S & M 2535

# Hybrid Algorithm Based on Simulated Annealing and Bacterial Foraging Optimization for Mining Imbalanced Data

Chou-Yuan Lee,[1*] Zne-Jung Lee,[1] Jian-Qiong Huang,[1] Fu-Lan Ye,[1]
Jie Yao,[1] Zheng-Yuan Ning,[1] and Teen-Hang Meen[2**]

[1]School of Technology, Fuzhou University of International Studies and Trade, Fuzhou 350202, China
[2]Department of Electronic Engineering, National Formosa University, Huwei 632 Yunlin, Taiwan

(Received October 21, 2020; accepted February 12, 2021; online published March 16, 2021)

***Keywords:*** bacterial foraging optimization, simulated annealing, hybrid, chemotaxis, imbalanced data

The bacterial foraging optimization (BFO) algorithm can simulate the mechanism of natural selection. However, as the direction of inversion is uncertain in the chemotaxis process, it easily falls into a local optimum. We propose a hybrid algorithm based on simulated annealing (SA) and BFO for mining imbalanced data. The key idea is to exploit the advantages of both SA and the BFO algorithm. In the proposed algorithm, SA finds the optimal solution by employing a jump process, so as to solve the uncertainty of the reversal direction in the chemotaxis process of BFO and avoid falling into a local optimum. SA is used to improve the chemotaxis process of BFO, and then the swarming process, reproduction process, and elimination–dispersal process of BFO are implemented. Four imbalanced datasets are used to test the performance of the proposed hybrid algorithm. In each imbalanced dataset used for testing, there is a certain correlation between the variables, making the dataset multivariate. Through the proposed algorithm, these four multivariate imbalanced datasets are effectively classified, and its performance compared with that of other algorithms. Experimental results show that for the different multivariate imbalanced datasets, the proposed algorithm is better than the original BFO algorithm in terms of various performance indicators. By combining the proposed algorithm with sensor-related technology, in the future, medical multivariate data and security monitoring system data obtained by sensors can be analyzed to improve the classification accuracy of multivariate data.

## 1. Introduction

Classification is important in data mining. Classification refers to the establishment of a data classification model based on the known data and its class attributes. The classification model classifies the data according to given classes in a database to predict new data. Most of the data used for disease diagnosis, face recognition, text classification, and financial risk prediction are imbalanced. When a traditional algorithm is used to solve such problems, the result of classification tends to the majority classification, which leads to the minority classification not being correctly recognized. However, in many practical applications, a few samples are more

[*]Corresponding author: e-mail: lqy@fzfu.edu.cn
[**]Corresponding author: e-mail: thmeen@nfu.edu.tw

https://doi.org/10.18494/SAM.2021.3167

ISSN 0914-4935 © MYU K.K.
https://myukk.org/

valuable than most other samples.(1,2) Thus, classification of imbalanced data is an important research topic in machine learning and data mining as the accuracy of algorithms depends on how correctly the data are classified. Data mining for imbalanced data can be performed using a decision tree (DT), artificial neural network (ANN), genetic algorithm (GA), and support vector machine (SVM).(3–6) Some methods such as the cost-sensitive classifier and snowball methods have been proposed to process imbalanced data.(7) The cost-sensitive classifier method focuses on minimizing misclassification costs as well as other types of cost with overspecific rules. The snowball method uses an ANN to learn the rules from the instances of minority classes, and then those of majority classes are added gradually when the ANN works dynamically. Unfortunately, the snowball method is only effective on some particular ANNs.(8)

The bacterial foraging optimization (BFO) algorithm is a heuristic swarm intelligence optimization algorithm proposed by Professor Passino of Ohio State University in 2002.(9) Its theory is based on the foraging behavior of *E. coli*. Based on chemotaxis, swarming, reproduction, and elimination–dispersal processes, the BFO has satisfactory performance in solving optimization problems.(10,11) However, during the chemotaxis process, the BFO depends on random search directions, which may delay reaching the universal solution. Recently, the combination of BFO with other algorithms to solve optimization problems has been proposed. Compared with a single method, a hybrid method involving BFO and other algorithms yields better results from various systems, resulting in improved optimization performance. In the simulated annealing (SA) algorithm, the Metropolis acceptance criterion is adopted.(12) The basic idea of SA comes from physical annealing. Starting from a high temperature, a constant decrease in the temperature leads to a random search for the global optimal solution of an objective function. That is, under the premise of obtaining the local optimal solution, the global optimal solution can be obtained.

In this paper, a hybrid algorithm involving SA and BFO for mining imbalanced data is proposed to solve the problem of the local optimum in the chemotaxis process of the BFO algorithm. In the proposed algorithm, SA finds the optimal solution of the location by employing a jump process to solve the uncertainty of the reversal direction in the chemotaxis process of BFO and avoid falling into a local optimum. After SA finds the optimal location, it then performs the BFO processes of swarming, reproduction, and elimination–disposal to improve the classification accuracy. The purpose of this study is to improve the classification accuracy of imbalanced data through the proposed hybrid algorithm of SA and BFO and to solve the problem that the original BFO easily falls into a local optimum. Four imbalanced datasets are used to test the performance of the proposed hybrid algorithm. In each imbalanced dataset used for testing, there is a certain correlation between the variables, making the dataset multivariate. Through the proposed algorithm, these four multivariate imbalanced datasets are effectively classified, and its performance compared with that of other algorithms. By combining the proposed algorithm with sensor-related technology, in the future, medical multivariate data and security monitoring system data obtained by sensors can be analyzed to improve the classification accuracy of multivariate data.

In Sect. 2, we first briefly review BFO and SA. The proposed algorithm is presented in Sect. 3. Simulation results are analyzed and discussed in Sect. 4. Finally, Sect. 5 concludes the paper.

