Journal of Ambient Intelligence and Humanized Computing
https://doi.org/10.1007/s12652-019-01624-4

ORIGINAL RESEARCH

Multi-feature fusion and selection method for an improved particle swarm optimization

Lu Xiong[1] · Ruey-Shun Chen[2] · Xuefeng Zhou[3] · Changqiang Jing[4]

Received: 8 August 2019 / Accepted: 4 December 2019

Abstract
In data mining, feature selection is an important part in data preprocessing, therefore selecting the optimal feature subset can effectively reduce the data dimension and computing cost of the learning algorithm. In this paper, Binary Particle Swarm Optimization is used to optimize the feature selection process. Firstly, a population initialization strategy based on chaos theory is proposed to reduce information redundancy and improve the quality of the initial population. Secondly, a chaos clonal operator, chaos mutation operator, and immune selection operator are proposed to improve the convergence speed and population diversity of the algorithm. Finally, three representative optimization algorithms were selected and compared in eight UCI datasets. The experimental results showed that the improved IBPSOCC algorithm is superior to other contrast algorithms in reducing the number of features and improving the classification accuracy.

Keywords Binary particle swarm optimization · Feature selection · Immune operator · Chaos optimization

1 Introduction

The feature selection problem, also known as the problem of feature subset selection, refers to selecting the N features from the existing M features to optimize the specific objectives of the system so as to decrease the data dimension and optimize the performance of learning algorithm (Ghosh 2018). With the progress of data mining, machine learning and computer science in recent years, an increasing amount of high-dimensional datasets are being applied to information systems in various fields, such as business management, financial analysis, and medical studies (Kou et al. 2014). Dimensional disasters caused by high-dimensional datasets dramatically increase the complexity of data processing. Therefore, eliminating redundant or fewer information features and selecting the optimal feature subset would greatly improve the efficiency of subsequent learning algorithms (Moradi and Gholampour 2016; Hamedmoghadam et al. 2018; Wei et al. 2017; Luo et al. 2016; Yang et al. 2018; Mo et al. 2018; Niu et al. 2018).

Supposing that there are n-dimensional features in the dataset, then the total search space is 2, and thus the problem of feature selection became an NP-hard problem (Gheyas and Smith 2010; Liu and Motoda 2007). Therefore, a large amount of computational and time overhead is wasted on exhaustive search for the optimal feature subset time, which is not suitable for settling the feature subset selection problem for large high-dimensional datasets. At the same time because of the advantages of the heuristic search algorithm, a set of solutions can be obtained after one operation, which can save less time and computational cost searching for the ideal solution set. In recent years, to solve the feature selection problem, an increasing number of heuristic algorithms based on feature selection algorithms have been proposed

✉ Changqiang Jing
jingchangqiang@lyu.edu.cn

Lu Xiong
luxiong0822@qq.com

Ruey-Shun Chen
chen1868@gmail.com

Xuefeng Zhou
xf.zhou@giim.ac.cn

[1] College of Automation, Zhongkai University of Agriculture and Engineering, Guangzhou, China

[2] Department of Computer Engineering, Dongguan Polytechnic, Dongguan 523808, China

[3] Guangdong Key Laboratory of Modern Control Technology, Guangdong Institute of Intelligent Manufacturing, Guangzhou, China

[4] Department of Inform, LinYi University, Linyi 276000, China

and have achieved good results (Dash and Liu 1997), for example, GA (genetic algorithm) (Yang and Honavar 1998), ACO (ant colony algorithm) (Sreeja and Sankar 2015), ABC (artificial bee colony algorithm) and PSO (particle swarm optimization algorithm) (Vieira et al. 2013). The PSO algorithm (Kennedy 1998) has attracted many scholars' attention due to its fast convergence speed and strong search ability, and researchers have put forward a lot of improved algorithms. In the literature (Wang et al. 2007; Wang and Guo 2017; Liu et al. 2015b), a PSO algorithm based on rough set was proposed to finding the optimal feature subset. The conclusion was drawn that the feature selection algorithm based on the PSO could solve the feature selection problem more effectively than a GA-based method. Other researchers (Zhang et al. 2014) have proposed a new mutation strategy to improve the PSO algorithm. The algorithm can avoid falling into local optimization by randomly flipping the position of particles. Udhaya Kumar and Hannah Inbarani (2016) proposed a method based on rough set feature selection technology and PSO to solve the problem of feature dimensionality reduction in brain-computer interface motion images research. The experimental results showed that the feature subset selected by this algorithm is applied to the proposed domain rough set classifier method for the classification of multi-class motion image, and good results were obtained. Maryam and Behrouz (2018) proposed a feature ranking method based on MPSO (multi-objective particle swarm optimization), which archives feature frequencies and guides particle evolution, its performance was checked on nine benchmark datasets, and it achieved ideal results. Saber and Yahya (2018) proposed a Bayesian information criterion based on logistic regression model and the algorithm of particle swarm optimization, as the fitness function. Experiments on many datasets showed that their method significantly improves classification performance and decreases the number of characteristic.

