the step length is 10. For each parameter combination pair $\langle \rho, T \rangle$, the experiment runs 20 times independently on 10 datasets, and then the mean results of the fitness values are summarized and plotted in Fig. 6.

The abscissa of Fig. 6 represents the value of ρ , with a total of 11 values, while the ordinate represents the value of total iteration numbers. It can be seen from the figure that the impact of the iteration numbers on the accuracy of the algorithm is obvious. In almost all datasets used here, the figure is gradually deepened from top to bottom. Especially in the Ionosphere dataset, the optimal solution can almost always be found when $t \geq 50$, so its color is deeper and more average than the others. In horizontal comparison, ρ values have different effects on the performance of the algorithm. For all used datasets, it can be seen that most of the deeper positions are located in [0.4,0.6] expect some outliers, which also indicates that the influence of parameter ρ on the algorithm

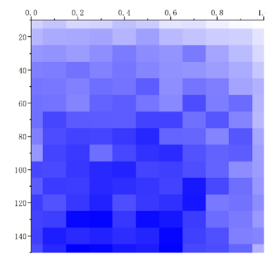
Fig. 6 Mean fitness values of part of SLBHA for different parameter combinations of parameter ρ and the number of iterations T on datasets



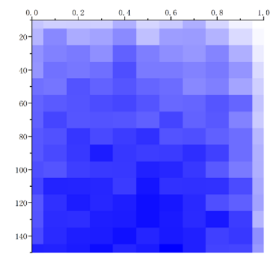
(a) Contraceptive Method Choice



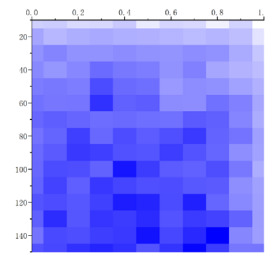
(b) Housing Prices



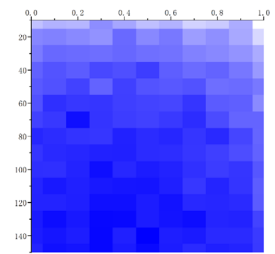
(c) Stroke Prediction



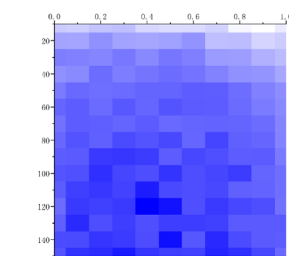
(d) Heart failure clinical records



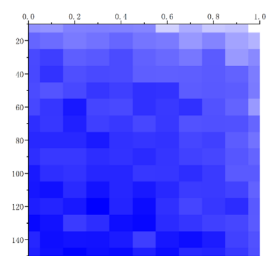
(e) Shill Bidding



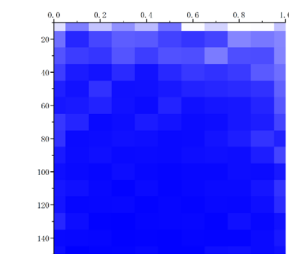
(f) Early Stage Diabetes Risk Prediction



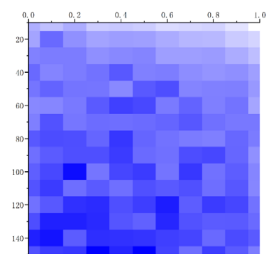
(g) Gender Gap in Spanish WP



(h) Steel Plates Faults



(i) Ionosphere



(j) QSAR biodegradation



Fig. 7 Mean fitness values of SLBHA for different population sizes N on datasets

is not always stable. Considering the meaning of parameter ρ , it represents the balance degree of the two paths in the SLBHA algorithm, and the random number used for control is generated in $[0,1]$. Therefore, only the logarithmic spiral path is effective when $\rho = 1$, and only the vector update path is effective when $\rho = 0$. Comparing the color depth of the left end ($\rho = 0$), the right end ($\rho = 1$), and the middle part ($\rho \in [0.4, 0.6]$), the middle part is deeper than the left and right ends, indicating that too large or small of the value ρ is not suitable for the algorithm. From another perspective, it also proves that the combination of the two paths is much better than using a single path and the proportion of these two paths should be relatively balanced to achieve the optimal performance of the algorithm. According to Fig. 6, it can be found that when $\rho = 0.5$ and $T = 100$, the algorithm can achieve satisfactory performance in all datasets while saving time and space resources.

The second part of the parameter experiments is based on the first part mentioned above, discussing the influence of the population size N on SLBHA. ρ and T are set as 0.5 and 100, respectively. The population size N is set within $[10, 50]$ and the step size is set as 5. For each value of N , SLBHA runs 20 times independently, and the average fitness values are obtained and plotted as a line graph as shown in Fig. 7.

As shown in Fig. 7, the abscissa represents different population sizes while the ordinate represents the fitness values. It is shown in Fig. 7 that the average fitness values of SLBHA decrease with the expansion of population size, but increase slightly to some extent after a certain threshold. There are two or more fluctuation trends shown on all the lines in the figure. This phenomenon first indicates that the efficiency of the algorithm increases with the expansion of the population size. However, after a certain critical value, this trend slows down and develops in the opposite direction, which is, the unsuitably over-large population will lead to a slight decline

in algorithm performance. In conclusion, the impact of population size N on the algorithms is fluctuating and complex and it should be selected thoughtfully. To prevent the fluctuation effect of algorithm performance decrease caused by a large population, the population size is set as 20 in this paper after comprehensive consideration.

To sum up, the parameter experiments, on the one hand, provide a reference for parameter settings in this paper. In a comprehensive view, the parameters are set as follows: $\rho = 0.5$, $T = 100$, $N = 20$. On the other hand, these experiments can also reflect the effectiveness of the improvement measures designed in this paper.

Analysis of the proposed strategies

In this section, the proposed strategies are analyzed and their effectiveness is verified through experiments. According to the beginning design ideas, the strategies of SLBHA shown in “[Analysis of the proposed strategies](#)” can be divided into two groups. Here, the star replacement mechanism and self-adaptive parameter strategy can be regarded as a group, while the other two strategies form another group. The improvement strategies within a group are interrelated and indivisible. Therefore, the part of improved SLBHA (only retained the logarithmic spiral path with parameter and the greedy retention strategy) and the completely improved SLBHA are compared with the traditional BHA. Table 3 shows the comparison results of the two algorithms, where LBHA represents the part of improved BHA.

