

# An Improved Bacterial-Foraging, Optimization-Based Machine Learning Framework for Predicting the Severity of Somatization Disorder

Xinen Lv <sup>1</sup>, Huiling Chen <sup>2,\*</sup>, Qian Zhang <sup>2</sup>, Xujie Li <sup>2</sup>, Hui Huang <sup>2</sup> and Gang Wang <sup>3</sup>

<sup>1</sup> Mental health education center, Wenzhou University, Wenzhou 325035, China; luxinen@163.com

<sup>2</sup> Department of Computer Science, Wenzhou University, Wenzhou 325035, China; qz\_zhangqian@163.com (Q.Z.), lixujie101@aliyun.com (X.L.), huanghui@wzu.edu.cn (H.H.)

<sup>3</sup> College of Computer Science and Technology, Jilin University, Changchun 130012, China; wanggang.jlu@gmail.com

\* Correspondence: chenhuiling.jlu@gmail.com

Received: 22 December 2017; Accepted: 30 January 2018; Published: 6 February 2018

**Abstract:** It is of great clinical significance to establish an accurate intelligent model to diagnose the somatization disorder of community correctional personnel. In this study, a novel machine learning framework is proposed to predict the severity of somatization disorder in community correction personnel. The core of this framework is to adopt the improved bacterial foraging optimization (IBFO) to optimize two key parameters (penalty coefficient and the kernel width) of a kernel extreme learning machine (KELM) and build an IBFO-based KELM (IBFO-KELM) for the diagnosis of somatization disorder patients. The main innovation point of the IBFO-KELM model is the introduction of opposition-based learning strategies in traditional bacteria foraging optimization, which increases the diversity of bacterial species, keeps a uniform distribution of individuals of initial population, and improves the convergence rate of the BFO optimization process as well as the probability of escaping from the local optimal solution. In order to verify the effectiveness of the method proposed in this study, a 10-fold cross-validation method based on data from a symptom self-assessment scale (SCL-90) is used to make comparison among IBFO-KELM, BFO-KELM (model based on the original bacterial foraging optimization model), GA-KELM (model based on genetic algorithm), PSO-KELM (model based on particle swarm optimization algorithm) and Grid-KELM (model based on grid search method). The experimental results show that the proposed IBFO-KELM prediction model has better performance than other methods in terms of classification accuracy, Matthews correlation coefficient (MCC), sensitivity and specificity. It can distinguish very well between severe somatization disorder and mild somatization and assist the psychological doctor with clinical diagnosis.

**Keywords:** kernel extreme learning machine; somatization disorder; parameter optimization; bacterial foraging optimization; opposition-based learning

## 1. Introduction

Since the middle of the 20th century, all countries in the world have been reforming the penal system to explore more humane, scientific and effective punishment methods to improve the psychology and behavior of criminals. Community correction systems are an entirely new way of punishment under such innovation background. According to statistics, there are about 700,000 community correction objects in our country. In order to understand the psychological and behavioral characteristics of community correction objects, a self-rating scale (SCL-90) [1,2] is generally adopted to evaluate their mental health level. On the scale, somatization disorder can be

reflected by 12 factors. Somatization factors mainly reflects subjective physical discomfort, including the main discomfort in cardiovascular, gastrointestinal, respiratory system as well as headaches, back pain, muscle pain and other physical performance of anxiety. According to the research, the application of artificial intelligence technology in this area has not yet been reported.

This paper proposed a new machine learning framework to predict the severity of somatization disorder for the first time. In order to improve the accuracy rate of assessment of the severity of the somatization disorder, this paper proposes a kernel extreme learning machine based on opposition-based BFO method (improved bacterial-foraging, optimization-based kernel extreme learning machine, IBFO-KELM). As a major machine learning model, a kernel extreme learning machine (KELM) develops on the basis of an extreme learning machine with the integration of the kernel concept [3]. Due to its unique advantages, KELM has been widely used in some classification tasks [4–19] and performed well especially in computer-assisted medical diagnosis [7–11]. However, KELM's performance is mainly influenced by the parameters in its models, and the research shows that the accuracy of its classification can be greatly improved by establishing appropriate model parameter settings. Therefore, the key parameters should be set to the appropriate values before they are applied to the actual problem, such as penalty factor and width of kernel function. Traditionally, these parameters are treated by means of grid search and gradient descent. However, these methods are easy to get into local optimal solutions. Recently, some bio-inspired meta heuristic search algorithms (such as genetic algorithms (GA) [15,20,21], particle swarm optimization algorithms (PSO) [18,22,23], the grey wolf optimization algorithms (GWO) [13]) have made it easier to find the global optimal solution, compared with the traditional method.

