

Research Article

The Optimization of Feature Selection Based on Chaos Clustering Strategy and Niche Particle Swarm Optimization

Longzhen Duan,¹ Shuqing Yang ,^{1,2} and Dongbo Zhang³

¹*School of Information Engineering, Nanchang University, Nanchang 330031, China*

²*School of Software, Jiangxi University of Science and Technology, Nanchang 330013, China*

³*Guangdong Key Laboratory of Modern Control Technology, Guangdong Institute of Intelligent Manufacturing, Guangzhou 510070, China*

Correspondence should be addressed to Shuqing Yang; ysqsinuo2010@163.com

Received 2 March 2020; Revised 27 May 2020; Accepted 13 June 2020; Published 9 July 2020

Academic Editor: George A. Papakostas

Copyright © 2020 Longzhen Duan et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the rapid increase of the data size, there are increasing demands for feature selection which has been a powerful tool to handle high-dimensional data. In this paper, we propose a novel feature selection of niche particle swarm optimization based on the chaos group, which is used for evaluating the importance of feature selection algorithms. An iterative algorithm is proposed to optimize the new model. It has been proved that solving the new model is equivalent to solving a NP problem with a flexible and adaptable norm regularization. First, the whole population is divided into two groups: NPSO group and chaos group. The two groups are iterated, respectively, and the global optimization is updated. Secondly, the cross-iteration of NPSO group and chaos group avoids the particles falling into the local optimization. Finally, three representative algorithms are selected to be compared with each other in 10 UCI datasets. The experimental results show that the feature selection performance of the algorithm is better than that of the comparison algorithm, and the classification accuracy is significantly improved.

1. Introduction

Feature selection has been widely researched and a large number of algorithms have been developed. These algorithms have been successful in solving the real-world problems such as medical image processing [1], malware detection [2, 3], customer churn prediction [4], music retrieval [5], text categorization [6, 7], intrusion detection [8], gene microarray analysis [9], and stock trend prediction [10], including image retrieval [11] and information retrieval [12]. It is also broadly studied as a data preprocessing technology in the field of machine learning and data mining [13]. In the process of machine learning and data mining, the performance of the learning algorithm is affected by the large number of redundant and noisy features in the processed dataset. The purpose of feature selection is to eliminate redundant features and noise features, search effective features from the original feature set to form feature subset,

and reduce the time and space complexity of learning algorithm. Feature selection has been proved to be a NP hard combinational optimization problem. There is no polynomial algorithm to solve it accurately. So researchers are committed to using heuristic search algorithm to solve the optimization problem.

As a typical heuristic search algorithm, genetic algorithm has made some achievements in feature selection research. Siedlecki and Sklansky took the lead in applying genetic algorithm to feature selection in large-scale datasets and opened the research of feature selection based on genetic algorithm [14]. Majid and Nicolas used filter feature selection technology to apply genetic algorithm to feature selection of satellite precipitation estimation [15]. Wang et al. introduced the selection operator of the space of preserving the best and eliminating the bad and the splicing and cutting operator in the genetic algorithm and then applied them to the feature selection [16].

A novel feature selection algorithm, which is based on the method [17] called high dimensional model representation, is proposed. The proposed algorithm is tested on some toy examples and hyperspectral datasets in comparison with conventional feature-selection algorithms in terms of classification accuracy, stability of the selected features, and computational time. The proposed approach has a feature ranking module to identify relevant features and a clustering module to eliminate redundant features using linear correlation-based multi-filter feature selection achieved the best classification accuracy [18]. Some of the most popular methods for selecting significant features are presented and a comparison between them is provided. Their advantages and disadvantages are outlined in order to provide a clearer idea of when to use each one of them for saving computational time and resources [19].

The main feature selection methods can be grouped into three groups: supervised, unsupervised, and semi-supervised. In the studies, we focus on supervised feature selection for optimization and use the terms class labels and target variable interchangeably. These methods are categorized into two main groups: subset evaluation and individual evaluation methods. The relevance of each variable is measured by individual evaluation with the target variable and assigns importance or rank according to its relevance, while subset evaluation selects a subset of variables for model construction based on some search strategy. Besides this optimization, these methods are categorized into filters, wrappers, and embedded and hybrid approaches based on their selection strategy [20]. Some more details for feature selection are discussed in some latest excellent literatures [21–23]. The majority of the existing studies have focused on improving individual methods, which have been proved to be effective in feature selection for training and testing data [24]. However, researchers suggest that there is no “one fit all” solution and the majority of the efforts are focused on finding an optimal solution for specific problem settings. Therefore, new methods are constantly appearing using different approaches. These studies for feature selection have obtained good results within the range of their respective conditions.