From the previous descriptions of the algorithm, it can be seen that the LBHA adds a logarithmic spiral path and random control parameter for adjustment as well as the greedy retention strategy compared to the classical BHA. The complete SLBHA adds a replacement mechanism of stars and uses a self-adaptive parameter for regulation, which can expand the global searching space of the agents and increase the population diversity, resulting in a better performance of the search capability. The experiments conducted here show the validity of the above strategies. As for the mean value, SLBHA performs better than LBHA on all datasets except the QSAR biodegradation dataset in terms of the average of the fitness values, and both algorithms perform better than BHA. When it comes to the minimum and maximum of the fitness values, SLBHA performs better than LBHA on more than 2/3 of the datasets. The possible reason is that SLBHA may have slightly insufficient local exploration for some datasets, despite its extended searching range in comparison. In addition, these algorithms are inherently heuristic, and the instability of the results due to randomness is not unusual. As can also be seen in Table 9, the LBHA algorithm has better standard deviation results than SLBHA on more than half of the datasets, which also indicates that the stability of these two algorithms is similar. However, the results

Table 3 Comparison between BHA and proposed algorithms

Datasets	Best			Worst			Mean		
	BHA	LBHA	SLBHA	BHA	LBHA	SLBHA	BHA	LBHA	SLBHA
Contraceptive Method Choice	0.6469	0.503	0.4389	0.7476	0.6743	0.7444	0.7071	0.6145	0.5652
Housing Prices	0.8349	0.5695	0.5304	1.0234	0.8358	0.7902	0.9176	0.7203	0.6386
Stroke Prediction	0.8072	0.6971	0.6615	0.8465	0.8348	0.8357	0.8253	0.7709	0.7394
Heart failure clinical records	0.9868	0.7097	0.7381	1.0842	0.9659	0.9617	1.0376	0.8549	0.8472
Shill Bidding	0.7768	0.6979	0.6721	0.8168	0.8092	0.7774	0.8041	0.7653	0.7439
Early Stage Diabetes Risk	1.1503	1.0604	1.0244	1.7939	1.4906	1.2526	1.5298	1.2148	1.1596
Lymphography	1.2889	0.756	0.7752	1.4545	1.2173	1.1144	1.3562	1.0507	0.9223
Mobile Price	1.4627	1.1454	1.1558	1.5237	1.4344	1.4384	1.4892	1.3651	1.2792
Gender Gap in Spanish WP	0.7106	0.5867	0.6322	0.742	0.7237	0.7135	0.7234	0.6879	0.684
Steel Plates Faults	0.9995	0.7793	0.7493	1.1358	1.0422	1.2408	1.0671	0.9799	0.9553
Ionosphere	0.7086	0.6969	0.696	1.2347	0.8648	0.7728	0.9512	0.7129	0.7094
QSAR biodegradation	0.5646	0.4049	0.5423	0.6799	0.635	0.6421	0.6355	0.5967	0.6031
Arcene	16.1197	11.642	11.6083	21.117	18.5873	12.5475	19.1976	12.7298	11.9341
Datasets	Standard deviation			Jaccard coefficient			FM values		
	BHA	LBHA	SLBHA	BHA	LBHA	SLBHA	BHA	LBHA	SLBHA
Contraceptive Method Choice	0.0183	0.0466	0.0806	0.2415	0.2939	0.3113	0.3901	0.4698	0.5061
Housing Prices	0.0431	0.0706	0.0739	0.3223	0.4106	0.4541	0.4944	0.5829	0.6303
Stroke Prediction	0.0094	0.0450	0.0354	0.5219	0.6907	0.7740	0.7062	0.8172	0.8716
Heart failure clinical records	0.0247	0.0501	0.0550	0.3827	0.5201	0.5317	0.5537	0.7023	0.7157
Shill Bidding	0.0091	0.0289	0.0251	0.4628	0.5961	0.6642	0.6485	0.7470	0.7974
Early Stage Diabetes Risk	0.1362	0.0841	0.0406	0.5059	0.5254	0.5250	0.6746	0.7084	0.7146
Lymphography	0.0335	0.1075	0.1060	0.4017	0.4947	0.4992	0.5730	0.6940	0.7026
Mobile Price	0.0160	0.0665	0.0667	0.1515	0.1891	0.2124	0.2634	0.3325	0.3871
Gender Gap in Spanish WP	0.0075	0.0324	0.0177	0.3560	0.4364	0.4546	0.5263	0.6112	0.6283
Steel Plates Faults	0.0245	0.0655	0.1161	0.4140	0.4585	0.4776	0.5886	0.6391	0.6609
Ionosphere	0.1714	0.0296	0.0185	0.4950	0.5383	0.5384	0.6831	0.7330	0.7331
QSAR biodegradation	0.0187	0.0484	0.0250	0.4668	0.4995	0.5128	0.6420	0.6790	0.6934
Arcene	0.9157	1.3857	0.2107	0.3802	0.4932	0.4973	0.5535	0.6964	0.7018

Bold texts represent the best values under the indicators (larger values are better under Jaccard coefficient and FM values, smaller values are better for the others)

of BHA are more stable than those of the two algorithms on 10 datasets, which shows that though the proposed algorithm improves the overall performance, its stability needs to be further improved to some extent. The calculating results of external indicators (including the Jaccard coefficient and FM values) of SLBHA in all datasets are higher than LBHA, which means that the distribution of clusters obtained by SLBHA is more similar to that of labels in the original datasets. In addition, these two algorithms also outperform BHA in these two indicators.

In summary, although BHA has higher stability, SLBHA and LBHA are better than BHA in algorithm performance, that is, the improvement measures added in LBHA are

effective. Furthermore, the SLBHA outperforms the LBHA showing the replacement mechanism of stars and the self-adaptive parameter contributes to the convergence of the proposed algorithm.

Results analysis and discussion

As mentioned above, several metrics are used to evaluate the experiments conducted in this paper. In this section, the experimental results will be analyzed and discussed. For comparison purposes, the best, worst, mean, and standard deviation of the 30 experimental fitness values are given in Tables 4, 5, 6 and 7. Tables 8 and 9 give the means of Jaccard

Table 4 Comparison in terms of the best fitness value

Datasets	K-means	K-means ++	FC-Kmeans	PSO	ABC	FA	BHA	WOA	SOS	IFA	IBHA	SLBHA
Contraceptive Method Choice	0.6953	0.8020	0.7818	0.6562	0.5074	0.6963	0.6469	0.7109	0.5647	0.6627	0.5491	0.4389
Housing Prices	0.9152	1.1105	1.0120	0.8702	0.6471	0.8137	0.8349	0.8542	0.6604	0.8939	0.6379	0.5304
Stroke Prediction	0.8174	0.9957	0.8502	0.8096	0.7263	0.8548	0.8072	0.7829	0.7250	0.8152	0.7009	0.6615
Heart failure clinical records	1.0083	1.2615	1.0698	0.8963	0.7475	1.0059	0.9868	1.0445	0.7152	1.0077	0.8027	0.7381
Shill Bidding	0.8116	0.9256	0.8377	0.7799	0.7275	0.8098	0.7768	0.7161	0.6789	0.8090	0.7466	0.6721
Early Stage Diabetes Risk Prediction	1.6275	1.9316	1.8128	1.2008	0.9915	1.5883	1.1503	1.1492	1.1507	1.4242	1.1412	1.0244
Lymphography	1.3232	1.5797	1.1603	1.3173	0.9887	1.3921	1.2889	1.1775	0.7775	1.1866	0.9234	0.7752
Mobile Price	1.4173	1.8533	1.4430	1.4108	1.1774	1.7973	1.4627	1.5380	1.1613	1.4166	1.2652	1.1558
Gender Gap in Spanish WP	0.7048	0.8294	0.8047	0.7023	0.6264	0.7512	0.7106	0.7400	0.6443	0.7048	0.6472	0.6322
Steel Plates Faults	1.1333	1.3998	1.0626	1.0417	0.7676	1.0561	0.9995	1.2167	0.7688	1.1083	0.9854	0.7493
Ionosphere	1.1837	0.8041	1.1894	0.7121	0.7015	0.7235	0.7086	0.7049	0.7054	0.7236	0.7013	0.6960
QSAR biodegradation	0.6462	0.7124	0.6592	0.6120	0.6090	0.6709	0.5646	0.6566	0.4706	0.6370	0.5618	0.5423
Arcene	18.0789	23.2504	12.1039	17.4467	12.0093	16.2024	16.1197	17.4777	11.7031	14.8824	11.8490	11.6083