## 2. BFO and SA

### 2.1 BFO

The BFO algorithm is optimized for random searches. Its mathematical model has four main basic steps: chemotaxis, swarming, reproduction, and elimination–dispersal.(13) The foraging behavior of bacteria is mainly based on these four operations. In the chemotaxis process, *E. coli* has two basic movements in foraging: swimming and tumbling. In general, bacteria roll more often in areas with poor environmental conditions and swim more often in better environments. When $Q(j,k,l)=\{\theta^i(j,k,l)\,|\,i=1,2,...,S\}$ indicates the *i*th bacterium in a population of *S* bacteria in the *j*th chemotaxis process, kth reproduction process, and *l*th elimination–dispersal process, *H*(*i*, *j, k, l*) indicates the cost at location *θ*(*j, k, l*) of the *i*th bacterium, and $N_c$ is the length of the bacteria in one direction of the chemotactic operation, the mathematical expression for the chemotaxis operation of the *i*th bacterium in the BFO model is defined as

$$\theta^i(j+1,k,l)=\theta^i(j,k,l)+\alpha(i)\frac{\beta(i)}{\sqrt{\beta^T(i)\beta(i)}}, \tag{1}$$

where *α*(*i*) is the length of the chemotaxis step of forward swimming and *β*(*i*) represents the unit vector in the random direction after tumbling.

The swarming process uses the concept that bacteria search for food through mutual attraction and repulsion. As a result of attraction, bacteria quickly gather together to search for food around the target (optimal) food, forming a population, while in the case of repulsion, each bacterium finds food independently. This is called the swarming operation in the BFO algorithm. In the BFO model, the mathematical expression for the swarming behavior is

$$\begin{aligned} H_{kk}(\theta,Q(j,k,l)) &= \sum_{i=1}^{s} H_{kk}^{i}(\theta,\theta^i(j,k,l)) \\ &= \sum_{i=1}^{s}\left[-d_{attract}\exp(-w_{attract}\sum_{m=1}^{n}(\theta_m-\theta_m^i)^2\right] \\ &\quad +\sum_{i=1}^{s}\left[-h_{repellent}\exp(-w_{repellent}\sum_{m=1}^{n}(\theta_m-\theta_m^i)^2\right] \end{aligned} \tag{2}$$

where $H_{kk}(\theta, Q(j, k, l))$ is the penalty added to the actual cost function, *S* is the number of bacteria, $\theta_m$ is the location of the fittest bacterium, $d_{attract}$ is the depth of attraction, $w_{attract}$ is the width of attraction, $h_{repellent}$ is the height of repulsion, and $w_{repellent}$ is the width of repulsion. In the swarming process, the number of biologically motivated choices is expressed as $N_s$. The formula used to calculate the fitness value of the swarming operation can be expressed as

$$H_{swarm}(i,j,k,l)=H(i,j,k,l)+H_{kk}(\theta,Q(j,k,l))\,. \tag{3}$$

The rule of biological evolution in nature is survival of the fittest. In the reproduction process of BFO, bacteria after foraging are sorted by their energy value using the cost function $H$. $S/2$ of the bacteria with the smallest energy values are eliminated. The remaining bacteria ($S/2$) are then reproduced by replication. The newly replicated bacteria have the same foraging ability as the original bacteria. The reproduction operation maintains the invariance of the population size. After $N_{re}$ reproduction steps, elimination–dispersal occurs according to a certain probability $P_{ed}$, where $N_{ed}$ is the number of steps of elimination–dispersal. An individual undergoing the elimination–dispersal process dies and a new individual is randomly generated at any location in the solution space. These new bacteria have a random characteristic and may have a different foraging ability from the original bacteria. This random characteristic results in a new population with a jump in the local optimal value, making it closer to the global optimal solution. The main steps of the BFO algorithm are as follows.

Step 1: Initialization parameters are obtained: population size $S$, number of times of chemotactic behavior $N_c$, maximum number of steps forward in chemotactic operation $N_s$, number of reproduction operations $N_{re}$, number of elimination–dispersal operations $N_{ed}$, elimination–dispersal probability $P_{ed}$, chemotaxis step $\alpha(i)$.

Step 2: Elimination–dispersal loop operation.

Step 3: Reproduction loop operation.

Step 4: Chemotaxis loop operation.

(1) The $i$th bacterium undergoes a chemotaxis step.

(2) Calculate $H(i, j, k, l)$ and store the optimal value $H_{best}$.

(3) The vector $\beta(i)$ is randomly generated, and the bacteria move according to step $\alpha(i)$; at this time, $\theta^i(j+1,k,l) = \theta^i(j,k,l) + \alpha(i)(\beta(i)\big/\sqrt{\beta^T(i)\beta(i)}\,)$.

(4) Calculate the fitness value $H(i, j + 1, k, l)$ according to the information of bacteria $\theta^i(j + 1, k, l)$.

(5) Rotation judgment condition, $m$ represents the counter of consecutive swimming. If $m < S_s$, $m = m + 1'$; if $H(i, j + 1, k, l) > H_{best}$, then $H_{best} = H(i, j + 1, k, l)$. This calculates a new $H(i, j + 1, k, l)$ according to the information of bacteria $\theta^i(j + 1, k, l)$, until $m$ reaches $N_s$.

(6) Repeat the process for the next bacteria.

Step 5: If $j < N_c$, return to step 4 for the bacterial chemotaxis operation.

Step 6: Reproduction: remove $S/2$ bacteria with the lowest energy values and replicate the remaining $S/2$ bacteria.

Step 7: If $k < N_{re}$, return to step 3.

Step 8: Elimination–dispersal: when certain conditions are met, bacteria find food again with probability $P_{ed}$. If $l < N_{ed}$, return to step 2; otherwise, end the optimization.

Step 9: Has the maximum number of BFO iterations been reached? If so, the result $H(i, j, k, l)$ is output.

## 2.2 SA

The SA algorithm is heuristic and simulates the physical process of cooling of a classical particle system in thermodynamics. Kirkpatrick *et al.* first proposed the SA algorithm in 1983.(14)

When the temperature $T$ of an isolated particle system decreases very slowly, it is considered that the system is in thermodynamic equilibrium, and the energy of the system is the lowest. In the SA algorithm, the Metropolis acceptance criterion is used to determine all the values of the control temperature parameter $T$. That is, the iterative process of "generating a new solution-making judgment by accepting or discarding" is repeated, and finally the system is found at temperature $T$ under the equilibrium with the optimal solution obtained. The SA algorithm flow is as follows.

(1) Determine the initial temperature $T_0$, the final temperature $T_f$, and the starting point $x_0$, and get the function value $f(x_0)$, the Metropolis number of the iteration $M_{iter}$, and the temperature cooling rate $\lambda$, $0 < \lambda < 1$.