In the paper, the strategy of feature selection and particle swarm optimization is introduced. Then, we describe NPSO based feature selection approaches, which is a very significant part of our research.

The concept of niche particle swarm of feature selection is expanding and a novel framework is developed that takes into account feature relevance and also provides a flexible mechanism for obtaining a balance between accuracy and efficiency.

2. Related Work

2.1. Strategy of Feature Selection. The definition of feature selection given by Dash et al. is to select as small a subset of features as possible, and meet the two conditions of not significantly reducing classification accuracy and not significantly changing class distribution, and propose a basic framework of feature selection, as shown in Figure 1. The

paper develops the research on the basis of this framework, which uses the niche particle swarm optimization algorithm of chaotic group as the feature search strategy, and the classification error rate and the number of features as the feature subset evaluation strategy.

2.2. Particle Swarm Optimization Algorithm. The basic concept of niche particle swarm optimization is that different particles do not have information interaction and independent evolution, so that the particles are in a separate and isolated environment, and the particles in the isolated environment in the algorithm reduce the convergence. If the value of a particle’s fitness function is basically unchanged in the continuous iteration process, take the particle as the center and the distance from the nearest particle as the radius to form a circular region niche. If the particle enters the circular region, it will be absorbed and the new niche will be merged. The niche in the algorithm can be described by the following formula:

$$T_{s,j} = \max S_{s,j}^g - S_{s,j}^i. \quad (1)$$

If the particles satisfy the following formula, different actions will be taken:

$$\|S_{s,j}^k - S_{s,j}^i\| \leq T_{s,j}, \quad (2)$$

$$\|S_{s,j}^k - S_{s,j}^i\| \leq T_{s,j} + T. \quad (3)$$

Formula (2) is satisfied, the niche particles absorb particles, formula (3) is satisfied, and the niche intersections are merged. The velocity update formula of particles is as follows:

$$v_i^{k+1} = -x_{g_{best}}^k + x_{p_{best}}^k + \omega v_{g_{best}}^k + \eta^k (1 - r_2^k), \quad (4)$$

$$x_i^{k+1} = x_{p_{best,i}}^k + \omega v_{g_{best}}^k + \eta^k (1 - r_2^k), \quad (5)$$

where η is the minimum value of the search range and r_2^k is the evenly distributed variable between 0 and 1.

2.3. Feature Selection Approaches. Feature selection is a data preprocessing method, which uses some evaluation criteria to select feature subsets from the original feature space. Scholars have defined feature selection from the perspectives of whether feature subsets can identify targets, can reduce prediction accuracy, and can change the distribution of original data classes. According to the search method of feature subset, the search strategy can be divided into global optimal search, sequence search, and random search. Sequential search and random search are called heuristic search strategies, which include four search strategies corresponding to four search starting points: forward search, backward search, bidirectional search, and random search.

Global optimal search is to find the global optimal subset of the original feature sets. Only the enumeration method and branch and bound method can be realized [25]. With the increase of dimension, the time complexity of the two

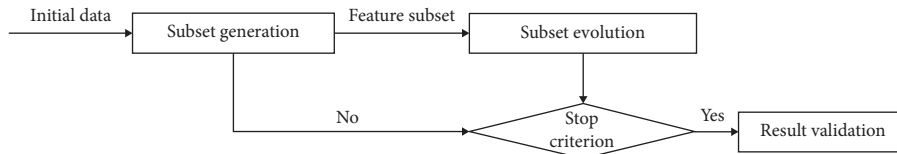


FIGURE 1: The basic framework of feature selection.

strategies increases exponentially, which is a NP hard problem.