Bold texts represent the minimum values

Table 5 Comparison in terms of the worst fitness value

Datasets	K-means	K-means ++	FC-Kmeans	PSO	ABC	FA	BHA	WOA	SOS	IFA	IBHA	SLBHA
Contraceptive Method Choice	0.7428	1.2435	1.0447	0.7573	0.7107	0.8537	0.7476	0.8233	0.7215	0.7640	0.7595	0.7444
Housing Prices	0.9901	1.4803	1.2554	0.9709	0.9183	1.1579	1.0234	1.1309	0.9230	0.9717	0.9121	0.7902
Stroke Prediction	0.8966	1.3872	1.1635	0.8944	0.8548	0.9541	0.8465	0.9761	0.7846	0.9307	0.8463	0.8357
Heart failure clinical records	1.0894	1.5729	1.2658	1.0692	1.0989	1.1483	1.0842	1.2294	0.9482	1.0838	1.0530	0.9617
Shill Bidding	0.8643	1.4263	1.1251	0.8133	0.8197	0.9495	0.8168	0.9733	0.7061	0.8378	0.8250	0.7774
Early Stage Diabetes Risk Prediction	1.7217	2.4786	2.1076	1.6541	1.6237	2.1878	1.7939	1.8955	1.2564	1.6355	1.6256	1.2526
Lymphography	1.3596	2.1481	1.6161	1.3520	1.4109	1.6016	1.4545	1.5670	1.1350	1.5430	1.3682	1.1144
Mobile Price	1.4567	1.9851	1.6467	1.4948	1.5184	1.8897	1.5237	1.7349	1.4612	1.4619	1.5182	1.4384
Gender Gap in Spanish WP	0.7203	1.2523	0.9650	0.7166	0.7382	0.8228	0.7420	0.8462	0.7346	0.7187	0.7620	0.7135
Steel Plates Faults	1.3576	2.0393	1.6261	1.2598	1.3300	1.5745	1.1358	1.5345	1.0992	1.3575	1.2799	1.2408
Ionosphere	1.1861	2.7231	1.9156	1.1653	1.1908	1.2694	1.2347	1.3401	0.8115	1.1787	1.1966	0.7728
QSAR biodegradation	0.7625	1.2754	1.2488	0.7320	0.6824	0.7366	0.6799	0.8413	0.6677	0.7614	0.6798	0.6421
Arcene	19.2952	32.4633	28.4894	19.2842	19.3214	25.2394	21.1170	23.7586	13.5395	19.2952	19.1146	12.5475

Bold texts represent the minimum values

Table 6 Comparison in terms of the average fitness value

Datasets	K-means	K-means ++	FC-Kmeans	PSO	ABC	FA	BHA	WOA	SOS	IFA	IBHA	SLBHA
Contraceptive Method Choice	0.7094	0.9636	0.8814	0.6969	0.6276	0.7654	0.7071	0.7747	0.6314	0.7122	0.6774	0.5652
Housing Prices	0.9356	1.2551	1.1157	0.9246	0.8024	1.0160	0.9176	1.0395	0.7967	0.9362	0.8026	0.6386
Stroke Prediction	0.8335	1.1235	0.9632	0.8309	0.8027	0.8848	0.8253	0.9207	0.7517	0.8379	0.8091	0.7394
Heart failure clinical records	1.0478	1.3648	1.1432	1.0298	0.9289	1.1021	1.0376	1.1581	0.8847	1.0431	0.9244	0.8472
Shill Bidding	0.8204	1.1128	0.9848	0.8045	0.7753	0.8540	0.8041	0.8636	0.6953	0.8131	0.7887	0.7439
Early Stage Diabetes Risk Prediction	1.6378	2.2761	1.9562	1.4988	1.4074	1.8828	1.5298	1.3786	1.2035	1.4981	1.3300	1.1596
Lymphography	1.3479	1.7806	1.4465	1.3365	1.1784	1.4953	1.3562	1.3581	0.9504	1.3353	1.1747	0.9223
Mobile Price	1.4363	1.9081	1.5456	1.4402	1.3841	1.8381	1.4892	1.6115	1.3439	1.4349	1.4352	1.2792
Gender Gap in Spanish WP	0.7081	0.9777	0.8875	0.7071	0.7120	0.7822	0.7234	0.7828	0.7035	0.7069	0.7201	0.6840
Steel Plates Faults	1.2296	1.7264	1.4226	1.1669	1.0802	1.3891	1.0671	1.4197	1.0295	1.2228	1.0741	0.9553
Ionosphere	1.1856	1.7721	1.5814	0.9452	0.9911	1.0848	0.9512	1.0918	0.7384	1.1002	0.8947	0.7094
QSAR biodegradation	0.6566	0.9396	0.8912	0.6375	0.6476	0.7029	0.6355	0.7174	0.6280	0.6523	0.6429	0.6031
Arcene	18.9428	26.6875	22.5393	18.9621	14.8229	23.2779	19.1976	22.0135	12.5119	18.5089	14.0394	11.9341

Bold texts represent the minimum values

Table 7 Comparison in terms of the standard deviation

Datasets	K-means	K-means ++	FC-Kmeans	PSO	ABC	FA	BHA	WOA	SOS	IFA	IBHA	SLBHA
Contraceptive Method Choice	0.0132	0.0928	0.0627	0.0204	0.0513	0.0280	0.0183	0.0309	0.0391	0.0220	0.0481	0.0806
Housing Prices	0.0210	0.0804	0.0686	0.0215	0.0663	0.0679	0.0431	0.0516	0.0610	0.0208	0.0795	0.0739
Stroke Prediction	0.0178	0.1027	0.0697	0.0138	0.0302	0.0262	0.0094	0.0366	0.0150	0.0238	0.0323	0.0354
Heart failure clinical records	0.0206	0.0714	0.0438	0.0335	0.0820	0.0262	0.0247	0.0382	0.0536	0.0200	0.0585	0.0550
Shill Bidding	0.0197	0.1140	0.1070	0.0049	0.0239	0.0257	0.0091	0.0599	0.0069	0.0078	0.0213	0.0251
Early Stage Diabetes Risk Prediction	0.0206	0.1336	0.0785	0.1256	0.1392	0.1226	0.1362	0.2054	0.0208	0.0501	0.1021	0.0406
Lymphography	0.0074	0.1326	0.1056	0.0095	0.1141	0.0499	0.0335	0.1046	0.0783	0.0559	0.1191	0.1060
Mobile Price	0.0080	0.0332	0.0496	0.0154	0.0797	0.0222	0.0160	0.0548	0.0720	0.0107	0.0587	0.0667
Gender Gap in Spanish WP	0.0054	0.1039	0.0464	0.0048	0.0237	0.0199	0.0075	0.0258	0.0193	0.0041	0.0216	0.0177
Steel Plates Faults	0.0742	0.1463	0.1459	0.0942	0.0925	0.1328	0.0245	0.0811	0.0863	0.0690	0.0725	0.1161
Ionosphere	0.0010	0.4477	0.2472	0.1666	0.1854	0.1626	0.1714	0.1764	0.0293	0.1453	0.1979	0.0185
QSAR biodegradation	0.0312	0.1302	0.1749	0.0184	0.0163	0.0176	0.0187	0.0421	0.0383	0.0350	0.0233	0.0250
Arcene	0.4701	1.9635	3.1772	0.5075	1.9510	2.0591	0.9157	1.7752	0.4191	0.9927	2.2591	0.2107