(2) The perturbation $\Delta x$ is generated randomly and the new point $x' = x + \Delta x$ is obtained. The function value $f(x')$ and the difference $\Delta f = f(x) - f(x')$ of the new point are obtained.

(3) If $\Delta f \leq 0$, the new point is accepted as the starting point for the next time.

(4) If $\Delta f > 0$, the acceptance probability of the new point is calculated as $P = e^{-\Delta f/T}$. The pseudorandom number $r_{rand} \in [0, 1]$ is generated, which is uniformly distributed in the interval [0, 1]. If $P > r_{rand}$, a new point is used as the starting point for the next operation. Otherwise, the original point is still used as the starting point for the next operation.

(5) The temperature is gradually reduced to $T \leftarrow \lambda T$, where $0 < \lambda < 1$, and the above process is repeated until the specified end condition is reached.

## 3. Proposed Algorithm

The purpose of this study is to improve the classification accuracy of mining imbalanced data by using an effective algorithm, that is, a hybrid algorithm based on SA and BFO. The algorithm solves the problem of falling into a local optimum of the original BFO. We propose to insert SA into the chemotaxis process of BFO, use the characteristics of the SA probability to get rid of the local optimum, and then improve the classification accuracy of the original BFO. This is the main innovation of this paper. Four datasets were used for testing the performance of the proposed hybrid algorithm: an *E. coli* dataset, a zoo dataset, a spam email dataset, and a Pima Indian diabetes dataset in the University of California Irvine (UCI) repository.(15) The *E. coli* dataset had a total of 334 instances in eight features with an imbalance ratio of the data of about 1:15.8 (Table 1). The zoo dataset had 101 instances in 17 features (Table 2) with an imbalance ratio of 1:25. The spam email dataset had 4601 e-mails in 58 features with an imbalance ratio of 1:1.54 (Table 3). The Pima Indian diabetes dataset had 768 instances in nine features with an imbalance ratio of 1:2.34 (Table 4). The flow chart of the hybrid algorithm is shown in Fig. 1. With the set of used parameters, the borderline synthetic minority oversampling technique (borderline-SMOTE) and the Tomek link were used to preprocess data. Then, the SA algorithm was used to improve the chemotaxis operation in BFO to classify the imbalanced data and solve the shortcoming of the BFO algorithm of falling into a local optimum.

The basic idea of the hybrid algorithm with borderline-SMOTE is to determine borderline minority instances, apply a SMOTE algorithm to generate synthetic instances to oversample the minority class, and finally balance the datasets.(18) To create the hybrid algorithm, we used the Euclidean distance to find the $k$ nearest neighbors of the instance $x_i \in S_{min}$, where $S_{min}$ is the

Table 1
Eight features of *E. coli* dataset.

| Number | Feature name | Description | Domain |
|---|---|---|---|
| 1 | Sequence name | Accession number for the SWISS-PROT database. | String |
| 2 | Mcg | McGeoch's method for signal sequence recognition. | [0.0, 0.89] |
| 3 | Gvh | von Heijne's method for signal sequence recognition. | [0.16, 1.0] |
| 4 | Lip | von Heijne's signal peptidase II consensus sequence score. | [0.48, 1.0] |
| 5 | Chg | Presence of charge on N-terminus of predicted lipoproteins. | [0.5, 1.0] |
| 6 | Aac | Score of discriminant analysis of the amino acid content of outer membrane and periplasmic proteins. | [0.0, 0.88] |
| 7 | Alm1 | Score of the ALOM membrane spanning region prediction program. | [0.03, 1.0] |
| 8 | Alm2 | Score of the ALOM program after excluding the putative cleavable signal region from the sequence. | [0.0, 0.99] |

Table 2
Seventeen features of zoo dataset.[16]

| Number | Feature name | Data type | Number | Feature name | Data type |
|---|---|---|---|---|---|
| 1 | animal name | continuous | 10 | backbone | nominal |
| 2 | hair | nominal | 11 | breathes | nominal |
| 3 | feathers | continuous | 12 | venomous | nominal |
| 4 | eggs | nominal | 13 | fins | nominal |
| 5 | milk | nominal | 14 | legs | nominal |
| 6 | airborne | nominal | 15 | tail | nominal |
| 7 | aquatic | nominal | 16 | domestic | nominal |
| 8 | predator | nominal | 17 | catsize | nominal |
| 9 | toothed | nominal | | | |

Table 3
Fifty-eight features of spam email dataset.[17]

| Number | Meaning | Range | Maximum value |
|---|---|---|---|
| 1–48 | Frequency of occurrence of a particular word | [0, 100] | <100 |
| 49–54 | Frequency of occurrence of a particular character | [0, 100] | <100 |
| 55 | Travel length of capital letters | [1, …] | 1102.5 |
| 56 | Longest capital travel | [1, …] | 9989 |
| 57 | Total travel length of capital letters | [1, …] | 15841 |
| 58 | Spam ID (1 for spam) | [0, 1] | 1 |

Table 4
Nine features of Pima Indian diabetes dataset.

| Number | Feature name | Description | Data type |
|---|---|---|---|
| 1 | Pregnancies | Number of pregnancies | continuous |
| 2 | Glucose | Glucose content in plasma within 2 h | continuous |
| 3 | Blood Pressure | Diastolic pressure in mm Hg | continuous |
| 4 | Skin Thickness | Triceps thickness in mm | continuous |
| 5 | Insulin | Serum insulin content within 2 h | continuous |
| 6 | BMI | Body mass index | continuous |
| 7 | Diabetes Pedigree Function | Diabetes family function | continuous |
| 8 | Age | Age | continuous |
| 9 | Outcome | Diabetes or not (0: No, 1: Yes) | class |