At the same time, some scholars have also combined PSO with different classifiers to settle the feature selection problem. Huang and Dun (2008) proposed a algorithm of feature selection combining PSO with SVM (support vector machine), which reduces the time cost by using the distributed architecture of Web services technology. Their method has been divided into two stages (Chhikara et al. 2016; Chien-Ming et al. 2019; Tsu-Yang et al. 2019). First, two typical filtering techniques are adopted: test and multiple regression, and the image selections are based on feature recognition ability. Second, the size of the population is dynamically changed in the PSO algorithm, and SVM is used as the classification for the feature test. Experimental results showed that the method could improve the accuracy of classification and reduce the feature dimension. For solving the problem of cancer classification, first, irrelevant and redundant features were filtered by a fast correlation based feature selection (FCBC) method, and then the SVM is optimized by PSO. The experimental results on nine cancer datasets showed that the PA-SVM proposed in this research has good flexibility and adaptability in dealing with complex non-linear problems (Lingyun et al. 2017). In order to settling the feature selection problem in medical diagnosis, Ye (2016) proposed the method of combining GA and PSO. In their method, first, selecting the feature subset by GA, and then Optimizing the parameters of SVM by PSO. Because PSO showed advantages in searching suitable continuous variables, the experiment is carried out according to UCI data. The results showed that the hybrid algorithm proposed in this research is significantly better than the single algorithm in choosing the best parameters of support vector machine model provided a more relevant feature subset for SVM, which further improves the classification of medical diagnosis problems. Tran et al. (2014) proposed a new initialization and update strategy in the binary algorithm of PSO, that uses the KNN (K Nearest Neighbour) as the classifier. Liao and Kuo (2018) improved the discrete Symbiotic Organisms Search algorithm based on BPSO and SOS, which using KNN as a classifier. The algorithm performance was evaluated depend on its classification accuracy and computing time. The results of comparative experiments on multiple datasets showed that the performance of the improved algorithm is significantly improved. One common disadvantage of intelligent optimization algorithms, such as PSO for feature selection, is that the workload for computing the fitness is large. To overcome this shortcoming and increase the practicability of PSO. Chaos and immune technology, using developments in computer technology, are introduced to shorten the calculation time and improve efficiency. Based on binary PSO, a feature selection method for binary immune particle swarm optimization based on chaotic cloning is proposed. In this method, the chaos clonal operator and immune operator are used to optimize the binary PSO. It has a strong search ability to find a better feature subset and reduce the algorithm's running time, and thus, a representative feature subset can be obtained quickly. Experimental results show that the algorithm made a significantly progress (Zhu et al. 2018; Pan et al. 2019; Chen et al. 2018, 2019).

2 The algorithm of BPSO

Kennedy proposed the BPSO algorithm in 1997 based on a continuous PSO algorithm to solve discrete optimization problems Bharti and Singh (2016). The BPSO algorithm simulates the foraging process of birds in flight. Each particle in the population is equivalent to a solution in solution sets. The particle has velocity, two attributes and position, the position vector represents the corresponding solution of the particle, and Adjusting the next flight of the particle by the velocity, so as to update the location and search for a new set of solutions. The flight direction and speed of particles are adjusted according to the flight experience of other particles and their historical experience of flight in the population. The optimal position of each particle in the historical flight process is named optimal solutionp_{best} of the individual. The best position of the whole population in the historical flight process is g_{best}, which is named the optimal solution of the global. Particles share information through $p_{\mathrm{best}}$and g_{best}, thus affecting the search behavior of the population in the evolutionary process.

2.1 Population initialization

Suppose a particle swarm consisting of N particles moves in D-dimensional search space. The ith particle position at the kth iteration is represented as $x_i^k = \left(x_{i1}^k, x_{i2}^k, \ldots, x_{iD}^k\right)$. The optimum position of a particle in flight is expressed as $p_i^k = \left(p_{i1}^k, p_{i2}^k, \ldots, p_{iD}^k\right)$, namely p_{best}. At the kth iteration, the Optimum location of all particles in particle swarm is expressed as $g_i^k = \left(g_{i1}^k, g_{i2}^k, \ldots, g_{iD}^k\right)$, namely g_{best}. The velocity of a particle is expressed as $v_i^k = \left(v_{i1}^k, v_{i2}^k, \ldots, v_{iD}^k\right)$.

2.2 Position updating and particle velocity

The position and velocity of particles in the iteration of k + 1 times are expressed as below:

$$v_{id}^{k+1} = \omega v_{id}^k + c_1 r_1 \left(p_{id}^k - x_{id}^k\right) + c_2 r_2 \left(g_{id}^k - x_{id}^k\right) \tag{1}$$

In Formula (1), c_1 and c_2 denote learning factor (acceleration constant), ωdenotes the inertia weight, $r_1$and r_2 denote random numbers in the range of (0.1). According to Formula (1), the velocity of particles includes three stages. The first stage is the velocity of the previous iteration of the particle. The next stage is "self-learning", and the factors that affect the next generation iteration flight of particles are self-flight. The third stage is "information sharing", which is the influence of the flight experience of all the particles in the particle swarm on the next iteration flight of particles.