Bold texts represent the minimum values

Table 8 Comparison in terms of the Jaccard coefficient

Datasets	K-means	K-means ++	FC-Kmeans	PSO	ABC	FA	BHA	WOA	SOS	IFA	IBHA	SLBHA
Contraceptive Method Choice	0.2317	0.2516	0.2511	0.2403	0.3020	0.2626	0.2415	0.2530	0.3037	0.2404	0.2696	0.3113
Housing Prices	0.3593	0.4294	0.4539	0.3561	0.3850	0.3704	0.3223	0.3555	0.4358	0.3718	0.3997	0.4541
Stroke Prediction	0.5243	0.5215	0.6419	0.5098	0.7490	0.5152	0.5219	0.5422	0.6715	0.5338	0.5960	0.7740
Heart failure clinical records	0.3677	0.3724	0.4413	0.3890	0.5184	0.3780	0.3827	0.3819	0.4760	0.3732	0.5075	0.5317
Shill Bidding	0.4593	0.4937	0.5657	0.4559	0.6490	0.4704	0.4628	0.5019	0.6585	0.4558	0.5445	0.6642
Early Stage Diabetes Risk Prediction	0.4708	0.4240	0.4757	0.5166	0.5190	0.4634	0.5059	0.5231	0.5190	0.4617	0.5059	0.5250
Lymphography	0.3663	0.3867	0.4480	0.3597	0.4847	0.3841	0.4017	0.4451	0.4851	0.3631	0.4715	0.4992
Mobile Price	0.1448	0.1523	0.1878	0.1467	0.1883	0.1582	0.1515	0.1622	0.2160	0.1464	0.1873	0.2124
Gender Gap in Spanish WP	0.4278	0.3357	0.3628	0.4384	0.4473	0.3804	0.3560	0.3959	0.4048	0.4354	0.4154	0.4546
Steel Plates Faults	0.3779	0.3722	0.4030	0.3938	0.4388	0.4066	0.4140	0.3897	0.4271	0.3801	0.4419	0.4776
Ionosphere	0.4319	0.4693	0.4678	0.4802	0.4771	0.4626	0.4950	0.4606	0.4971	0.4614	0.4882	0.5384
QSAR biodegradation	0.4534	0.4684	0.4814	0.4566	0.4934	0.4703	0.4668	0.4791	0.4995	0.4498	0.4729	0.5128
Arcene	0.3729	0.3853	0.3926	0.3725	0.4900	0.4250	0.3802	0.4067	0.4432	0.3679	0.4623	0.4973

Bold texts represent the maximum values

Table 9 Comparison in terms of the FM values

Datasets	K-means	K-means ++	FC-Kmeans	PSO	ABC	FA	BHA	WOA	SOS	IFA	IBHA	SLBHA
Contraceptive Method Choice	0.3769	0.4066	0.4051	0.3888	0.4851	0.4223	0.3901	0.4082	0.4962	0.3909	0.4340	0.5061
Housing Prices	0.5392	0.6040	0.6235	0.5349	0.5556	0.5453	0.4944	0.5297	0.6093	0.5518	0.5727	0.6303
Stroke Prediction	0.7067	0.7053	0.7847	0.6970	0.8559	0.7010	0.7062	0.7188	0.8047	0.7125	0.7561	0.8716
Heart failure clinical records	0.5382	0.5431	0.6154	0.5604	0.7011	0.5489	0.5537	0.5531	0.6536	0.5439	0.6879	0.7157
Shill Bidding	0.6469	0.6731	0.7218	0.6428	0.7864	0.6537	0.6485	0.6776	0.7908	0.6437	0.7088	0.7974
Early Stage Diabetes Risk Prediction	0.6402	0.5939	0.6523	0.6862	0.7011	0.6408	0.6746	0.7078	0.7041	0.6317	0.6892	0.7146
Lymphography	0.5359	0.5601	0.6328	0.5299	0.6832	0.5558	0.5730	0.6329	0.6854	0.5335	0.6648	0.7026
Mobile Price	0.2531	0.2647	0.3287	0.2560	0.3301	0.2750	0.2634	0.2818	0.4000	0.2556	0.3297	0.3871
Gender Gap in Spanish WP	0.6024	0.5033	0.5327	0.6128	0.6213	0.5527	0.5263	0.5678	0.5834	0.6107	0.5876	0.6283
Steel Plates Faults	0.5489	0.5427	0.5765	0.5665	0.6167	0.5809	0.5886	0.5618	0.6035	0.5516	0.6205	0.6609
Ionosphere	0.6036	0.6500	0.6489	0.6658	0.6623	0.6450	0.6831	0.6429	0.6834	0.6406	0.6752	0.7331
QSAR biodegradation	0.6324	0.6464	0.6628	0.6346	0.6712	0.6464	0.6420	0.6547	0.6809	0.6278	0.6493	0.6934
Arcene	0.5442	0.5600	0.5692	0.5438	0.6923	0.6099	0.5535	0.5868	0.6337	0.5380	0.6570	0.7018

Bold texts represent the maximum values

coefficient values and FM values. The minimum values for each row in Tables 4, 5, 6 and 7 as well as the maximum values for each row in Tables 8 and 9 are shown in bold for clarity.

Tables 3 and 4 show that the algorithm proposed in this paper performed better results on most of the datasets. It can be found that even though SLBHA can always converge to the best fitness value in iterations, it cannot get the results steadily. The iteration curves of the clustering algorithms are shown in Fig. 8. It can be seen that the proposed SLBHA can quickly and accurately find satisfactory solutions in all datasets than the other comparison algorithms. Table 6 shows that the SLBHA is better compared to the other algorithms in terms of average fitness values on all the datasets except the Shill Bidding dataset. For comparison, the experimental results from Tables 4, 5, 6 and 7 on the Shill Bidding dataset indicate that though the average fitness values obtained by the SOS algorithm are better than the two proposed algorithms, the minimum values are not good enough. In Table 7, we can find that the SOS algorithm is more stable than the proposed methods in this paper on several datasets. It may be concluded that the SOS algorithm failed to find the optimal solution and got trapped in the local optimal solution, then this situation repeated several times stably and caused such a result. The same phenomenon can also be seen in the results of other algorithms. For example, the traditional method K-means algorithm performs better than most of the heuristic algorithms in Table 7, which indicates that the K-means algorithm is more stable. The main reason for this phenomenon may be that the convergence of the K-means algorithm is certifiable and it has a simple structure and easy rules. It should be noted here that we pay more attention to the performance of the algorithm than stability. Therefore, although the K-means algorithm is stable enough, it cannot make us satisfied. It should also be pointed out that the other selected algorithms and the proposed algorithm are all heuristic algorithms, which are influenced by randomness as mentioned above, and there is no algorithm whose stability is better than other algorithms in Table 7. However, the proposed algorithm in this paper still needs to pay attention to the improvement of stability. Considering the Arcene dataset, which has 10,000 dimensions and 900 instances, its high dimension brings challenges to clustering algorithms. It can be seen from the experiments that the proposed method performs better than other algorithms, which shows that the SLBHA is also suitable for such high dimensional datasets and it is a feasible solution to this challenge.

It can be found from Tables 8 and 9 that the proposed algorithm outperforms the other methods in terms of the external metrics except for the Mobile Price dataset which means that clusters obtained by SLBHA are closer to the distribution of the original datasets compared to the other algorithms. In summary, it can be proved that the proposed algorithm can

effectively find the best clustering centroids that are closer to the real distribution on the above datasets.