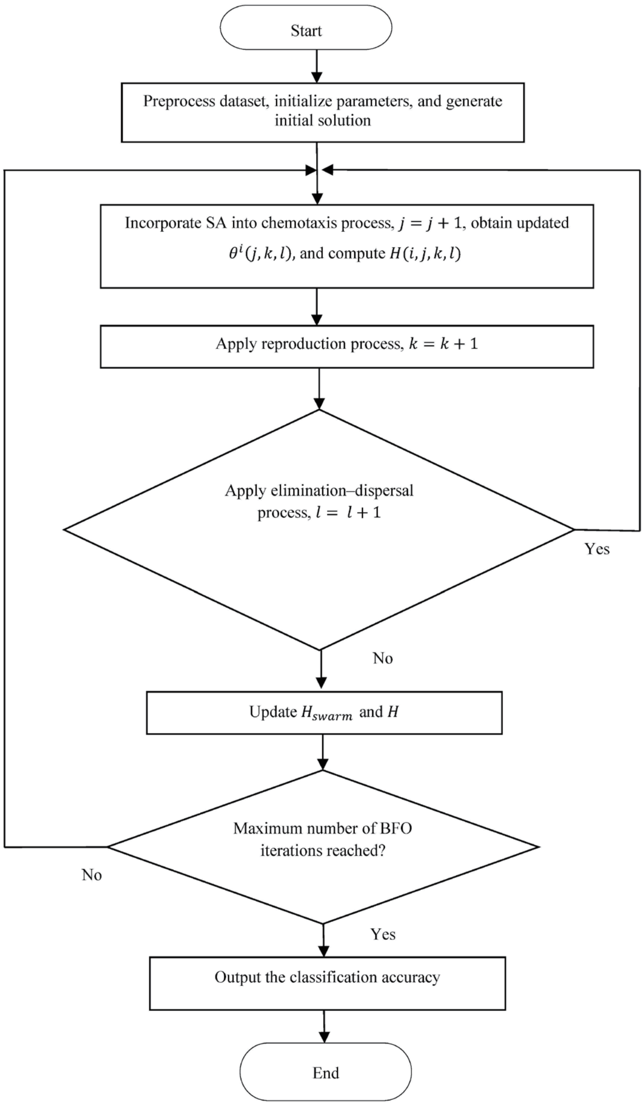

Start
Preprocess dataset, initialize parameters, and generate initial solution
Incorporate SA into chemotaxis process, $j = j + 1$, obtain updated $\theta^i(j,k,l)$, and compute $H(i,j,k,l)$
Apply reproduction process, $k = k + 1$
Apply elimination–dispersal process, $l = l + 1$
Yes
No
Update $H_{swarm}$ and $H$
Maximum number of BFO iterations reached?
No
Yes
Output the classification accuracy
End

Fig. 1. Flow diagram of the proposed algorithm.

minority classes, $x_i \in \{1, ..., n\}$, and $n$ is the number of minority instances. Here, $\overline{x}_i \in \{1, ..., k\}$ is one of the $k$ nearest neighbors. Next, we randomly selected $x_i$ from the $k$ nearest neighbors, and then generated a random number $P_{rand} \in [0, 1]$. Finally, we used Eq. (4) to generate a new instance and repeated the previous step until the number of instances balanced.

$$x_{new} = x_i + (\overline{x}_i - x_i) \times P_{rand} \tag{4}$$

The Tomek link reduces the impact of class overlap on classification performance.[19] The basic idea is as follows. Given a pair of instances $(x_i, x_j)$, where $x_i$ belongs to the majority class and $x_j$ to the minority class, the distance between the two points is defined as $d(x_i, x_j)$. If there is no instance $x_k$, $d(x_i, x_k) < d(x_i, x_j)$ or $d(x_j, x_k) < d(x_i, x_j)$, then $x_i$ and $x_j$ form a Tomek link pair.

In this study, the parameter $k$ used for SMOTE was set to $k = 3$. After preprocessing the data, $\theta^i$ was generated. Thereafter, the BFO of chemotaxis, swarming, reproduction, and elimination–dispersal is iterated. To solve the uncertainty of the reversal direction in the chemotaxis process of BFO and avoid the local optimum, SA was incorporated into BFO. As SA was added to the chemotaxis process of each individual bacterium, the cost of the bacterium was decided according to SA. The process of the hybrid algorithm was as follows.

Step 1: In the chemotaxis process, the SA algorithm begins with four parameters: $M_{iter}$, $T_0$, $T_f$, and $\lambda$. $M_{iter}$ denotes the maximum number of iterations, $T_0$ represents the initial temperature, $T_f$ is the final temperature at which the proposed algorithm stops as the temperature decreases, and $\lambda$ is the coefficient controlling the cooling rate. The current temperature $T$ is set to be the same as $T_0$. The solution is represented as features in the dataset followed with $\theta^i$ as shown in Fig. 2. An initial solution $\tau$ is generated according to the representation of the solution in Fig. 2. For each generation, the next solution $\eta$ is generated from $\tau$ by randomly swapping features and randomly generating $\theta^i$ in the current solution. Let $obj(\tau)$ denote the testing classification accuracy of $\tau$ and $\Delta$ denote the difference between $obj(\tau)$ and $\mathrm{obj}(\eta)$. That is, $\Delta = obj(\tau) - obj(\eta)$. If $\Delta \leq 0$, the probability of replacing $\tau$ with $\eta$ is 1, where $\tau$ is the current solution and $\eta$ is the next solution. Meanwhile, if $\Delta > 0$, the probability of replacing $\tau$ with $\eta$ is $e^{-\Delta/T}$. This is achieved by generating a random number $r_{rand} \in [0, 1]$ and replacing the solution $\tau$ with $\eta$ when $e^{-\Delta/T} > r_{rand}$. The hybrid algorithm is repeated until $T$ is lower than $T_f$. Thereafter, the SA obtains the best solution in the chemotaxis process.

Step 2: In the swarming process, Eq. (3) is applied to evaluate the cost $H_{kk}$.

Step 3: In the reproduction process, $S_r$ (= $S$/2) bacteria with the lowest costs H die and the other bacteria with the highest costs H split into two bacteria at the same location.

Step 4: In the elimination–dispersal process, the new $\theta^i$ obtained by SA is generated with the probability $P_{ed}$. The hybrid algorithm is repeated until the maximum number of BFO iterations is reached. Otherwise, the process goes back to Step 1.

Step 5: When the maximum number of BFO iterations is reached, the BFO stops. Finally, the classification accuracy result is reported.