The algorithm of sequence search can be divided into three categories: forward search, backward search, and bi-directional search. Sequence forward search (SFS) is a greedy method to add the feature with the highest score to the selected feature subset. Sequence forward floating search (SFSS) and generalized sequence forward search (GSFS) are its improved strategies. The sequence backward search (SBS) of backward search removes one feature from the selected feature subset at a time. The improved strategies include sequence backward floating search (SBFS), generalized sequence forward search (GSFS), floating generalized backward search (FGSBS), etc. Bidirectional search is a combination strategy of forward search and backward search, which can add and delete features, including plus q and minus r algorithm, generalized plus q and minus r algorithm, etc. The feature selection of random search strategy is random, with strong uncertainty. The feature subsets selected this time and next time are quite different. However, the transformation range of heuristic rule subsets gradually slows down and gradually approaches the optimal feature subsets. Random search has a certain chance to make the algorithm jump out of the local optimum, that is, to prevent falling into the local optimum and find the approximate optimal solution. Therefore, in general, random search strategy selection of the feature subset is better than sequence search. Common random search methods include simulated annealing (SA), differential evolution (DE), ant colony optimization (ACO), genetic algorithm (GA), quantum evolutionary algorithm (QEA), harmony search algorithm (HSA), particle swarm optimization (PSO), crawling Mountain search, artificial immune system, tabu search algorithm, beam search, artificial bee colony, etc.

2.4. NPSO-Based Feature Selection Approaches. Qiu et al. proposed a competitive multi-objective particle swarm optimization algorithm (D-CMOPSO) based on diversity and proposed the learning mechanism of diversity competition and the initialization strategy of maximum information coefficient, so that it has strong competitiveness in feature selection [26]. Zhang et al. proposed a new unsupervised feature selection method based on particle swarm optimization algorithm. One is the spatial reduction strategy based on average mutual information, which is used to quickly remove the irrelevant and weak coherent features. The other is the local filtering search strategy based on feature redundancy, which improves the search ability of the group. The evaluation function of particle swarm optimization based on feature similarity and the parameter

updating strategy of particle swarm optimization are proposed to verify the superiority and effectiveness of the algorithm [27]. Wang and Feng et al. proposed a hybrid feature selection method based on multi-filter weights and multi-feature weights [28]. Firstly, all samples are normalized and discretized; secondly, the vector of multi-filter weight and the matrix of multi-feature weight are calculated to get different feature subset sets; finally, a feature association calculation method based on Q space is proposed to measure the relationship between different features, and greedy search strategy is used to filter. The method not only improves the classification accuracy, but also improves the running speed. Xiong et al. proposed a population initialization strategy based on chaos theory and proposed chaos clone operator, chaos mutation operator, and immune selection operator in the algorithm. Experimental results show that the improved algorithm is better than other algorithms [29, 30].

2.5. Other Evolutionary Algorithms for Feature Selection. Hussien et al. proposed a new binary whale optimization algorithm (bWOA) to select the best feature subset for dimension reduction and classification. Based on the S-shaped transfer function, the free position of beluga is transformed into the corresponding binary solution. The algorithm has remarkable performance in optimizing the optimal features [23]. Aljarah et al. proposed two improved WOA algorithms: one is to use EOBL at the initial stage, and the other is to merge evolutionary operators of differential evolutionary algorithm including mutation, crossover, and selection operators at the end of each WOA iteration. In addition, information gain is used as the filtering technology of feature selection. The results show that the accuracy is better than other algorithms [31]. Yang et al. proposed a new unsupervised feature selection method, which can remove redundant features of HSI by feature subspace decomposition and feature combination optimization and decompose feature subset by fuzzy C-means (FCM) algorithm [32]. The optimal feature selection is based on the optimization process of grey wolf optimizer (GWO) algorithm and maximum entropy (ME) principle [33]. The results show that the performance is improved.

3. A Framework and Pseudocode

In this section, a chaos group niching particle swarm optimization is proposed, which is a multi-objective algorithm. The algorithm proposed in this paper combines the NPSO algorithm with the chaos mechanism and applies it to feature selection. The improved idea is to divide the whole

population into two groups: NPSO and chaos. According to the early maturity decision strategy, the population is optimized in two stages.

3.1. Chaos Optimization. Chaos refers to an irregular activity caused by a certain system. Although chaos does not have the characteristics of symmetry and periodicity, some ordered states can be found from the inside of chaos, such as order in structure and constant in random motion. Chaos is characterized by ergodicity, randomness, and regularity. The basic idea of using chaos theory to optimize is to first use the above characteristics to establish the relationship between chaos space and solution space, that is, the mapping relationship. The regression equation of logistic model is as follows:

$$x_{n+1} = x_n \times \mu \times (1 - x_n), \quad (6)$$

where x_n represents chaos domain; when $x_n \in (0, 1)$ and $x_n \notin (0.25, 0.5, 0.75)$, the system is in chaos region. μ is a constant, $\mu \in (0, 4]$. The results show that when $\mu \in (3.5699456, 4]$, the system is in a chaotic state, the sequence generated by logistic is random, and the particle's motion track in the interval shows chaotic characteristics. At this time, x_n will traverse $(0, 1)$, and the determined chaotic sequence x_1, x_2, \dots, x_n can be generated, so the chaotic motion can be used to traverse the whole world, so as to search for the optimal solution in all solution spaces.