Solving the optimization problem by the algorithm of BPSO in a discrete space. Based on the binary coding principle, each value of the particle position vector in the algorithm can only be 0 or 1. Kennedy proposed using a Sigmoid function as a conversion function to map each dimension of the position vector in the continuous space to 0 or 1 in binary coding. For the j dimensional velocity vector $v_{ij}^k$of particle i in the kth iteration, Calculating the corresponding value of Sigmoid function by Formula (2) as follows:

$$\mathrm{S}\left(v_{ij}^{k+1}\right) = \frac{1}{1 + e^{-v_{ij}^{k+1}}} \tag{2}$$

After the corresponding function value is obtained, the position vector x_{ij}^{k+1} of the corresponding dimension of the particle in the k + 1 iteration is determined according to Formula (3):

$$x_{ij}^{k+1} = \begin{cases} 1 & \rho_{id}^{k+1} < \mathrm{S}\left(v_{id}^{k+1}\right) \\ 0 & \rho_{id}^{k+1} \geq \mathrm{S}\left(v_{id}^{k+1}\right) \end{cases} \tag{3}$$

where, ρ_{id}^{k+1} is a positive real number randomly generated between (0,1).

2.3 Population evaluation

Evaluating the corresponding solution of the particle with fitness value for each particle in the population. For any particle $i = 1, 2, \ldots, N$, calculating the corresponding fitness value as Fitness (i).

After obtaining the fitness value of each particle in the population, the particle can be judged as to whether it is good or bad in the flight process of the corresponding individual and the flight process of the population. If the particle fitness value i is better than the fitness value corresponding to the historical optimum $p_{besti}$of particle i, then particle I searches for the optimal position in the history of current flight process, and thus the particle i position vector is used to update the p_{besti} vector. Similarly, if the fitness value of particle i is better than the fitness value corresponding to the global optimal g_{best} of the population, then particle i searches for the optimal position of the whole population in the current flight history, and thus the g_{best} vector is updated with the position vector of particle i; otherwise, the g_{best} vector remains unchanged.

2.4 Termination condition of the algorithm

If the maximum number of iterations meets the requirements, then terminate the algorithm and ouput the feature selection scheme corresponding to the optimal solution g_{best}, otherwise, 2.2 is returned for the next iteration.

3 Algorithmic design

To accelerate the convergence speed and keep the diversity of particle swarm, a chaotic optimization algorithm and clone selection mechanism in an artificial immune system are introduced. The objective function and constraints are regarded as antigens, and the particle swarm is regarded as the antibodies. According to the value of its affinity, the affinity antibodies are proportional to the cloning, and the affinity and the chaotic mutation is in reverse ratio. To the diversity of the population, low affinity antibodies are reinitialized in proportion (Liu et al. 2015a).

3.1 Chaotic initialization population

The process of initializing population is uncertain When using PSO. Many particle swarms may also be distributed far away from the optimal solution region, and the particles of these particle swarms are unevenly distributed. However, at this time, chaos initialization can ensure the uniform distribution of the particle swarm. In the initialization stage, to find out the particles that can be used as the initial population chaotic particle swarm optimization (CPSO) is used to make some particles using the chaotic sequence to ensure the high quality of the initial population.

3.2 Chaotic clone operator

According to the affinity order of antibody antigens, the elite clonal population A_k is constituted by M antibodies with the highest affinity constitute. The remaining (N–M) antibodies constitute population B_k, and population size M is a variable. Assuming that the antibody affinity function is f(*), then A_k is expressed as follows:

$$A_k = \left\{x_i | f\left(x_i\right) \geq \bar{f}, i = 1, 2, \ldots, N\right\}, \bar{f} = \frac{1}{N}\sum_{j=1}^{N} f_j \tag{4}$$

In formula $A_k = \left\{x_1, x_2, \ldots, x_M\right\}$, chaotic clone operator is defined as follows:

M antibodies $x_1, x_2, \ldots, x_M$ of populationA_k, and the affinity values are$f_1, f_2, \ldots, f_M$. According to population size N and affinity, x_i is cloned into q_i identical points $x_{i1}, x_{i2}, \ldots, x_{iq_i}$.

The probability that x_i produces new individuals is p_i:

$$p_i = f(i) / \sum_{j=1}^{M} f(j), i = 1, 2, \ldots, M \tag{5}$$

The number of clones of antibody x_i is q_i:

$$q_i = \frac{p_i \cdot N}{i} \tag{6}$$

The antibodies to be cloned are sorted in descending order of affinity, i is the sequence number. The higher the affinity of the antibody, the more clones it has. q_i can be adaptively adjusted according to p_i and i, and * is the upper integral function.

After cloning, the population changes to $C_k = \left\{A'_1, A'_2, \ldots, A'_M\right\}$, where $A'_j = \left\{x_{j1}, x_{j2}, \ldots, x_{jq_j}\right\}$, j = 1, 2, …, M.

3.3 Chaotic mutation operator

Chaotic variables are sensitive to initial conditions and have ergodicity, regularity and randomness. Compared with the algorithm of general random search, chaotic search has stronger ability in small spaces for local searching. Therefore, a chaotic mutation is used instead of traditional Gaussian mutation operator.