In summary, it can be found that the algorithm proposed in this article performs better than most comparative algorithms in terms of best, worst, and average fitness values, as well as Jaccard coefficient and FM values, including traditional clustering algorithms and heuristic algorithms. However, the proposed SLBHA performs less prominently in terms of standard deviation. It can be concluded that SLBHA has a good ability to converge to the optimal solution and find the results closest to the original label distribution. However, this ability may pose a risk of instability. The algorithm performs well on multiple datasets, which also verifies the universality of the application of the algorithms in this paper.

Time complexity

SLBHA proposed in this paper is designed on the framework of classical BHA. The time complexity of the BHA is mainly related to the total number of iterations T and population size N , and its time complexity is lower than other heuristic algorithms [54]. The modification works of the former two strategies compared to the BHA are constant in cost, so its time complexity is comparable to that of the BHA. Then the SLBHA adds a replacement mechanism and a random control parameter. The cost of the replacement mechanism is linearly related to the population size N . Therefore, although its time complexity is larger than that of BHA, these extra costs will not reduce the availability of the algorithm.

To further compare the time complexity of the proposed algorithm and the other algorithms, experiments of running time are conducted on all the datasets. The parameter settings for experiments are the same as the previous content. The experimental results of the average running time are listed in Table 10. It can be seen from Table 10 that the running time of K-means, K-means ++ and FC-Kmeans, three traditional-based clustering algorithms, is significantly shorter than that of heuristic algorithms. Compared with the K-means algorithm, K-means ++ has fewer iterations, which makes it get the least running time among all the above algorithms. The FC-Kmeans algorithm, on the other hand, takes longer than both because it combines the K-means and the K-means ++ algorithms. Among all the heuristic algorithms, SLBHA and classical PSO have the least running time, which means that the proposed algorithm performs better while also achieving faster convergence speed. It is worth noting that the experimental results show that the running time of the BHA is higher than that of the SLBHA. After observing and analyzing the experimental process, we believe that the possible reason for this phenomenon is that the BHA algorithm is prone to falling into the local optima. In the process of BHA, it can be observed that each star in each iteration needs to be calculated whether it is too close to the black hole. When

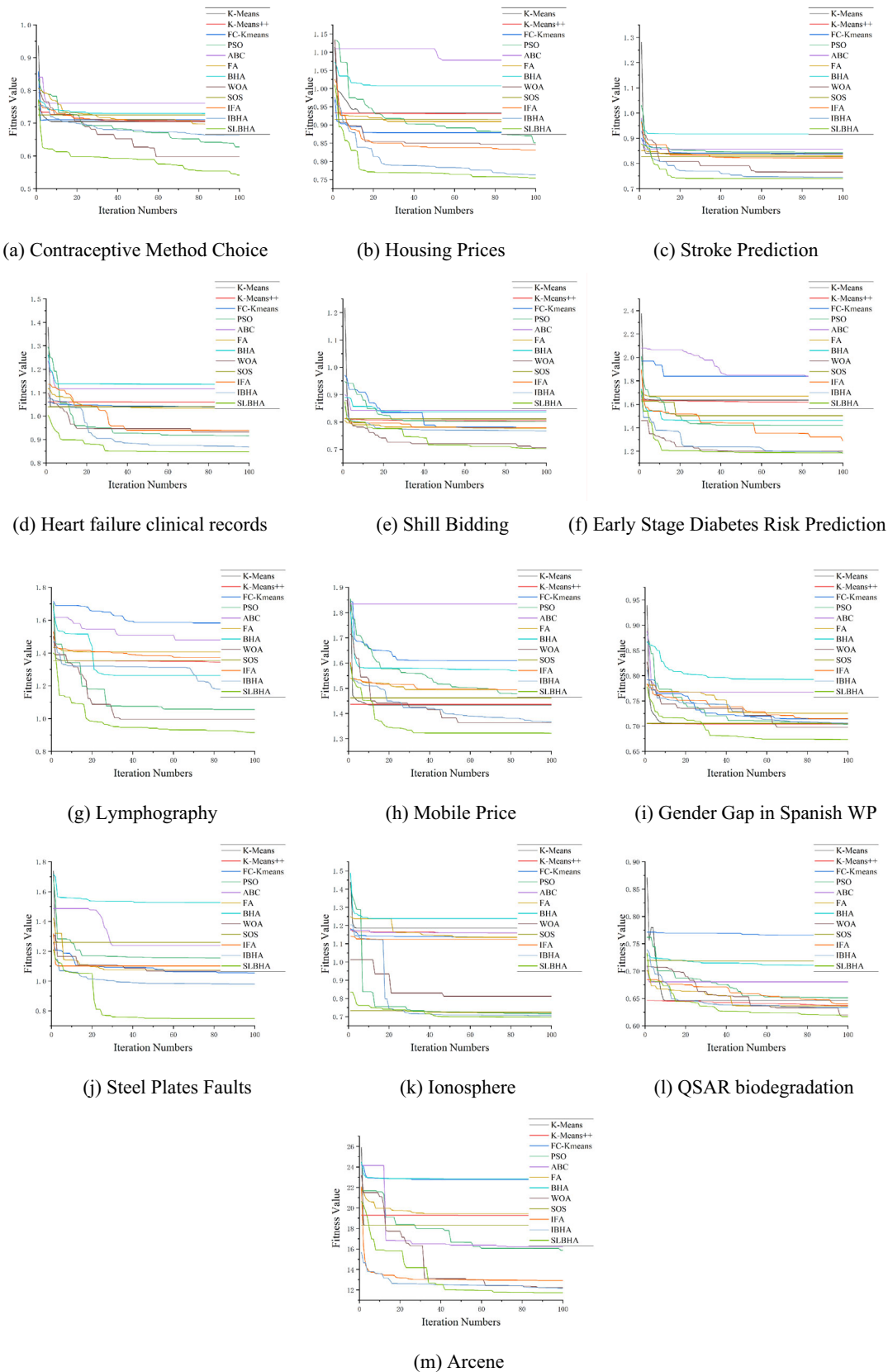


Fig. 8 Iterative curves for different datasets

Table 10 Comparison in terms of the running time

Datasets	K-means	K-means +	FC-Kmeans	PSO	ABC	FA	BHA	WOA	SOS	IFA	IBHA	SLBHA
Contraceptive Method Choice	0.0433	0.0175	0.0585	0.8047	1.3977	7.4721	1.0242	0.8980	3.3200	7.4705	1.2588	0.8908
Housing Prices	0.0288	0.0072	0.0361	0.5305	0.9324	4.9404	0.6248	0.5735	2.1926	4.9558	0.8111	0.5994
Stroke Prediction	0.0863	0.0207	0.1099	1.8866	2.8048	47.0581	6.4014	1.9027	6.9965	46.6054	2.9582	1.8056
Heart failure clinical records	0.0187	0.0016	0.0206	0.3202	0.5970	3.0667	0.4345	0.3304	1.4058	3.2852	0.5314	0.4173
Shill Bidding	0.1027	0.0248	0.3042	1.9560	3.5514	52.6641	2.8564	5.6599	8.4182	22.9819	8.9081	2.0192
Early Stage Diabetes Risk Prediction	0.0244	0.0030	0.0266	0.4327	0.7746	4.0442	0.5520	0.4428	1.8088	4.2939	0.7189	0.5186
Lymphography	0.0169	0.0010	0.0181	0.2831	0.5472	2.7147	0.3699	0.3027	1.2042	2.9643	0.4759	0.3927
Mobile Price	0.0947	0.0813	0.1612	1.9287	3.1491	17.9641	2.4590	2.2358	8.0068	18.1309	2.7688	1.8863
Gender Gap in Spanish WP	0.5178	0.0930	0.5456	6.4546	6.9374	91.0443	5.8795	9.7317	17.3088	60.0612	14.4549	3.3412
Steel Plates Faults	0.0765	0.0203	0.0904	1.6496	3.0513	15.6960	2.5747	1.9318	6.8974	15.9668	2.9886	1.7465
Ionosphere	0.0271	0.0038	0.0299	0.4798	0.8767	4.5240	0.6097	0.4992	2.0569	4.7259	0.7787	0.5209
QSAR biodegradation	0.0561	0.0137	0.0643	1.1248	2.0370	11.0830	1.7596	1.3620	4.9327	10.9370	2.1525	1.2569
Arcene	3.6631	0.2852	3.3808	72.4763	111.7453	736.8279	83.3874	71.0098	278.7954	677.5785	97.2481	65.4349