3.2. Group Search Strategy. In this section, although the current NPSO algorithm and the corresponding improved algorithm have achieved good results, the search accuracy of NPSO and some improved PSO is low, and the population convergence speed is slow in the later stage. An improved NPSO algorithm is proposed in this paper.

Because of the ergodic property of chaos, chaos algorithm is easy to jump out of the local optimum, which is conducive to the global search of the algorithm. NPSO has the advantages of simple algorithm, fast convergence speed, and strong search ability. In this paper, the advantages of the two algorithms are fully utilized. Firstly, according to the different ways of particle search, the whole particle swarm is divided into two groups: NPSO group and chaos group, named NP group and C group. NP group update particle velocity position according to formulas (4) and (5), where $v_{g_{best}}^k$ is the optimal value of K time of the whole particle in formula (4).

According to the strategy of precocity in formula, the search process is divided into two stages and NP group and C group are iterated according to NPSO search mechanism and chaos mechanism, respectively. In the first stage, NP group and C group iterate at the same time, comparing the extreme values of NP group and C group, and updating the extreme value of the optimal group. Iterative search is performed to reduce the possibility of particles falling into precocious state. If the particles do not fall into precocious state, only execute the first stage search; otherwise, perform the second stage search. In the second stage, NP and C

particles cross-iterate with each other. At this time, the NP group particles give the C group chaos search according to the fitness value and replace the NP group particles with the better part of the fitness value of the C group particles, so as to effectively avoid the local optimization of the NP group particles.

In this paper, the strategy of group fitness variance is used to judge whether particles fall into premature state. The group fitness variance is as follows:

$$x_{n+1} = x_n \times \mu \times (1 - x_n), \quad (7)$$

$$\sigma^2 = \frac{1}{n} \sum_{i=1}^n \left(\frac{f_i - f_{avg}}{f} \right)^2, \quad (8)$$

$$f = \max \left(1, \max |f_i - f_{avg}| \right).$$

In the formula, n represents total number of particles in NP group, f_i is the individual fitness of the i th particle, and f_{avg} is the total average fitness of NP group. The smaller the variance of group fitness σ^2 is, the more convergent the NP group tends to be. In this paper, setting a threshold for σ^2 , when σ^2 is less than the threshold, the algorithm will fall into premature state. At the same time, the optimal fitness threshold is set to prevent the global optimal from being misjudged as premature convergence.

3.3. Adaptive Inertia Weight. In this section, considering the importance of the value of inertia weight ω in formulas (4) and (5), the inertia weight of the algorithm should be large in the early stage of search, which is conducive to global search, but not easy to get accurate solution. In the later stage of search, the inertia weight should be smaller, which is beneficial to local mining, but easy to fall into local optimum. In order to improve the optimization performance of the algorithm, the particle inertia weight ω is used to adjust the size of the strategy according to the current adaptive value, that is, the adaptive adjustment of the inertia weight strategy. The formula is as follows:

$$\omega = (\omega_{max} - \omega_{min}) \exp \left(- \left(\tau \frac{cur}{Loop} \right)^2 \right) + \omega_{min}, \quad (9)$$

where ω_{max} and ω_{min} are maximum and minimum, cur is the current number of iterations, $Loop$ is the maximum number of iterations, and τ is the empirical coefficient; the value is between 20 and 55.

Because of the negative exponential term in the formula, the initial value cur is smaller, the inertia weight ω is larger, and particles update speed and position in the whole solution space, while later the value of cur is larger, the inertia weight ω is smaller, and particles update speed and position in a small range, so the adjustment strategy ensures the diversity and convergence of group solution.