In the multivariate imbalanced data, there is a certain correlation between the variables. The classification accuracy is used in multivariate data analysis, which indicates that the classification effect is good, but for the classification of imbalanced data, if the model returns its classification result as the major class, the classification accuracy can be high. Therefore, for the classification of imbalanced data evaluation indicators, this paper also utilizes precision, recall, f1 score, the receiver operating characteristic (ROC) curve, and the area under the curve (AUC). These performance indicators are calculated on the basis of the confusion matrix shown in Table 5, where $TP$ is the number of instances that are predicted to be positive and are actually positive, $TN$ is the number of instances that are predicted to be negative and are actually negative, $FP$

| | | | | | |
|---|---|---|---|---|---|
| Feature # 1 | Feature# 2 | Feature# 3 | … | Feature# $n$ | $\theta^i$ |

Fig. 2. Representation of the solution.

Table 5
Confusion matrix.

| Predicted | Actual | |
|---|---|---|
| | Actual positive | Active negative |
| Predicted positive | *TP* (true positive) | *FP* (false positive) |
| Predicted negative | *FN* (false negative) | *TN* (true negative) |

is the number of instances that are actually negative but are predicted to be positive, and *FN* is the number of instances that are actually positive but are predicted to be negative. In this paper, classification accuracy reflects the classifier's ability to judge the entire instance as positive or negative. The classification accuracy is given by

$$Classification\ accuracy = (TP + TN) / (TP + FN + FP + TN) \times 100\%. \tag{5}$$

The precision rate represents the proportion of correctly predicted positive instances out of all positive predictions determined by the classifier. The precision rate calculation formula is

$$Precision = \frac{TP}{(TP+FP)}. \tag{6}$$

The recall rate represents the proportion of positive instances predicted to be positive instances. The recall formula is

$$Recall = TP / (TP + FN). \tag{7}$$

The f1 score is a measure used in classification problems. It uses the harmonic average method to comprehensively consider the precision rate and the recall rate, and its maximum is 1 and minimum is 0. The f1 score calculation formula is

$$f1score = \frac{2\times Precision\times Recall}{Precision + Recall} \tag{8}$$

In order to verify the performance of the model, the evaluation criterion used in this experiment is AUC. The AUC value is expressed as the area under the ROC curve. The larger the AUC value, the more effective the model. The equation for AUC is

$$AUC = \frac{\sum_{i\in positiveclass} Rank_i - \frac{M\times(M+1)}{2}}{M\times N}, \tag{9}$$

where $\sum_{i\in positiveclass} Rank_i$ represents the sum of the sequence numbers for positive instances, $Rank_i$ is the sequence number of the *i*th instance, *M* denotes the number of positive instances, and *N* denotes the number of negative instances.

## 4. Results and Discussion

### 4.1 Setting of experimental parameters

The simulations required the parameters for the algorithm. The parameters of BFO used in this study were $S = 50$, $N_c = 100$, $N_s = 4$, $N_{re} = 4$, $N_{ed} = 2$, $P_{ed} = 0.25$, $d_{attract} = 0.05$, $h_{repellent} = 0.05$, $w_{attract} = 0.05$, $w_{repellent} = 0.05$, and $\alpha(i) = 0.1$, $i = 1, 2,\ldots$, S. The number of BFO iterations was 800 ($N_c \times N_{re} \times N_{ed} = 100 \times 4 \times 2 = 800$). SA was used to solve the uncertainty of the reversal direction during the chemotaxis of BFO and avoid falling into a local optimum. As the parameters of SA, the maximum number of iterations was $M_{iter} = 5000$, the initial temperature was $T_0 = 100$, the final temperature was $T_f = 0.01$, and the cooling rate was $\lambda = 0.95$. The evaluation model of the test dataset was verified by 10-fold cross-validation, and 90% of the data was used as training data and the rest used as testing data.

The main key to algorithm performance and efficiency is the parameter setting of the algorithm. As BFO can have many parameters, the optimal parameters are required to optimize the performance of the algorithm. For this, the analysis of parameters and the convergence and computational complexity are as follows.

(1) The population size $S$ affects the performance of the BFO. When the population is small and the calculation speed of the BFO is high, the diversity of the population is reduced, which affects the optimization performance of the algorithm. The higher the value, the better the algorithm avoids falling into a local optimal value. However, when the population is too large, the number of calculations of the algorithm increases and the convergence speed of the algorithm decreases.

(2) The larger the value of $N_c$, the number of executions of the chemotaxis operation, the more detailed the search of the algorithm. However, the complexity of the algorithm also increases. On the contrary, the smaller the $N_c$, the more easily the algorithm falls into a local optimal value. Then, the performance becomes more dependent on the operation.

(3) The larger the value of $N_{re}$, the number of iterations, the higher the algorithm's speed. Of course, a very large $N_{re}$ increases the complexity of the algorithm, while a very small $N_{re}$ makes the algorithm converge prematurely.

(4) $N_{ed}$ is the number of executions in the elimination–dispersal operation. If $N_{re}$ is too small, the algorithm does not randomly search the area in the elimination–dispersal operation. On the contrary, the larger the $N_{ed}$, the larger the area that the algorithm searches and the greater the diversity of the solution. Then, the algorithm is prevented from falling into precocity and the complexity of the algorithm also increases. Choosing an appropriate value for the elimination–dispersal probability $P_{ed}$ helps the algorithm jump out of a local optimal value, but if the value of $P_{ed}$ is too large, BFO becomes a random search algorithm.

The advantage of a heuristic search algorithm is that one set of solutions is obtained in one run, reducing the time and computational cost required to find the ideal solution and achieve good results. In this study, the SA algorithm solved the problem of the local optimization of the original BFO by combining it with the BFO chemotaxis operation. This improved the original BFO chemotaxis operation.

### 4.2 Comparison of experimental results with other algorithms

We tested the following algorithms for comparison with the hybrid algorithm: SVM, DT, k-nearest neighbor (KNN), back-propagation network (BPN), and BFO. SVM easily performs nonlinear classification by replacing the kernel. For the radial basis kernel function, the parameters of SVM used in this study were a penalty of 1 and a gamma of 0.1. DT is a decision support tool for graphics or decision models such as trees. The parameters of DT used in this study were a minimum case of 2 and a pruning confidence factor of 0.1. KNN is a simple machine learning method that classifies according to the distance between different feature values. The KNN parameter used in this study was $k = 3$ and the Euclidean distance was used. BPN is a learning method used in many neural networks, and its network behavior is based on the training data of input/output patterns. It is suitable for applications in diagnosis, prediction, classification, and other problems. In this paper, BPN used the sigmoid function in its hidden layer, namely, $f(x) = 1/(1 + e^{-x})$, the hidden layer node was set to 15, and the output layer used the linear function $f(x) = x$. The learning rate was 0.05 and the maximum number of iterations was 25000. The BFO algorithm is described in Sect. 2.1.