Based on *Escherichia coli*'s food consuming behavior in the human intestine, the bacteria optimization algorithm (BFO) was proposed by Passino in 2002 as a new kind of bionic algorithm [24] that imitates four kinds of intelligent behavior, such as the bacterial foraging trend, as well as cluster, copy, and disperse. This algorithm is another hot spot in the field of bio-heuristic computing due to its advantages, such as the parallel search of the swarm intelligence algorithm and that it can easily jump out of local minima. The BFO algorithm has been widely used in many fields [25–33] since it was put forward. However, in the process of optimization, BFO is extremely easy to get into the local optimum [34–36], and it is difficult to find the global optimal solution. Aiming at resolving this problem, this paper introduces opposition-based learning strategy [37] into BFO as an improved bacterial foraging optimization algorithm (IBFO) to improve the population diversity and the rate of convergence. The principle of opposition-based learning strategy [37] is to generate corresponding opposite solutions for each initial candidate solution. From these two kinds of solutions (candidate solutions and corresponding opposite solutions), the solutions with relatively better fitness are selected as members of the initial populations. This will help to improve the rate of convergence in the process of optimization. Through an opposition-based learning strategy, opposite bacteria populations are obtained to help increase the diversity of the population and distribute the initial population of individuals evenly as far as possible and improve the convergence rate of the optimization process. The IBFO algorithm is then used to solve the parameter optimization problem of KELM and obtain the optimal model (IBFO-KELM). Furthermore, this model will be used to predict the somatization severity of community correction personnel. As far as we know, this paper is the first to solve the parameter optimization problem of the KELM model. In the experiment, a 10-fold cross-validation method is used on data in a symptom self-assessment scale (SCL-90) to make detailed comparison between IBFO-KELM, BFO-KELM (model based on the primitive bacterial foraging optimization model), GA-KELM (model based on genetic algorithms), PSO-KELM (model based on particle swarm optimization algorithms) and Grid-KELM (model based on grid search). The experimental results show that the proposed IBFO-KELM prediction model has better performance than other methods in the classification accuracy, Mathews correlation coefficient (MCC), sensitivity and specificity.

The main contributions of this study are as follows:

- (a) First, in order to fully explore the potential of the KELM classifier, we introduce an opposition-based, learning-strategy-enhanced BFO to adaptively determine the two key parameters of

KELM, which aided the KELM classifier in more efficiently achieving the maximum classification performance.

- (b) The resulting model, IBFO-KELM, is applied to serve as a computer-aided decision-making tool for predicting the severity of somatization disorder.
- (c) The proposed IBFO-KELM method achieves superior results, and offers more stable and robust results when compared to the four other KELM models.

The remainder of this paper is structured as follows. In Section 2, background information regarding KELM, BFO and opposition-based learning are presented. The implementation of the proposed methodology is explained in Section 3. In Section 4, the experimental design is described in detail. The experimental results and a discussion of the proposed approach are presented in Section 5. Lastly, the conclusions and recommendations for future work are summarized in Section 6.

## 2. Background Information

### 2.1. Kernel Extreme Learning Machine (KELM)

In this section, a brief description of KELM is provided. KELM is an improved version of ELM, proposed by Huang et al. [3], which combines the kernel function into ELM, ensuring that the network has good generalization performance and greatly improves the forward nerve network learning speed and avoids many problems of gradient descent training methods represented by back propagation neural networks, such as their easily falling into local optimal, infinite iterations, etc. KELM not only has the multi-dominance of the ELM algorithm, but also combines the kernel function to map the linear inseparable pattern non-linearly to the high-dimensional feature space in order to achieve linear separability and further improve the accuracy.