falling into a local optimal solution, the stars are very similar and near the black hole, which means that there are too many new stars generated in this process. Once BHA falls into a local optimum too early, more stars are regenerated which results in an increase in time cost. In contrast, although the strategy proposed in this article increases a portion of the runtime, the stars are more dispersed in the solution space, with fewer agents being regenerated due to being too close to the black hole. The reduced time required to generate new agents exceeds the time required for redundant operations. Therefore, SLBHA not only greatly improves BHA’s performance, but also controls time costs when addressing the clustering problem. Overall, the burden of being forced to generate a large number of new agents due to local optima is a potential flaw of BHA that has not been discovered before, and this phenomenon is worth noting.

In conclusion, the running time experiments supplement the analysis theory of time complexity mentioned above. The experimental results indicate that SLBHA also outperforms most of the compared heuristic algorithms in terms of time cost. It should be seen as an advantage of SLBHA that it ensures the quality of the obtained solutions while reducing time costs.

Statistical tests

Friedman test

The Friedman test is a non-parametric test that can be used as a tool for determining whether there is a statistically significant difference between three or more groups [57]. This test is particularly useful when the size of samples is very small. The null hypothesis is set as that there is no significant difference between the given algorithms. Then the alternate hypothesis is that at least two of them are different from each other. Here, the test statistic for Friedman test is given as follows:

$$F_R = \frac{12}{N_d K_m (K_m + 1)} \sum R_i^2 - 3N_d (K_m + 1), \tag{14}$$

where N_d is the total number of datasets, K_m is the total number of algorithms and R_i is the sum of ranks of all datasets for algorithm i . The average rank of the algorithms based on the average fitness values is shown in Table 11. After calculating the test statistic, we can draw a conclusion about whether these algorithms are significantly different through the decision rules about the Friedman test. For one thing, if the statistic F_R is larger than the critical value that can be found in Friedman’s critical values table, the null hypothesis can be rejected. For another thing, if the p value is less than or equal to the α which is the level of significance, we can also reject the null hypothesis. Otherwise, the null hypothesis

Table 11 The average rank of the algorithms

Algorithms	K-means	K-means + +	FC-Kmeans	PSO	ABC	FA	BHA	WOA	SOS	IFA	IBHA	SLBHA
Rank	7.3846	12.0	10.6153	5.6153	4.1538	9.3846	6.0	9.3076	2.0	6.4615	4.0	1.0769

p value = $1.7827e - 21$ statistic = 125.0

should be accepted. Table 11 also shows the calculated results of the Friedman test. It can be seen in Table 11 that SLBHA got the best average rank and the K-means algorithm got the worst. The statistic F_R is 125.0 and the p value is $1.7827e - 21$. The former is greater than the critical value while the latter is less than alpha, which means that we can reject the null hypothesis. In conclusion, there is a significant difference between the algorithms tested in this paper.

Wilcoxon rank sum test

The Wilcoxon rank sum test is also a non-parametric test. It can be used to determine whether two dependent groups are selected from the same distribution [58]. The p values of the Wilcoxon rank sum test of the proposed SLBHA are reported in Table 11. The values below 0.05 are underlined in the tables. As shown in Table 12, there is a significant difference between SLBHA with the selected algorithms because all of the Wilcoxon rank sum test results are less than 0.05. Combined with the other tables mentioned above, it can be concluded that SLBHA is significantly superior to other algorithms.

Conclusions and future work

In this article, an improved self-adaptive logarithmic spiral path black hole algorithm (SLBHA) is discussed and analyzed. The path improvement measures introduced in SLBHA effectively enhance the local search capability of the algorithm. Then the replacement mechanism of the stars increases the diversity of the population. Additionally, SLBHA uses a self-adaptive parameter to balance the global and local search phases. Therefore, the algorithm effectively improves the exploration and exploitation capabilities of BHA and the ability to maintain the balance between them, with high availability and effectiveness. Moreover, the effectiveness of the proposed algorithm is experimentally verified. The quantization error and external criteria (Jaccard coefficient and FM values) are utilized to measure the performance of clustering algorithms. The experimental results show that the SLBHA outperforms the other comparative algorithms on most of the datasets and the generated clusters are closer to the label distribution in reality. The time complexity of SLBHA is also better than other heuristic algorithms. Statistical tests indicate that there is a significant difference between the proposed algorithm and other compared algorithms. However, the experiments also show that the shortcoming of the algorithm is that its stability is strongly affected by randomness. Further future work is mainly in two aspects. On the one hand, there is still room for improvement in the stability of the proposed algorithms and it may relate to the control of randomness, which is a common problem

Table 12 The Wilcoxon rank sum test results of SLBHA

Datasets	K-means	K-means ++	FC-Kmeans	PSO	ABC	FA	BHA	WOA	SOS	IFA	IBHA
Contraceptive Method Choice	2.60E-06	1.73E-06	1.22E-04	3.18E-06	6.04E-03	1.73E-06	2.88E-06	1.73E-06	9.63E-04	2.60E-06	7.69E-06
Housing Prices	1.73E-06	1.73E-06	3.05E-05	1.73E-06	3.18E-06	1.73E-06	1.73E-06	1.73E-06	1.92E-06	1.73E-06	4.29E-06
Stroke Prediction	1.92E-06	1.73E-06	1.73E-06	1.73E-06	3.52E-06	1.73E-06	1.73E-06	1.73E-06	3.33E-02	1.73E-06	1.49E-05
Heart failure clinical records	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.11E-03	1.73E-06	1.73E-06	1.73E-06	6.04E-03	1.73E-06	2.05E-04
Shill Bidding	1.73E-06	1.73E-06	1.73E-06	1.73E-06	1.64E-05	1.73E-06	1.73E-06	3.18E-06	2.60E-06	1.73E-06	3.52E-06
Early Stage Diabetes Risk Prediction	1.73E-06	1.73E-06	1.73E-06	1.73E-06	4.73E-06	1.73E-06	1.92E-06	3.52E-06	3.72E-05	1.73E-06	2.35E-06
Lymphography	1.73E-06	1.73E-06	1.73E-06	1.73E-06	2.60E-06	1.73E-06	1.73E-06	1.73E-06	2.80E-02	1.73E-06	6.98E-06
Mobile Price	1.73E-06	1.73E-06	1.73E-06	1.73E-06	9.71E-05	1.73E-06	1.73E-06	1.73E-06	4.11E-03	1.92E-06	2.13E-06
Gender Gap in Spanish WP	4.29E-06	1.73E-06	1.73E-06	2.88E-06	1.48E-04	1.73E-06	1.73E-06	1.73E-06	1.25E-04	3.52E-06	7.69E-06
Steel Plates Faults	1.73E-06	1.73E-06	1.73E-06	5.22E-06	4.07E-05	1.73E-06	1.49E-05	1.73E-06	2.83E-04	1.92E-06	6.32E-05
Ionosphere	1.73E-06	1.73E-06	1.73E-06	2.35E-06	1.13E-05	1.73E-06	1.92E-06	1.73E-06	7.51E-05	2.13E-06	3.11E-05
QSAR biodegradation	1.73E-06	1.73E-06	1.73E-06	1.49E-05	4.73E-06	1.73E-06	9.32E-06	1.73E-06	3.85E-03	4.73E-06	1.73E-06
Arcene	1.73E-06	1.73E-06	1.73E-06	1.73E-06	2.88E-06	1.73E-06	1.73E-06	1.73E-06	4.73E-06	1.73E-06	5.22E-06