3.4. Fitness Function. In this section, a feature selection method based on CGNPSO will be introduced, with the overall classification performance as the fitness function. A

new fitness function is proposed to further improve the classification performance and reduce the number of selected features. The fitness function is to minimize the classification error rate and maximize the classification accuracy. The classification error rate is calculated according to

$$\text{Error rate} = \frac{\text{FP} + \text{FN}}{\text{FP} + \text{FN} + \text{TP} + \text{TN}}, \quad (10)$$

where FP indicates the number of predicted errors of negative samples, FN indicates the number of predicted errors of positive samples, TP indicates the number of predicted errors of positive samples, and TN indicates the number of predicted errors of negative samples.

CGNPSO algorithm can regard feature selection as a double objective optimization problem. The purpose is to balance the performance and the number of features. It not only solves the minimum classification error rate, but also solves the minimum number of features. Therefore, on the basis of equation (9), it is proposed to increase the weighting of the number of features, which is calculated according to

$$\text{Fitness}(i) = \alpha \times \frac{\text{number of feature}}{\text{all of features}} + (1 - \alpha)\text{error rate}(i). \quad (11)$$

In the formula, Number of Features represents the number of features of the i th particle, All of Features represents the number of all features, and ErrorRate (i) represents the classification error rate of the i th particle. α represents the correlation between the number of features and the classification error rate, $\alpha \in (0, 1]$; the value is 0.2.

3.5. Flow of Algorithm. p_{best} and $NP_{g_{\text{best}}}$ represent the individual and global extremum of NP group, $C_{g_{\text{best}}}$ is the global extremum of C group, and g_{best} indicates the global extremum of all groups.

3.5.1. Step 1: Initialization. Set the initial population number m , fitness threshold ε , maximum number of iterations G_{max} , and learning factor c_1, c_2 . A random n -dimensional vector $Z_1 = (z_{11}, z_{12}, \dots, z_{1n})$ with components between 0 and 1 is generated. Taking Z_1 as the initial value, N vectors Z_1, Z_2, \dots, Z_N are obtained by iteration according to formula (7). The resulting vector is introduced into the range of optimization variables. The fitness values of each vector are calculated by the objective function, and the first m fitness values are selected as the initial positions of the two sub-swarm particles.

3.5.2. Step 2: Update Position and Extremum. NP group updates the positions of particles by formulas (4) and (5). Group C iteratively obtains n vectors Z_1, Z_2, \dots, Z_N according to formula (7) and linearly transforms them to solution space to update particle position. Adjust the particle inertia weight coefficient according to formula (9); substitute the particle position vector into the objective function to calculate the fitness value of NP group and C group. Update

$p_{\text{best}}, NP_{g_{\text{best}}}, C_{g_{\text{best}}}$, compare the sizes of $NP_{g_{\text{best}}}, C_{g_{\text{best}}}$, and update g_{best} .

3.5.3. Step 3: Determine Whether to End the Iteration. Repeat Step 2 to satisfy the convergence requirements or the maximum number of iterations, then turn to update $p_{\text{best}}, NP_{g_{\text{best}}}, C_{g_{\text{best}}}$, compare the sizes of $NP_{g_{\text{best}}}, C_{g_{\text{best}}}$, and update g_{best} . If the particle falls into precocious state, go to Step 4.

3.5.4. Step 4: Position and Extremum of Second Update. NP group and C group are sorted by particle fitness, and in NP group, the first N particles are selected to replace the worst N particles in C group for chaos optimization. The classification of each particle in group C is mapped to the logistic equation, the chaotic variable is obtained, and the chaotic variable is used to search. Finally, the chaotic sequence is inversely mapped to the original solution space. Refer to Step 2 to update particle velocity and position of NP group particles. Substitute particle position vector into objective function to calculate fitness of NP group and C group. Update $p_{\text{best}}, NP_{g_{\text{best}}}, C_{g_{\text{best}}}$, compare the sizes of $NP_{g_{\text{best}}}, C_{g_{\text{best}}}$, and update g_{best} .

3.5.5. Step 5: Judge Whether to End the Second Iteration. If the number of iterations of population reaches the maximum or the optimal position of population search satisfies the convergence accuracy, update $p_{\text{best}}, NP_{g_{\text{best}}}, C_{g_{\text{best}}}$, compare the sizes of $NP_{g_{\text{best}}}, C_{g_{\text{best}}}$, and update g_{best} . If not, go to Step 4.

3.5.6. Step 6: End of Iteration. At the end of iteration, the global optimal solution and objective function value are output.