(1) From Table 6, the average classification accuracy in the proposed algorithm for the *E. coli* dataset is 97.61%. The average classification accuracies for the zoo, spam email, and Pima Indian diabetes datasets in the proposed algorithm are 99.55, 96.32, and 97.66%, respectively. It can be seen from Table 6 that the proposed algorithm in this paper has a higher classification accuracy than the other algorithms for the four datasets. This is because the performance of the classification for these tested datasets is found on the basis of heuristic information. In fact, the proposed approach also has a similar performance, so it performs well in terms of classification accuracy.

(2) The classification accuracies of the proposed algorithm of 97.61, 99.55, 96.32, and 97.66% for the *E. coli*, zoo, spam email, and Pima Indian diabetes datasets are better than those for the original BFO method of 90.12, 94.36, 95.25, and 93.64%, respectively. This is because the proposed algorithm adds SA to improve the chemotaxis process, that is, SA is added to the chemotaxis process of each individual bacterium. Owing to the ability of probabilistic jumping, the proposed algorithm overcomes the problem that the original BFO easily falls into a local optimum during the chemotaxis process, and then performs the swarm process, reproduction process, and elimination–dispersal process of BFO, so it has better classification accuracy.

(3) In 2017, Yang *et al.* proposed an intelligent algorithm based on BFO and a robust fuzzy algorithm (RFA) to analyze asthma data.(20) In the RFA-BFO algorithm, the classification accuracy of the UCI zoo multivariate imbalanced data used for testing was 99.5%. In

Table 6
Classification accuracy of different algorithms.

| Datasets | Algorithms | | | | | |
|---|---|---|---|---|---|---|
| | SVM | DT | KNN | BPN | BFO | Proposed algorithm |
| *E.coli* | 89.96% | 85.34% | 81.25% | 82.84% | 90.12% | 97.61% |
| zoo | 93.52% | 92.71% | 91.46% | 89.01% | 94.36% | 99.55% |
| spam email | 94.47% | 91.81% | 91.58% | 90.78% | 95.25% | 96.32% |
| Pima Indian diabetes | 91.25% | 89.33% | 90.17% | 89.92% | 93.64% | 97.66% |

this paper, the proposed algorithm also utilizes the same zoo dataset and has an accuracy of 99.55% (to two decimal places). RFA-BFO and the proposed algorithm have good classification results for imbalanced data. In the RFA-BFO algorithm, RFA with the property of robustness can reduce the influence of noise or outliers. It can establish a fuzzy model and analyze multivariate imbalanced data. However, the disadvantage of RFA based on experience is that the simple fuzzy processing of information may reduce the accuracy of data classification. On the other hand, in this study, SA is embedded in the chemotaxis of BFO to solve the problem of BFO easily falling into a local optimum. That is, SA is added to the chemotaxis process of each individual bacterium. In the chemotaxis process, SA is performed to obtain the updated location of the solution, and then the swarming, reproduction, and elimination–dispersal processes of BFO are performed. In the elimination–dispersal process, a new location of the solution is generated by SA according to the probability $P_{ed}$. Finally, when the criterion is satisfied, the classification accuracy results are output. Our purpose is to obtain an efficient search algorithm for multivariate imbalanced data, and we do not intend to discuss which of the two methods, RFA-BFO or the proposed algorithm, is better.

(4) To verify that the model does not return the classification results to the majority class to increase the classification accuracy, we calculate the precision, recall, and f1 score to obtain the results in Table 7. It can be seen from Table 7 that for the four datasets, although the imbalance rate is different, the proposed algorithm shows balance between the majority class

Table 7
Performance indicators of different datasets.

| Datasets | Algorithm | Precision | Recall | f1 score |
|---|---|---|---|---|
| *E. coli* | SVM | 0.8152 | 0.8351 | 0.8250 |
| | DT | 0.8060 | 0.8865 | 0.8443 |
| | KNN | 0.7855 | 0.815 | 0.8000 |
| | BPN | 0.8193 | 0.8054 | 0.8123 |
| | BFO | 0.8809 | 0.8875 | 0.8842 |
| | Proposed algorithm | 0.9714 | 0.9786 | 0.9750 |
| zoo | SVM | 0.9346 | 0.9256 | 0.9301 |
| | DT | 0.8768 | 0.8754 | 0.8761 |
| | KNN | 0.8213 | 0.8152 | 0.8182 |
| | BPN | 0.7351 | 0.7563 | 0.7455 |
| | BFO | 0.9313 | 0.9275 | 0.9294 |
| | Proposed algorithm | 0.9747 | 0.9815 | 0.9781 |
| spam email | SVM | 0.845 | 0.8716 | 0.8581 |
| | DT | 0.8667 | 0.8453 | 0.8559 |
| | KNN | 0.8255 | 0.8179 | 0.8217 |
| | BPN | 0.8377 | 0.8125 | 0.8249 |
| | BFO | 0.9412 | 0.9437 | 0.9424 |
| | Proposed algorithm | 0.9651 | 0.9847 | 0.9748 |
| Pima Indian diabetes | SVM | 0.8664 | 0.8575 | 0.8619 |
| | DT | 0.8321 | 0.8422 | 0.8371 |
| | KNN | 0.8572 | 0.8651 | 0.8611 |
| | BPN | 0.8653 | 0.8728 | 0.8690 |
| | BFO | 0.9035 | 0.9124 | 0.9079 |
| | Proposed algorithm | 0.9778 | 0.9524 | 0.9649 |

and the minority class in the classifier training. The f1 score of the proposed algorithm is also better than that of the other algorithms.

### 4.3 AUC evaluation

In signal detection theory, the value of AUC is between 0 and 1, and the larger the value, the better the model. The experimental results in Table 8 are the AUC values of the hybrid method with the four datasets. The AUC of the *E. coli* dataset was 0.974 as shown in Fig. 3. The AUC values of the other three datasets were 0.993 (zoo, Fig. 4), 0.997 (spam email, Fig. 5), and 0.963 (Pima Indian diabetes, Fig. 6). The AUC for each dataset exceeded 0.96, which proved the effectiveness of the hybrid algorithm in this study.