for heuristic algorithms. On the other hand, the proposed algorithm can be applied to solve other optimization problems and they may need to make the corresponding changes according to the application scenarios.

Funding This work was supported by the Key Project of Ningxia Natural Science Foundation [2022AAC02043], the National Natural Science Foundation of China under Grant [11961001], the Natural Science Foundation of NingXia Hui Autonomous Region [2021AAC03185], the Research Startup Foundation of North Minzu University [2020KYQD23], First-class Discipline Construction Fund project of Ningxia Higher Education [NXYLXK2017B09], and Major scientific Research Project of Northern University for Nationalities [ZDZX201901], Basic discipline research projects supported by Nanjing Securities [NJZQJCXK202201].

Data availability The datasets analysed during the current study are available in the website of Kaggle (<https://www.kaggle.com/>) and the UCI Repository (<http://archive.ics.uci.edu/ml/index.php>).

Declarations

Conflict of interest The authors declare that there are no conflict of interests, we do not have any possible conflicts of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Xu D, Tian Y (2015) A comprehensive survey of clustering algorithms. *Ann Data Sci* 2:165–193. <https://doi.org/10.1007/s40745-015-0040-1>
- Hossain MZ, Akhtar MN, Ahmad RB et al (2019) A dynamic K-means clustering for data mining. *Indonesian J Elect Eng Comput Sci* 13:521–526. <https://doi.org/10.11591/ijeecs.v13.i2.pp521-526>
- Dhanachandra N, Manglem K, Chanu YJ (2015) Image segmentation using K-means clustering algorithm and subtractive clustering algorithm. *Procedia Comput Sci* 54:764–771. <https://doi.org/10.1016/j.procs.2015.06.090>
- Wiwie C, Baumbach J, Röttger R (2015) Comparing the performance of biomedical clustering methods. *Nat Methods* 12:1033–1038. <https://doi.org/10.1038/nmeth.3583>
- Cooper C, Franklin D, Ros M et al (2017) A comparative survey of VANET clustering techniques. *IEEE Commun Surv Tutor* 19:657–681. <https://doi.org/10.1109/COMST.2016.2611524>
- Fraley C, Raftery AE (1998) How many clusters? Which clustering method? Answers via model-based cluster analysis. *Comput J* 41:578–588. <https://doi.org/10.1093/comjnl/41.8.578>
- Madhulatha TS (2012) An overview on clustering methods. *IOSR J Eng* 2:4
- Rokach L, Maimo O (2005) Clustering methods. *The data mining and knowledge discovery handbook*
- Xie H, Zhang L, Lim CP et al (2019) Improving K-means clustering with enhanced Firefly Algorithms. *Appl Soft Comput* 84:105763. <https://doi.org/10.1016/j.asoc.2019.105763>
- Qiao S, Zhou Y, Zhou Y et al (2019) A simple water cycle algorithm with percolation operator for clustering analysis. *Soft Comput* 23:4081–4095. <https://doi.org/10.1007/s00500-018-3057-5>
- Boushaki SI, Kamel N, Bendjeghaba O (2018) A new quantum chaotic cuckoo search algorithm for data clustering. *Expert Syst Appl* 96:358–372. <https://doi.org/10.1016/j.eswa.2017.12.001>
- Bouyer A, Hatamlou A (2018) An efficient hybrid clustering method based on improved cuckoo optimization and modified particle swarm optimization algorithms. *Appl Soft Comput* 67:172–182. <https://doi.org/10.1016/j.asoc.2018.03.011>
- Niknam T, Fard ET, Pourjafarian N et al (2011) An efficient hybrid algorithm based on modified imperialist competitive algorithm and K-means for data clustering. *Eng Appl Artif Intel* 24:306–317. <https://doi.org/10.1016/j.engappai.2010.10.001>
- Xiao J, Yan YP, Zhang J et al (2010) A quantum-inspired genetic algorithm for k-means clustering. *Expert Syst Appl* 37:4966–4973. <https://doi.org/10.1016/j.eswa.2009.12.017>
- Hatamlou A, Hatamlou M (2013) PSOHS: an efficient two-stage approach for data clustering. *Memetic Comp* 5:155–161. <https://doi.org/10.1007/s12293-013-0110-x>
- Hatamlou A (2013) Black hole: a new heuristic optimization approach for data clustering. *Inf Sci* 222:175–184. <https://doi.org/10.1016/j.ins.2012.08.023>
- Soto R, Crawford B, Olivares R et al (2018) Adaptive black hole algorithm for solving the set covering problem. *Math Probl Eng*. <https://doi.org/10.1155/2018/2183214>
- Hatamlou A (2018) Solving travelling salesman problem using black hole algorithm. *Soft Comput* 22:8167–8175. <https://doi.org/10.1007/s00500-017-2760-y>
- Pashaei E, Aydin N (2017) Binary black hole algorithm for feature selection and classification on biological data. *Appl Soft Comput* 56:94–106. <https://doi.org/10.1016/j.asoc.2017.03.002>
- Eskandarzadehalamdary M, Masoumi B, Sojodishijani O (2014) A new hybrid algorithm based on black hole optimization and bisecting k-means for cluster analysis. In: 2014 22nd Iranian Conference on Electrical Engineering (ICEE), 1075–1079. <https://doi.org/10.1109/IranianCEE.2014.6999695>
- Pashaei E, Pashaei E, Aydin N (2019) Gene selection using hybrid binary black hole algorithm and modified binary particle swarm optimization. *Genomics* 111:669–686. <https://doi.org/10.1016/j.ygeno.2018.04.004>
- Deeb H, Sarangi A, Mishra D et al (2020) Improved Black Hole optimization algorithm for data clustering. *J King Saud Univ Comput Inf Sci*. <https://doi.org/10.1016/j.jksuci.2020.12.013>
- Pal SS, Pal S (2020) Black hole and k-means hybrid clustering algorithm. *Computational intelligence in data mining*. *Adv Intell Syst Comput*. https://doi.org/10.1007/978-981-13-8676-3_35
- Khairi M S, Zuwairie I, Hamdan D et al (2016) A new hybrid gravitational search-black hole algorithm
- Yaghoobi S, Mojallali H (2016) Modified black hole algorithm with genetic operators. *Int J Comput Intell Syst* 9:652–665. <https://doi.org/10.1016/10.1080/18756891.2016.1204114>
- Mohammed S, Ibrahim Z, Daniyal H et al (2017) Improving the effectiveness of the black hole algorithm using a local search technique. *Int J Simul Syst Sci Technol*. <https://doi.org/10.5013/IJSSST.a.18.04.12>
- Ibrahim Z, Mohammed S, Subari N et al (2018) Black hole white hole algorithm with local search. 2018 International Conference on Artificial Life and Robotics (ICAROB2018)