4. Experiment and Results

4.1. Experimental Scheme. In this section, CGNPSO algorithm is compared with two algorithms, NSGA-II and MOEA/D. The population size of all multi-objective evolutionary algorithms is set to 50, and the number of iterations is set to 100. In NSGA-II and MOEA/D, single point crossover and random reversal mutation were used, and the mutation probability was set as $1/d$.

In this experiment, 10 datasets in UCI machine learning database are selected. Their names, characteristic numbers, category numbers, and sample numbers are shown in Table 1. Among them, the number of features and samples in the first 6 datasets is less, making the solution less difficult; the number of features and samples in the last 4 datasets is more, making the solution more difficult. After standardizing the dataset, it is divided into training set and test set; 70% of data samples are training sets and 30% of data samples are test sets.

4.2. Experimental Results and Analysis. In this section, Table 2 lists the feature selection results of the three algorithms

TABLE 1: Datasets.

Dataset	Number of features	Number of classes	Number of instances
German	24	2	1000
WBCD	30	2	569
Ionosphere	34	2	351
Wine	13	3	178
Lung cancer	56	3	32
Libras	90	15	360
HillValley	100	2	606
Musk1	166	2	476
Madelon	500	2	2600
Isolet5	617	2	1559

TABLE 2: Experimental results.

Dataset	Method	Ave-size	Error rate (%)
German	All	24	32
	NSGA-II	12.5	31.27
	MOEA/D	8.8	31.09
	CGNPSO	8.58	31
WBCD	All	30	14
	NSGA-II	15	7.2
	MOEA/D	7.6	7.2
	CGNPSO	7.1	7.02
Ionosphere	All	34	17.1
	NSGA-II	15	12
	MOEA/D	10	10.8
	CGNPSO	8.8	10.48
Wine	All	13	37.8
	NSGA-II	8.5	4.04
	MOEA/D	8.0	3.77
	CGNPSO	5.0	3.06
Lung cancer	All	56	30
	NSGA-II	26.5	27.5
	MOEA/D	24	26.7
	CGNPSO	22.01	26.5
Libras	All	90	31.1
	NSGA-II	45	27.8
	MOEA/D	32.7	25.5
	CGNPSO	31.1	25.46
HillValley	All	100	48.5
	NSGA-II	48	42.5
	MOEA/D	37.2	42.7
	CGNPSO	37.05	42.4
Musk1	All	166	16.08
	NSGA-II	84.1	15.12
	MOEA/D	78.9	14.36
	CGNPSO	80.1	14.4
Madelon	All	500	29.1
	NSGA-II	248.9	22.17
	MOEA/D	240.1	22.98
	CGNPSO	239.35	22.88
Isolet5	All	617	1.55
	NSGA-II	309.5	1.51
	MOEA/D	300.2	1.49
	CGNPSO	301.1	1.5

on the dataset. First of all, the test results of NSGA-II are analyzed, and the classification error rate of all datasets is better than that of all features, and the number of features is

less than the total number of features. In the test dataset, NSGA-II can reduce the number of features to 50% of the total.

In the test of each dataset, a feature subset whose classification error rate is obtained in MOEA/D algorithm is less than that of using all features. MOEA/D reduces the average feature number of feature set to 40% of the total feature number. Among them, MOEA/D results are better than other algorithms in datasets Musk1 and Isolet5.

In the experimental results of CGNPSO, at least one feature subset with a classification error rate lower than that with all features can be obtained in the test of each dataset. The feature number of CGNPSO average feature set is reduced to 36% of the total. In Table 2, it can be seen that CGNPSO can get lower characteristic number in most cases. When the characteristic number is equal, CGNPSO can select the characteristic combination with lower error rate. Especially in ionosphere and wine datasets, CGNPSO can get much lower characteristic number and error rate than NSGA-II and MOEA/D. However, in Musk1 and Isolet5 datasets, CGNPSO obtained higher characteristic number and error rate than MOEA/D.

In the German datasets, the performance of CGNPSO is slightly superior to MOEA/D, but significantly better than the method of NSGA-II in the number of feature section; however, the improvement is not obvious in error rate. The similar circumstances happened in other datasets with fewer feature selection numbers. In Musk1 and Isolet5, MOEA/D is superior to CGNPSO in a few solutions, and CGNPSO's solution set in other tests is superior to MOEA/D. The results show that CGNPSO has group diversity, to some extent, which improves the optimization ability of feature number and classification performance.