Table 8
AUCs of hybrid algorithm for four datasets.

| Datasets | AUC |
|---|---|
| *E. coli* | 0.974 |
| zoo | 0.993 |
| spam email | 0.997 |
| Pima Indian diabetes | 0.963 |

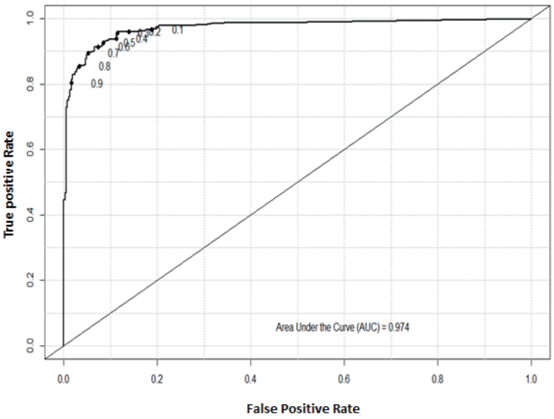


Fig. 3. ROC curve and AUC for *E. coli* dataset.

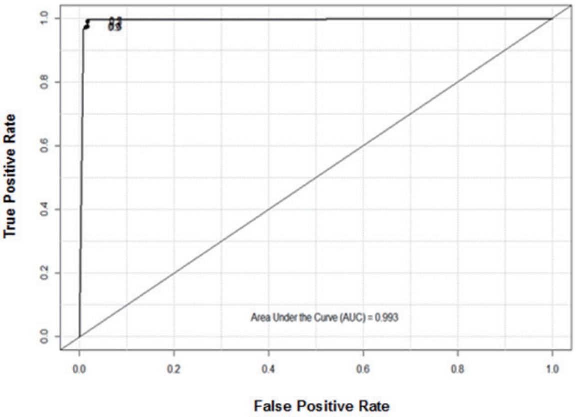


Fig. 4. ROC curve and AUC for zoo dataset.

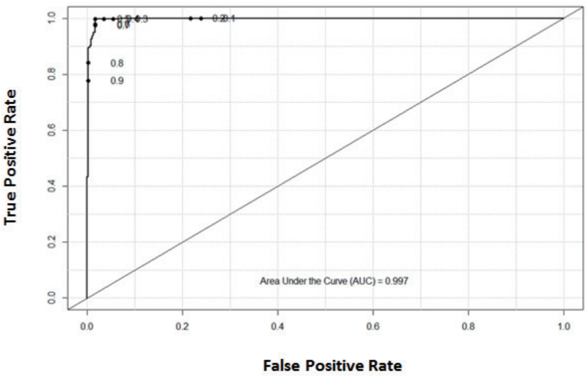


Fig. 5. ROC curve and AUC for spam email dataset.

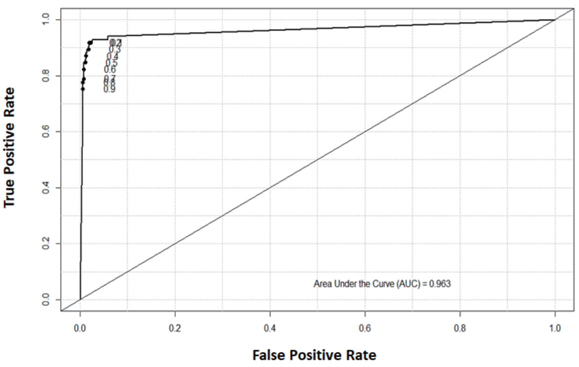


Fig. 6. ROC curve and AUC for Pima Indian diabetes dataset.

## 5. Conclusions

A hybrid algorithm based on SA and BFO for mining imbalanced data was proposed in this study. For the preprocessing of imbalanced data, we utilize borderline-SMOTE and the Tomek link. Because the SA algorithm has the characteristic of a jumping process based on probability, it can effectively avoid falling into a local optimum during the search process. The proposed hybrid algorithm adopted the BFO algorithm with the characteristics of SA to effectively solve the uncertainty of the chemotaxis process. Four multivariate imbalanced datasets (*E. coli*, zoo, spam email, Pima Indian diabetes) and other algorithms (SVM, DT, KNN, BPN, and BFO) were used for testing and comparison with the performance of the hybrid algorithm. The average classification accuracy of the hybrid algorithm for the *E. coli* dataset was 97.61%. The average classification accuracies of the *E. coli*, zoo, spam email, and Pima Indian diabetes datasets with the proposed algorithm were 97.61, 99.55, 96.32, and 97.66%, respectively. According to the experimental results, the hybrid algorithm achieved a significant improvement in various performance indicators compared with the other methods. The proposed algorithm can be applied to sensor-related experimental data to classify multivariate data obtained by sensors to improve prediction results.

To build on the hybrid algorithm based on SA and BFO for mining imbalanced data proposed in this study, we make the following suggestions:

(1) Improve the operation of the BFO algorithm by improving the chemotaxis, reproduction, and elimination–dispersion processes, and increase the classification accuracy of imbalanced data.

(2) Use the advantages of other large-scale optimization algorithms to propose more effective classification results.

## Acknowledgments

This research was supported by Fuzhou University of International Studies and Trade (grant numbers 2018KYTD-05, FWB19004, and 2019-SG-2) and by projects with grant numbers 2019020079010 and FWXXS19093. We also acknowledge the support of the Hundred Talents Program for Fujian Province to Introduce High-Level Talents from Taiwan.