Ulam-von-Neumann chaotic mapping is utilized to realize the chaotic mutation.

$$p_{n+1} = 1 - 2p_n^2 \tag{7}$$

Assuming that $x_{ij}(d)$ is the d-th component of antibody x_{ij} in population C_k, then i = 1, 2, …, M, j = 1, 2, …, q_i, $d = 1, 2, \ldots, D$. The range of $x_{ij}(d)$ is $\left[L_d, U_d\right]$. According to Formula (8), it is mapped to the chaotic ergodic interval [− 1,1]. Then, according to (9) and (10), the mutated antibody x'_{ij} is calculated as follows:

$$p_{ij}(d) = \frac{2}{U_d - L_d} \cdot x_{ij}(d) - \frac{U_d + L_d}{U_d - L_d} \tag{8}$$

$$p'_{ij}(d) = f_{cm}\left(P_{ij}(d)\right) \tag{9}$$

$$x'_{ij}(d) = p'_{ij}(d) \cdot \frac{U_d - L'_d}{2} + \frac{U_d + L'_d}{2} \tag{10}$$

where, f_{cm} represents the chaos mapping formula of (7).

3.4 Immune selection operator

The population D_k after chaotic mutation, for any i = 1, 2, …, M has the following antibodies.

$$d_i = \left\{x'_{ij} | \max f\left(x'\right), j = 1, 2, \ldots, q_i\right\} \tag{11}$$

The population D'_k is composed of d_i, and the offspring population D'_k and the initial population A_k are combined to select the competition scale Q. A new generation of elite population A_{k+1} is formed through the way of tournament. The chaotic clone operator can effectively improve the local optimization ability by expanding and compressing the space.

For population B_k, the natural extinction process of 5% of the B cells in biological cloning selection was simulated. In B_k, d antibodies with the lowest affinity were selected and discarded by using the extinction operator $\Gamma(*)$. The reinitialization of B_k can maintain the diversity of the population as follows:

$$\Gamma(*) = \text{rand}() \cdot (\text{up} - \text{low}) + \text{low} \tag{12}$$

3.5 IBPSOCC algorithm

The basic idea of the immune binary PSO based on chaotic clonal selection (IBPSOCC algorithm) is as follows: After the antibody is initialized, the updating formula of position and velocity in the basic PSO is used to guide its "flight" direction. To accelerate the convergence rate, high-affinity antibodies are selected for cloning operation, so that more antibodies can be gathered near the "good" position. Next, the cloned antibody is chaotically mutated to search in all directions near the "good" position. Finally, the worst affinity antibodies are discarded and reinitialized to guarantee the population diversity. In this paper, a chaotic clone operator is designed for clone amplification and hypermutation. It is set in D-dimensional search space with N particles. The current population updates the speed and location of all particles (antibodies) according to Formula (1) and (2).

The IBPSOCC process based on Chaotic Clonal selection is as follows (Fig. 1):

Step 1 Population size N, fitness threshold ε, maximum iteration number G_{maz}, and the learning factor c_1, c_2 are determined.

Step 2 Chaos initializes the population and determines the initial velocity and position of each particle (antibody).

Step 3 According to the particle (antibody) x_i, the affinity of each particle (antibody) is calculated by Formula (12) and the p_i and p_g are updated.

Step 4 The algorithm terminates when the termination condition has been reached.

Step 5 Update the position and speed of all particles (antibodies) according to Formula (1) and (2) and limit them to no more than the boundary.

Step 6 For the current population p_k, according to the affinity order of antibody antigens, M particles (antibodies) with the highest affinity were selected to form the elite clonal population A_k, and the remaining (N–M) particles (antibodies) form the population B_k, which transforms the binary variables into chaotic variables.

Step 7 The chaotic cloning, chaotic mutation, and mutation-selection operators are used to obtain a new generation of elite population A_{k+1}.

Step 8 In the population B_k, d particles (antibodies) with the lowest affinity are discarded by the extinction operator and re-initialized as follows: $x_i' = \Gamma\left(x_i\right)(i = 1, 2, \ldots, d)$, and a new population B_{k+1} is obtained.

Step 9 A new generation of population $p_{k+1} = A_{k+1} \cup B_{k+1}$ is obtained by combining populations A_{k+1} and B_{k+1}. The Chaotic variables are transformed into binary variables.

Step 10 Return to Step3.

4 Experiment result and analysis

To evaluate the performance of the improved strategy and the superiority of the improved algorithm, two groups of comparative experiments were set up. The first group of comparative experiments verified the effectiveness of the chaos mutation operator and the immune selection operator proposed in this research. The second group of comparative experiments compared the improved algorithm of feature selection with three representative method of feature selection to verify the superiority of the proposed method.

4.1 Experiment setup

All the comparative experiments in this paper were run on a PC with 10.0 GB of memory, and the running environment was MATLAB R2016b. All the algorithms used SVM (support vector machines) as classifiers to evaluate the feature selection schemes represented by particles in the population. The size of population was set to 40, the learning factor $c_1 = c_2 = 2$, the inertia weight $\omega = 0.9$, the maximum particle velocity $v_{max} = 4$and the minimum particle velocity $v_{max} = -4$.

Table 1 demonstrates the eight UCI datasets that were utilized in this experiment, including two-class and multi-class datasets, with a minimum number of features of 13 and a maximum of 309.