5. Conclusion

In this paper, niche particle swarm optimization (NPSO) algorithm is applied to feature selection. Aiming at the shortcomings of NPSO in solving complex optimization problems, the algorithm is improved. On this basis, a feature selection algorithm, CGNPSO, based on chaotic clustering is proposed. The population is divided into two groups: NPSO group and chaos group. When the particle iteration appears premature, the cross-iteration between NPSO group and chaos group contributes to avoiding falling into local

optimum. The experimental results show that the algorithm is effective, and the efficiency of feature selection is better than the comparison algorithm. Experimental results confirm that the proposed method selects features, which are more informative and diverse resulting in improved accuracy of the prediction model. However, in the process of experiment, the algorithm in this paper will converge to suboptimal individuals, which needs further research in future work.

Data Availability

The data used to support the findings of this study can be downloaded from UCI machine learning database. The website is as follows: <https://archive.ics.uci.edu/ml/index.php>.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This paper was supported by the National Natural Science Foundation of China (61972184 and 61562032), Modern Agricultural Research Collaborative Innovation Project of Jiangxi (JXXTCXQN201906), GDAS' Project of Building a World-Class Research Institution (2020GDASYL-20200402007), GDAS' Project of Science and Technology Development (2018GDASCX-0115), and GDAS' Project of Science and Technology Development (2017GDASCX-0115).

References

- [1] V. Szilárd, K. Alexandros, J. Stefan et al., "Feature selection for automatic tuberculosis screening in frontal chest radiographs," *Journal of Medical Systems*, vol. 42, no. 8, p. 146, 2018.
- [2] B. M. Khammas, A. Monemi, J. Stephen Bassi et al., "Feature selection and machine learning classification for malware detection," *Jurnal Teknologi*, vol. 77, no. 1, pp. 243–250, 2015.
- [3] Q. Jiang, X. Zhao, and K. Huang, "A feature selection method for malware detection," in *Proceedings of the 2011 IEEE International Conference on Information and Automation*, pp. 890–895, Shenzhen, China, June 2011.
- [4] K. Huang, S. Anwar, and A. Adnan, "Customer churn prediction in the telecommunication sector using a rough set approach," *Neurocomputing*, vol. 237, pp. 242–254, 2017.
- [5] J. Pickens, "A survey of feature selection techniques for music information retrieval," in *Proceedings of the 2nd Annual International Symposium on Music Information Retrieval 2001*, pp. 1–6, Bloomington, IN, USA, September 2001.
- [6] M. E. Ruiz and P. Srinivasan, "Hierarchical text categorization using neural networks," *Information Retrieval*, vol. 5, no. 1, pp. 87–118, 2002.
- [7] S. Wang, J. Cai, Q. Lin, and W. Guo, "An overview of unsupervised deep feature representation for text categorization," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 3, pp. 504–517, 2019.
- [8] K. C. Khor, C. Y. Ting, and S. P. Amnuaisuk, "A feature selection approach for network intrusion detection," in *Proceedings of the International Conference on Information Management & Engineering*, pp. 133–137, IEEE, Kuala Lumpur, Malaysia, April 2009.
- [9] A. Nowe, J. Taminau, and S. Meganck, "A survey on filter techniques for feature selection in gene expression microarray analysis," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 9, no. 4, pp. 1106–1119, 2012.
- [10] Y. Xu, Z. Li, and L. Luo, "A study on feature selection for trend prediction of stock trading price," in *Proceedings of the 2013 International Conference on Computational and Information Sciences*, pp. 579–582, Shiyang, China, June 2013.
- [11] J. G. Dy, C. E. Brodley, and A. M. Aisen, "Unsupervised feature selection applied to content-based retrieval of lung images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 3, pp. 373–378, 2003.
- [12] O. Egozi, E. Gabrilovich, and S. Markovitch, "Concept-based feature generation and selection for information retrieval," in *Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence, AAAI 2008*, pp. 1132–1137, Chicago, Illinois, USA, July 2008.
- [13] M. Dash, "Feature selection via set cover," in *Proceedings of the 1997 IEEE Knowledge and Data Engineering Exchange Workshop*, Newport Beach, CA, USA, November 1997.
- [14] W. Siedlecki and J. Sklansky, "A note on genetic algorithm for large scale feature selection," *Pattern Recognition Letters*, vol. 10, no. 11, pp. 335–347, 1989.
- [15] M. Majid and H. Y. Nicolas, "On the use of the genetic algorithm filter-based feature selection technique for satellite precipitation estimation," *IEEE Geoscience and Remote Sensing Letters*, vol. 9, no. 5, pp. 963–967, 2012.
- [16] L. Wang, Y. Dong, and J. Gu, "Improved Elite Genetic Algorithm and its application in feature selection," *Computer Engineering and Design*, vol. 35, no. 5, pp. 1792–1796, 2014.
- [17] L. Taskin, H. Kaya, and L. Bruzzone, "Feature selection based on high dimensional model representation for hyperspectral images," *IEEE Transactions on Image Processing*, vol. 26, no. 6, pp. 2918–2928, 2017.
- [18] A. Haq, D. Zhang, He Peng, and S. Rahman, "Combining multiple feature-ranking techniques and clustering of variables for feature selection," *IEEE Access*, vol. 7, 2019.
- [19] Z. M. Hira and D. F. Gillies, "A review of feature selection and feature extraction methods applied on microarray data," *Advances in Bioinformatics*, vol. 2015, Article ID 198363, 13 pages, 2015.
- [20] A. Bagheri, M. Saraee, and F. D. Jong, "Sentiment classification in persian: introducing a mutual information-based method for feature selection," in *Proceedings of the 2013 21st Iranian Conference on Electrical Engineering (ICEE)*, Mashhad, Iran, May 2013.
- [21] N. Neggaz, E. H. Houssein, and K. Hussain, "An efficient henry gas solubility optimization for feature selection," *Expert Systems with Applications*, vol. 152, p. 113364, 2020.
- [22] E. H. Houssein, M. E. Hosney, D. Oliva, W. M. Mohamed, and M. Hassaballah, "A novel hybrid Harris hawks optimization and support vector machines for drug design and discovery," *Computers & Chemical Engineering*, vol. 133, Article ID 106656, 2020.
- [23] A. G. Hussien, A. E. Hassanien, E. H. Houssein et al., "S-shaped binary whale optimization algorithm for feature selection," *Recent Trends in Signal and Image Processing*, Springer, Berlin, Germany, 2019.
- [24] A. M. Mihaela, W. Shicai, and G. Yike, "Combining multiple feature selection methods and deep learning for high-dimensional data," *Transactions on Machine Learning and Data Mining*, vol. 9, no. 1, pp. 27–45, 2016.