28. Abualigah L, Elaziz MA, Sumari P et al (2022) Black hole algorithm: a comprehensive survey. *Appl Intell*. <https://doi.org/10.1007/s10489-021-02980-5>
29. Maulik U, Bandyopadhyay S (2000) Genetic algorithm-based clustering technique. *Pattern Recognit* 33:1455–1465. [https://doi.org/10.1016/S0031-3203\(99\)00137-5](https://doi.org/10.1016/S0031-3203(99)00137-5)
30. Fatahi M, Moradi S (2020) An FPA and GA-based hybrid evolutionary algorithm for analyzing clusters. *Knowl Inf Syst* 62:1701–1722. <https://doi.org/10.1007/s10115-019-01413-7>
31. D W van der M, A P E (2003) Data clustering using particle swarm optimization. *The 2003 Congress on Evolutionary Computation*, 1: 215–220. <https://doi.org/10.1109/CEC.2003.1299577>
32. Li Y, Chu X, Tian D et al (2021) Customer segmentation using K-means clustering and the adaptive particle swarm optimization algorithm. *Appl Soft Comput* 113:107924. <https://doi.org/10.1016/j.asoc.2021.107924>
33. Shelokar PS, Jayaraman VK, Kulkarni BD (2004) An ant colony approach for clustering. *Anal Chim Acta* 509:187–195. <https://doi.org/10.1016/j.aca.2003.12.032>
34. Niknam T, Olamaei J, Amiri B (2008) A hybrid evolutionary algorithm based on ACO and SA for cluster analysis. *J Appl Sci*. <https://doi.org/10.3923/jas.2008.2695.2702>
35. Niknam T, Amiri B (2010) An efficient hybrid approach based on PSO, ACO and k-means for cluster analysis. *Appl Soft Comput* 10:183–197. <https://doi.org/10.1016/j.asoc.2009.07.001>
36. Hatamlou A, Abdullah S, Nezamabadi-pour H (2012) A combined approach for clustering based on K-means and gravitational search algorithms. *Swarm Evol Comput* 6:47–52. <https://doi.org/10.1016/j.swevo.2012.02.003>
37. Dowlatshahi MB, Nezamabadi-pour H (2014) GGSA: a grouping gravitational search algorithm for data clustering. *Eng Appl Artif Intell* 36:114–121. <https://doi.org/10.1016/j.engappai.2014.07.016>
38. Han XH, Quan L, Xiong XY et al (2017) A novel data clustering algorithm based on modified gravitational search algorithm. *Eng Appl Artif Intell* 61:1–7. <https://doi.org/10.1016/j.engappai.2016.11.003>
39. Senthilnath J, Omkar SN, Mani V (2011) Clustering using firefly algorithm: performance study. *Swarm Evol Comput* 1:164–171. <https://doi.org/10.1016/j.swevo.2011.06.003>
40. Pranesh D, Dushmanta KD, Shouvik D (2018) A modified Bee Colony Optimization (MBCO) and its hybridization with k-means for an application to data clustering. *Appl Soft Comput* 70:590–603. <https://doi.org/10.1016/j.asoc.2018.05.045>
41. Zhou Y, Wu H, Luo Q et al (2019) Automatic data clustering using nature-inspired symbiotic organism search algorithm. *Knowl Inf Syst* 163:546–557. <https://doi.org/10.1016/j.knosys.2018.09.013>
42. Tawhid MA, Ibrahim AM (2023) An efficient hybrid swarm intelligence optimization algorithm for solving nonlinear systems and clustering problems. *Soft Comput* 27:8867–8895. <https://doi.org/10.1007/s00500-022-07780-8>
43. Chakraborty S, Saha AK, Chakraborty R et al (2021) An enhanced whale optimization algorithm for large scale optimization problems. *Knowl Based Syst* 233:107543. <https://doi.org/10.1016/j.knosys.2021.107543>
44. Almotairi KH, Abualigah L (2022) Hybrid reptile search algorithm and remora optimization algorithm for optimization tasks and data clustering. *Symmetry* 14:458. <https://doi.org/10.3390/sym14030458>
45. Mirjalili S, Lewis A (2016) The whale optimization algorithm. *Adv Eng Softw* 95:51–67. <https://doi.org/10.1016/j.advengsoft.2016.01.008>
46. Wu J, Wang YG, Burrage K et al (2020) An improved firefly algorithm for global continuous optimization problems. *Expert Syst Appl* 149:113340. <https://doi.org/10.1016/j.eswa.2020.113340>
47. Sharma S, Kumar S, Nayyar A (2019) Logarithmic Spiral Based Local Search in Artificial Bee Colony Algorithm. *Industrial Networks and Intelligent Systems. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, 257. https://doi.org/10.1007/978-3-030-05873-9_2
48. Kapoor A, Singhal A (2017) A comparative study of K-Means, K-Means++ and Fuzzy C-Means clustering algorithms, 2017 3rd International Conference on Computational Intelligence & Communication Technology (CICT), 1–6. <https://doi.org/10.1109/CICT.2017.7977272>
49. Ay M, Özbakır L, Kulluk S et al (2023) FC-Kmeans: fixed-centered K-means algorithm. *Expert Syst Appl* 211:118656. <https://doi.org/10.1016/j.eswa.2022.118656>
50. Karaboga D, Ozturk C (2011) A novel clustering approach: Artificial Bee Colony (ABC) algorithm. *Appl Soft Comput* 11:652–657. <https://doi.org/10.1016/j.asoc.2009.12.025>
51. Karaboga D, Akay B (2009) A comparative study of Artificial Bee Colony algorithm. *Appl Math Comput* 214:108–132. <https://doi.org/10.1016/j.amc.2009.03.090>
52. Nasiri N, Khiyabani FM, Yoshise A (2018) A whale optimization algorithm (WOA) approach for clustering. *Cogent Math Stat* 5:1. <https://doi.org/10.1080/25742558.2018.1483565>
53. Chen X, Cheng L, Liu C et al (2020) A WOA-based optimization approach for task scheduling in cloud computing systems. *IEEE Syst J* 14:3117–3128. <https://doi.org/10.1109/JSYST.2019.2960088>
54. Dua D, Graff C (2019) UCI Machine Learning Repository [<http://archive.ics.uci.edu/ml>]. Irvine, CA: University of California, School of Information and Computer Science.
55. Halkidi M, Batistakis Y, Vazirgiannis M (2001) On clustering validation techniques. *Int J Intell Syst* 17:107–145. <https://doi.org/10.1023/A:1012801612483>
56. Saxena A, Prasad M, Gupta A et al (2017) A review of clustering techniques and developments. *Neurocomputing* 267:664–681. <https://doi.org/10.1016/j.neucom.2017.06.053>
57. Ibrahim A, Maria H, Hossam F et al (2020) A dynamic locality multi-objective salp swarm algorithm for feature selection. *Comput Ind Eng* 147:106628. <https://doi.org/10.1016/j.cie.2020.106628>
58. Sayed GI, Khoriba G, Haggag MH (2018) A novel chaotic salp swarm algorithm for global optimization and feature selection. *Appl Intell* 48:3462–3481. <https://doi.org/10.1007/s10489-018-1158-6>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.