Table 1 Information of Datasets

Data set	Characteristic number	Sample size	Classification number
Hillvalley	100	606	2
Ionosphere	34	351	2
Libras	90	360	15
Lung Cancer	56	32	3
LVST	309	126	2
Muskl	166	476	2
Sonar	60	208	2
Wine	13	178	2

4.2 Analysis of the experimental results

The first group of experiments were conducted to verify that the chaotic strategy and immune selection operation proposed in this research could effectively improve the PSO performance. In addition to BPSO, the chaotic mutation and immune selection operation are added into BPSO to obtain a BPSO based on chaotic theory (CBPSO), BPSO based on immune selection operation (IBPSO) and BPSO based on chaotic mutation and immune selection operation (IBPSOCC). Four algorithms were experimented with the above eight datasets. The maximum number of iterations was 200. The convergence and the algorithm search performance were evaluated according to the trend of the algorithm fitness value, thus verifying the performance of the two strategies proposed in this paper.

The results of comparative experiments on eight datasets of BPSO, CBPSO, IBPSO and IBPSOCC showed in Fig. 2. The X-axis coordinate denotes iteration times, and the Y-axis coordinate denotes the fitness function value. The trend of polyline graph in Fig. 2 shows that, with the increase of algorithm iteration times, the fitness values all decreased. However, compared with BPSO, CBPSO, and IBPSO, particles with smaller fitness values were found when the search times were the same. In contrast, IBPSOCC could better search particles on eight datasets than the other three algorithms, which fully verifies the efficiency of the improved strategy proposed in this research.

On LVST datasets, Libras datasets, Lung Cancer datasets, and Wine datasets, IBPSOCC rapidly reduced the fitness value of the initial best to a very small level in the initial search of the algorithm, which shows that the immune selection operation proposed in this research can significantly improve the convergence performance of the method. In addition, the analysis of the later search curve also fully verified that the immune selection operation proposed in this paper can still guarantee the effective global search performance of the algorithm after it falls into the local optimal state while continuing to search for a better feature subset than the current one. IBPSOCC can search for particles with smaller fitness than CBPSO and IBPSO on most datasets. Therefore, it is verified that the combination of the two improved strategies proposed in this paper can greatly improve the stability and search performance of the BPSO algorithm.

4.3 Comparison and analysis of the algorithm experimental results

The effectiveness of the improved strategy was verified in 4.2 and compared with other algorithms in 4.3. CPSO, PSO, CBPSO, IBPSOCC and LDW-PSO of LDW-PSO(linear descending strategy particle swarm optimization)were compared. Meanwhile, the parameters were set as follows: PSO algorithm $\omega = 0.75$, $c_1 = c_2 = 2$; CPSO algorithm, CBPSO algorithm, IBPSOCC algorithm and LDW-PSO algorithm