- [25] J. Somol, P. Pudil, and J. Kittler, "Fast branch & bound algorithms for optimal feature selection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 7, pp. 900–912, 2004.
- [26] J. Qiu, F. Cheng, L. Zhang et al., "A diversity based competitive multi-objective pso for feature selection," in *Intelligent Computing Theories and Application*, pp. 26–37, Springer, Berlin, Germany, 2019.
- [27] C. Peng, H. G. Li, and Q. Wang, "A filter-based bare-bone particle swarm optimization algorithm for unsupervised feature selection," *Applied Intelligence*, vol. 49, no. 8, pp. 2889–2898, 2019.
- [28] Y. Wang and L. Feng, "A new hybrid feature selection based on multi-filter weights and multi-feature weights," *Applied Intelligence*, vol. 49, no. 6, pp. 4033–4057, 2019.
- [29] L. Xiong, R. Chen, X. Zhou et al., "Multi-feature fusion and selection method for an improved particle swarm optimization," *Journal of Ambient Intelligence and Humanized Computing*, 2019.
- [30] L. Xiong, D. Zhang, K. Li, and L. Zhang, "The extraction algorithm of color disease spot image based on Otsu and watershed," *Soft Computing*, vol. 24, no. 10, pp. 7253–7263, 2019.
- [31] I. Aljarah, M. A. M. Abushariah, and N. Idris, "Improved whale optimization algorithm for feature selection in Arabic sentiment analysis," *Applied Intelligence*, vol. 49, no. 5, pp. 1688–1707, 2018.
- [32] J. Yang, C. Lei, and F. Li, "Unsupervised hyperspectral feature selection based on fuzzy c-means and grey wolf optimizer," *International Journal of Remote Sensing*, vol. 40, no. 9, pp. 3344–3367, 2019.
- [33] D. Zhang, G. Yin, X. Jin et al., "Two-stage and bi-direction feature selection method for EEG channel based on CSP and SFFS-SFBS," *Dongnan Daxue Xuebao (Ziran Kexue Ban)/Journal of Southeast University (Natural Science Edition)*, vol. 49, no. 1, pp. 125–132, 2019.