## References

1 W. Hao and F. Liu: Symmetry **12** (2020) 1204. https://doi.org/10.3390/sym12081204

2 C. Y. Liu and P. Y. Hsieh: IEEE Trans. Knowl. Data Eng. **32** (2020) 1543. https://doi.org/10.1109/TKDE.2019.2905559

3 K. Liu, K. M. Chang, Y. J. Liu, and J. H. Chen: Symmetry **12** (2020) 1261. https://doi.org/10.3390/sym12081261

4 F. Wang, L. Yan, and J. Xiao: Sens. Mater. **31** (2019) 1335. https://doi.org/10.18494/SAM.2019.2288

5 J. Ruan, H. Jiang, X. Li, Y. Shi, F. T. S. Chan, and W. Z. Rao: IEEE Trans. Ind. Inf. **15** (2019) 6510. https://doi.org/10.0.4.85/TII.2019.2914158

6 J. Patalas-Maliszewska and D. Halikowski: Symmetry **11** (2019) 1151. https://doi.org/10.3390/sym11091151

7 L. Zhang and D. Zhang: IEEE Trans. Neural Networks Learn. Syst. **28** (2017) 3045. https://doi.org/10.1109/TNNLS.2016.2607757

8 Q. Li, P. Xu, Y. Y. Chan, Y. Wang, Z. Wang, H. Qu, and X. Ma: IEEE Trans. Visual Comput. Graphics **23** (2017) 211. https://doi.org/10.1109/TVCG.2016.2598415

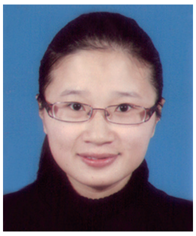
9 K. M. Passino: IEEE Control Syst. Mag. **22** (2002) 52. https://doi.org/10.1109/MCS.2002.1004010

10 G. Wang, J. Guo, Y. Chen, Y. Li, and Q. Xu: IEEE Access **7** (2019) 18840. https://doi.org/10.1109/ACCESS.2019.2897283
11 B. Niu, J. Liu, T. Wu, X. G. Chu, Z. G. Wang, and Y. M. Liu: IEEE/ACM Trans. Comput. Biol. Bioinf. **15** (2018) 1865. https://doi.org/10.1109/TCBB.2017.2742946
12 N. Metropolis, A. W. Rosenblush, M. N. Rosenblush, and A. H. Teller: J. Chem. Phys. **21** (1953) 1087. https://doi.org/10.1063/1.1699114
13 Q. Zhang, H. Chen, J. Luo, Y. Xu, C. Wu, and C. Li: IEEE Access **6** (2018) 64905. https://doi.org/10.1109/ACCESS.2018.2876996
14 B. S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi: Science **220** (1983) 671. https://doi.org/10.1126/science.220.4598.671
15 C. Blake, E. Keogh, and C. J. Merz: UCI Repository of Machine Learning Databases, Department of Information and Computer Science, University of California: Irvine, CA, USA (1998), https://doi.org/archive.ics.uci.edu/ml/datasets.php (accessed May 2019).
16 C. Y. Lee and Z. J. Lee: Appl. Soft Comput. **12** (2012) 2481. https://doi.org/10.1016/j.asoc.2012.03.051
17 F. L. Ye, C. Y. Lee, Z. J. Lee, J. Q. Huang, and J. F. Tu: Symmetry **12** (2020) 229. https://doi.org/10.3390/sym12020229
18 S. F. Abdoh, M. A. Rizka, and F. A. Maghraby: IEEE Access **6** (2018) 59475. https://doi.org/10.1109/ACCESS.2018.2874063
19 Z. Wang, C. Wu, K. Zheng, X. Niu, and X. Wang: IEEE Access **7** (2019) 129678. https://doi.org/10.1109/ACCESS.2019.2940061
20 M. R. Yang, Z. J. Lee, C. Y. Lee, B. Y. Peng, and H. Huan: Int. J. Fuzzy Syst. **19** (2017) 1181. https://doi.org/10.1007/s40815-017-0294-1

## About the Authors

**Chou-Yuan Lee** received his B.S. degree in automatic control engineering and his M.S. degree in automatic control engineering from Feng-Chia University (FCU), Taichung, Taiwan, in 1989 and 1991, respectively, and his Ph.D. degree in electrical engineering from National Taiwan University of Science and Technology (NTUST), Taipei, Taiwan, in 2008. He is a full professor with the Department of Big Data Management and Application, School of Technology, Fuzhou University of International Studies and Trade, China. His research interests include big data, data mining, and computational intelligence. (lqy@fzfu.edu.cn)

**Zne-Jung Lee** received his B.S. degree in electronic engineering and his M.S. degree in automatic control engineering from Feng-Chia University (FCU), Taichung, Taiwan, in 1986 and 1988, respectively, and his Ph.D. degree in electrical engineering from National Taiwan University of Science and Technology (NTUST), Taipei, Taiwan, in 2002. He is a full professor with the Department of Information Management and Information, School of Technology, Fuzhou University of International Studies and Trade, China. His research interests include data mining and computational intelligence. (lrz@fzfu.edu.cn)



**Jian-Qiong Huang** graduated from Fuzhou University in 2012 with an M.S. degree in computer technology. She is now an associate professor with the Department of Big Data Management and Application, School of Technology, Fuzhou University of International Studies and Trade. Her current research interests include data mining and database technology. (hjq@fzfu.edu.cn)

**Fu-Lan Ye** graduated from Fuzhou University in 2009 with an M.S. degree in computer technology. She is now an associate professor with the Department of Information Management and Information, School of Technology, Fuzhou University of International Studies and Trade. Her main research direction is data mining. (yfl@fzfu.edu.cn)

**Jie Yao** graduated from Beijing University of Posts and Telecommunications in 2011 with an M.S. degree in electronic and communication engineering. She is now researching big data management and applications at Fuzhou University of International Studies and Trade. Her main research direction is artificial intelligence. (yj@fzfu.edu.cn)

**Zheng-Yuan Ning** is a professor with the Department of Information Management and Information, School of Technology, Fuzhou University of International Studies and Trade, Fujian, China. He is mainly engaged in intelligent computing, algorithmic analysis, big data, and cloud computing. (nzy@fzfu.edu.cn)

**Teen-Hang Meen** received his B.S. degree from the Department of Electrical Engineering, National Cheng Kung University (NCKU), Tainan, Taiwan in 1989, and his M.S. and Ph.D. degrees from the Institute of Electrical Engineering, National Sun Yat-Sen University (NSYSU), Kaohsiung, Taiwan, in 1991 and 1994, respectively. He was the chairman of the Department of Electronic Engineering of National Formosa University, Yunlin, Taiwan from 2005 to 2011. He received the Excellent Research Award given by National Formosa University in 2008 and 2014. Currently, he is a distinguished professor with the Department of Electronic Engineering, National Formosa University, Yunlin, Taiwan. He is also the president of the International Institute of Knowledge Innovation and Invention (IIKII) and the chair of the IEEE Tainan Section Sensors Council. In recent years, he has published more than 100 SCI and SSCI papers. (thmeen@nfu.edu.tw)