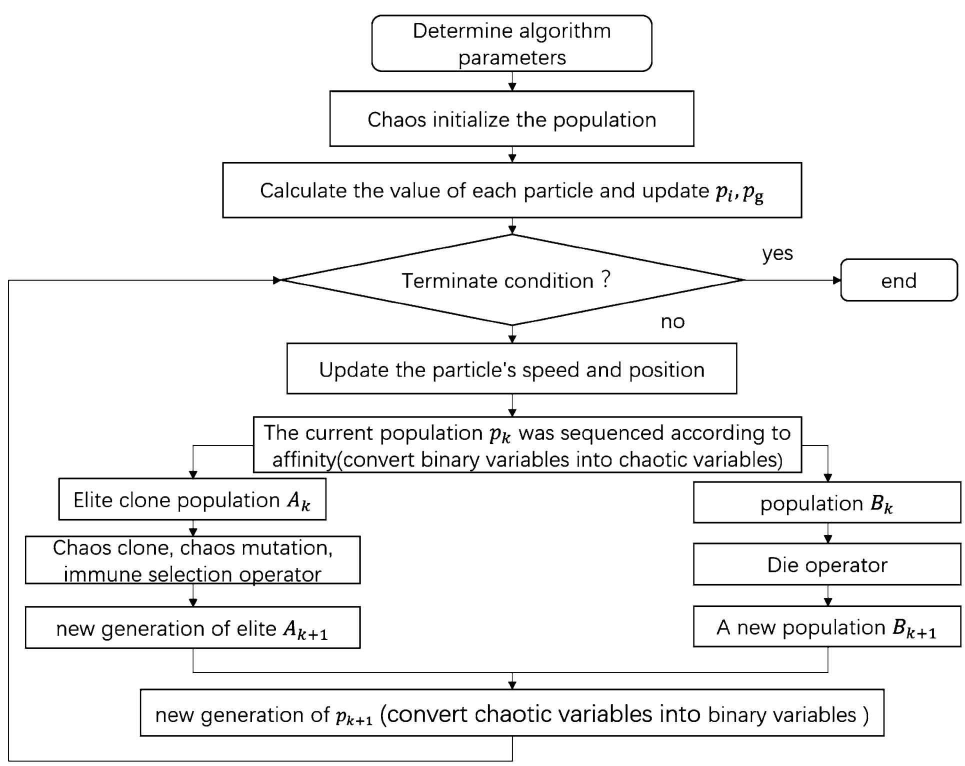
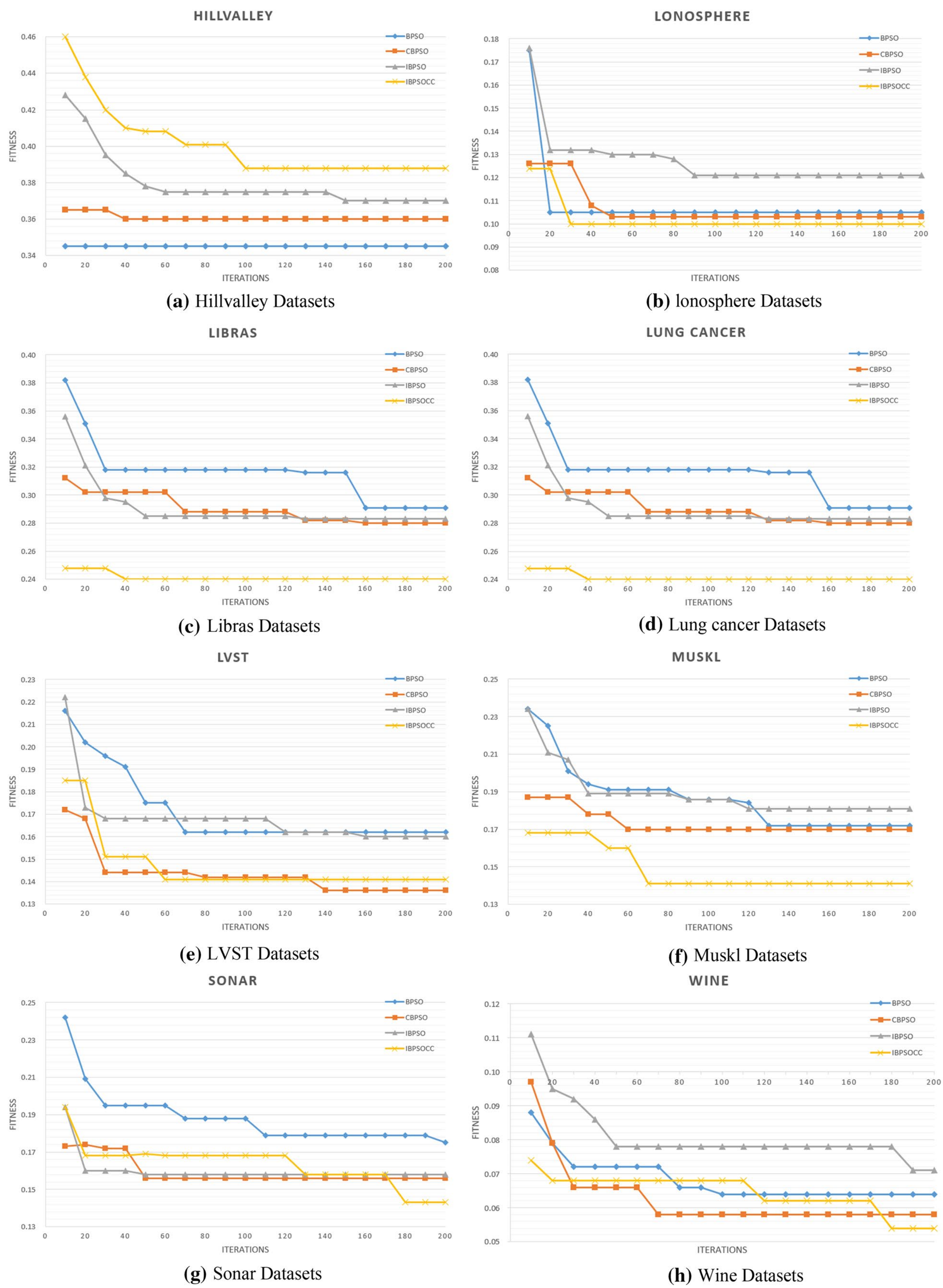


Fig. 1 Flow chart of IBPSOCC algorithm



(a) Hillvalley Datasets

(b) Ionosphere Datasets

(c) Libras Datasets

(d) Lung cancer Datasets

(e) LVST Datasets

(f) Muskl Datasets

(g) Sonar Datasets

(h) Wine Datasets

Fig. 2 The experimental results of the improved strategy on 8 datasets

Table 2 Benchmark functions

Function name	Function expression	x^*	$f(x^*)$	Scope
Sphere	$f_1 = \sum_{i=1}^{l} x_i^2$	$[0, 0, \ldots, 0]$	0	$[-50, 50]$

$\omega_{\max} = 0.9, \omega_{\min} = 0.4, c_1 = c_2 = 2$. The iteration threshold was 2000, then set the number of population particles as 20. CPSO, PSO, CBPSO, IBPSOCC, and LDW–PSO were each ran 30 times. Meanwhile, the difference and the average best fitness were recorded. The final records are shown in Table 3. When the chaotic mutation was performed on the stagnant particles, the initial value of the total number of particles was set to 30. The test function is shown in Table 2.

Table 3 shows that for the functions, compared with five different PSO algorithms in ten dimensions, the average fitness value and variance ranks from low to high are as follows: IBPSOCC algorithm < CBPSO algorithm < CPSO algorithm < LDW-PSO algorithm < PSO algorithm. Compared with five different PSO algorithms in 20 dimensions, the average fitness value and variance ranks from low to high are as follows: CBPSO algorithm < CPSO algorithm < IBPSOCC algorithm < LDW–PSO algorithm < PSO algorithm. Compared with five different PSO algorithms in thirty dimensions, the average fitness value and variance ranks from low to high are as follows: LDW–PSO algorithm < IBPSOCC algorithm < PSO algorithm < CPSO algorithm < CBPSO algorithm.

In summary, an analysis of the above results shows that the IBPSOCC algorithm has certain advantages over the other four methods for 10-dimensional functions, and, in addition, the optimization performance is better, and the stability and accuracy are improved. For the 20-dimensional functions, the CBPSO algorithm has some advantages over the other four methods. The optimization effect is better, and the accuracy and stability are improved. However, for the thirty-dimension functions, the two improved methods had a poor optimization effect compared to the other methods, and the search results were not ideal. This demonstates that the improved algorithm is effective in stability and accuracy.

5 Conclusion and discussion

Due to the fact that PSO is prone to fall into local optimal state, this paper introduces a chaos optimization algorithm and clone selection mechanism in an artificial immune system to timely adjust particles, maintain population diversity, and avoid the algorithm premature convergence. Meanwhile, in combination with chaos theory, the initialization strategy of BPSO and the quality of the initial population is improved. The results from eight datasets demonstate that, the IBPSOCC algorithm has certain advantages over the other four methods for 10-dimensional functions, and, in addition, the optimization performance is better, and the stability and accuracy are improved. For the 20-dimensional functions, the CBPSO algorithm has some advantages over the other four methods. The optimization effect is better, and the accuracy and stability are improved. However, for the thirty-dimension functions, the two improved methods had a poor optimization effect compared to the other methods, and the search results were not ideal. This demonstates that the improved algorithm is effective in stability and accuracy. IBPSOCC algorithm can greatly reduce the redundant features in the feature selection scheme while ensuring the accuracy of feature classification, and the performance of the algorithm is significantly improved.

Table 3 The optimal fitness value in the function in the difference particle swarm algorithm

Sphere	Average fitness value			Variance		
	10	20	30	10	20	30
CPSO	0.3333	1.0020	10.3474	3.3333	8.4940	99.5898
CBPSO	0.2781	1.0008	10.4587	2.7813	8.4206	100.1285
IBPSOCC	0.2677	1.0028	3.3008	2.6771	8.5019	72.2098
PSO	1.1014	2.3894	3.3048	6.3230	18.7823	82.3547
LDW–PSO	0.6715	2.3437	2.6732	6.4304	25.3554	71.3189

Acknowledgements The work is supported in part by GDAS' foundation of "National and Provincial Science and Technology Talent" award winner Grant NO.2018GDASCX-0603, Guangdong Province Public Welfare Research and Capacity Building Project Grant No. 2017A010102017, Guangdong Province Science and Technology Major Projects Grant No. 2017B010116005, Guangdong Province Science and Technology Major Projects Grant No. 2016B090910003, Guangdong Province Applied Science and Technology Research Project Grant No. 2015B090922010. This research was also supported by National Natural Science Foundation of China (No. 61,901,206).

References

Bharti KK, Singh PK (2016) Opposition chaotic fitness mutation based adaptive inertia weight bpso for feature selection in text clustering. Appl Soft Comput 43:20–34

Chen Y-C, Sun H-M, Chen R-S, Yeh J-H, Fan X, Chou I-H (2018) Intelligent radioactive waste process cloud-base system for nuclear power plant decommissioning. J Ambient Intell Hum Comput. https://doi.org/10.1007/s12652-018-0913-1 **(SCI)**

Chen C-M, Wang K-H, Yeh K-H, Xiang B, Wu T-Y (2019) Attacks and solutions on a three-party password-based authenticated key exchange protocol for wireless communications. J Ambient Intell Hum Comput 10(8):3133–3142

Chhikara RR, Sharma P, Singh L (2016) A hybrid feature selection approach based on improved pso and filter approaches for image steganalysis. Int J Mach Learn Cybern 7(6):1195–1206

Chien-Ming C, Xiang B, Liu Y, Wang K-H (2019) A secure authentication protocol for internet of vehicles. IEEE Access 7(1):12047–12057

Dash M, Liu H (1997) Feature selection for classification. Intell Data Anal 1(1–4):131–156

Gheyas IA, Smith LS (2010) Feature subset selection in large dimensionality domains. Pattern Recogn 43(1):5–13

Ghosh I (2018) Probabilistic feature selection in machine learning. In: Rutkowski L, Scherer R, Korytkowski M, Pedrycz W, Tadeusiewicz R, Zurada J (eds) Artificial intelligence and soft computing, vol 10841. Springer, Cham, pp 623–632

Hamedmoghadam H, Jalili M, Yu X (2018) An opinion formation based binary optimization approach for feature selection. Physica A Stat Mech Appl 491:142–152

Huang CL, Dun JF (2008) A distributed pso-svm hybrid system with feature selection and parameter optimization. Appl Soft Comput 8(4):1381–1391

Kennedy J (1998) Particle swarm optimization. In: Proc. of 1995 IEEE Int. Conf. Neural Networks, (Perth, Australia), Nov. 27-Dec

Kou G, Peng Y, Wang G (2014) Evaluation of clustering algorithms for financial risk analysis using mcdm methods. Inf Sci 275(11):1–12

Liao TW, Kuo RJ (2018) Five discrete symbiotic organisms search algorithms for simultaneous optimization of feature subset and neighborhood size of knn classification models. Appl Soft Comput 64:581–595

Lingyun G, Mingquan Y, Changrong W (2017) Cancer classification based on support vector machine optimized by particle swarm optimization and artificial bee colony. Molecules 22(12):2086

Liu H, Motoda H (2007) Computational methods of feature selection (Chapman & Hall/Crc Data Mining and Knowledge Discovery Series). Chapman & Hall/CRC, Boca Raton

Liu G, Guo W, Niu Y, Chen G, Huang X (2015a) A PSO-based-timing-driven Octilinear Steiner tree algorithm for VLSI routing considering bend reduction. Soft Comput 19(5):1153–1169

Liu G, Huang X, Guo W, Niu Y, Chen G (2015b) Multilayer obstacle-avoiding X-architecture steiner minimal tree construction based on particle swarm optimization. IEEE Trans Cybernet 45(5):989–1002

Luo F, Wenzhong G, Yuanlong Y, Guolong C (2016) A multi-label classification algorithm based on kernel extreme learning machine. Neurocomputing 260:313–320

Maryam A, Behrouz MB (2018) Optimizing multi-objective pso based feature selection method using a feature elitism mechanism. Expert Syst Appl 113:499–514

Mo Y, Xing L, Lin Y-K, Guo W (2018) Efficient analysis of repairable computing systems subject to scheduled checkpointing. IEEE Trans Dependable Secure Comput Online Publ. https://doi.org/10.1109/TDSC.2018.2869393

Moradi P, Gholampour M (2016) A hybrid particle swarm optimization for feature subset selection by integrating a novel local search strategy. Appl Soft Comput 43:117–130

Niu Y, Chen J, Guo W (2018) Meta-metric for saliency detection evaluation metrics based on application preference. Multimed Tools Appl Online Publ 77(20):26351–26369

Pan J-S, Lee C-Y, Sghaier A, Zeghid M, Xie J (2019) Novel systolization of subquadratic space complexity multipliers based on toeplitz matrix-vector product approach. IEEE Trans Very Large Scale Integr Syst 27(7):1614–1622

Saber QO, Yahya AZ (2018) Feature selection using particle swarm optimization-based logistic regression model. Chemometrics Intell Lab Syst 182:41–46

Sreeja NK, Sankar A (2015) Pattern matching based classification using ant colony optimization based feature selection. Appl Soft Comput 31:91–102

Tran B, Xue B, Zhang M (2014) Overview of particle swarm optimisation for feature selection in classification. In: 10th international conference, simulated evolution and learning (SEAL) 2014. Springer, 261–276

Tsu-Yang W, Chen C-M, Wang K-H, Meng C, Wang EK (2019) A provably secure certificateless public key encryption with keyword search. J Chin Inst Eng 42(1):20–28

Udhaya Kumar S, Hannah Inbarani H (2016) Pso-based feature selection and neighborhood rough set-based classification for bci multiclass motor imagery task. Neural Comput Appl 28(11):3239–3258

Vieira SM, Mendonça LF, Farinha GJ, Sousa JMC (2013) Modified binary pso for feature selection using svm applied to mortality prediction of septic patients. Appl Soft Comput 13(8):3494–3504

Wang S, Guo W (2017) Robust co-clustering via dual local learning and high-order matrix factorization. Knowl Based Syst 138:176–187

Wang X, Yang J, Teng X, Xia W, Jensen R (2007) Feature selection based on rough sets and particle swarm optimization. Pattern Recogn Lett 28(4):459–471

Wei J, Zhang R, Yu Z, Hu R, Tang J, Gui C et al (2017) A bpso-svm algorithm based on memory renewal and enhanced mutation mechanisms for feature selection. Appl Soft Comput 58(1):176–192

Yang J, Honavar V (1998) Feature subset selection using a genetic algorithm. In: Liu H, Motoda H (eds) Feature extraction, construction and selection. The Springer International Series in Engineering and Computer Science, vol 453. Springer, Boston, MA

Yang Y, Liu X, Zheng X, Rong C, Guo W (2018) Efficient traceable authorization search system for secure cloud storage. IEEE Trans Cloud Comput Online Publ. https://doi.org/10.1109/TCC.2018.2820714

Ye F (2016) Evolving the svm model based on a hybrid method using swarm optimization techniques in combination with a genetic algorithm for medical diagnosis. Multimed Tools Appl 77(3):3889–3918

Zhang Y, Wang S, Phillips P, Ji G (2014) Binary pso with mutation operator for feature selection using decision tree applied to spam detection. Knowl Based Syst 64:22–31

Zhu W, Guo W, Yu Z, Xiong H (2018) Multitask allocation to heterogeneous participants in mobile crowd sensing, Wirel Commun Mobile Comput 10 pages, Article 7218061,

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