Using  $L$  hidden nodes in the output layer, the output function of the single-hidden-layer feed-forward neural networks (SLFNs) can be expressed as:

$$f(x) = h(x)\beta \quad (1)$$

where  $\beta = [\beta_1, \beta_2, \dots, \beta_L]^T$  is the output weight vector between the hidden layer with  $L$  neurons and the output neurons,  $h(x) = [h_1(x), h_2(x), \dots, h_L(x)]$  is the output vector relative to the hidden layer of input  $x$ , which maps the data from the input space to the ELM feature space.

In the newly-developed KELM, the introduction of a positive coefficient into the learning system makes the ELM more stable. If it is non-singular, the coefficient  $C$  can be added to the diagonal of  $HH^T$  when calculating the output weight  $\beta$ . Thus, the corresponding output function of the regularized ELM can be expressed as follows:

$$f(x) = h(x)\beta = h(x)H^T \left( \frac{I}{C} + HH^T \right)^{-1} T \quad (2)$$

It has been shown that ELM with a kernel matrix can be defined as follows.

$$\Omega_{KELM} = HH^T : \Omega_{KELM i,j} = h(x_i) \cdot h(x_j) = K(x_i, x_j) \quad (3)$$

The output function can be written as:

$$f(x) = h(x)H^T \left( \frac{I}{C} + HH^T \right)^{-1} T = \begin{bmatrix} K(x, x_1) \\ \dots \\ K(x, x_N) \end{bmatrix}^T \left( \frac{I}{C} + \Omega_{KELM} \right)^{-1} T \quad (4)$$

In this case, we do not need to know the hidden layer feature map  $h(x)$  because we can replace it with the corresponding kernel function  $K(u, v)$ .

## 2.2. Bacterial Foraging Optimization (BFO)

The bacterial foraging algorithm (BFO) is a novel swarm intelligence algorithm proposed by Passino in 2002 [24], based on the competitive cooperative mechanism of *E. coli* in the process of searching for food in the human intestine. By simulating the foraging behavior of *E. coli*, the BFO mainly consists of four behaviors: chemotaxis, swarming, reproduction, and elimination–dispersal.

- (1) Chemotaxis: Chemotaxis operation is the core of the algorithm, which simulates the foraging behavior of *E. coli* moving and tumbling. In poorer areas, the bacteria tumble more frequently, while bacteria move in areas where food is more abundant. The chemotaxis operation of the  $i$ th bacterium can be represented as

$$\begin{aligned}\theta^i(j+1, k, l) &= \theta^i(j, k, l) + C(i) * dct_i \\ dct_i &= \frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}}\end{aligned}\quad (5)$$

where the  $\theta^i(j, k, l)$  represents the  $i$ th bacterium at the  $j$ th chemotactic,  $k$ th reproductive, and  $l$ th elimination–dispersal steps.  $C(i)$  is the trend step length of bacteria  $i$  in a random direction ( $dct_i$ ).  $\Delta$  is a random vector between  $-1$  and  $1$ .

- (2) Swarming: In the chemotactic of bacteria to the foraging process, in addition to searching for food in their own way, there is both gravitation and repulsion among the individual bacteria. Bacteria will generate attractive information to allow individual bacteria to travel to the center of the population, bringing them together; at the same time, individual bacteria are kept at a distance based on their respective repulsion information.
- (3) Reproduction: According to the natural mechanism of survival of the fittest, after some time, bacteria with weak ability to seek food will eventually be eliminated, and bacteria with strong feeding ability will breed offspring to maintain the size of the population. By simulating this phenomenon, a reproduction operation is proposed. In  $S$ -sized populations,  $S/2$  bacteria with poor fitness were eliminated and  $S/2$  individuals with higher fitness self-replicated after the bacteria performed the chemotaxis operator. After the execution of the reproduction operation, the offspring will inherit the fine characteristics of the parent completely, protect the good individuals, and greatly accelerate the speed towards the global optimal solution.
- (4) Elimination–Dispersal: In the process of bacterial foraging, do not rule out the occurrence of unexpected conditions leading to the death of bacteria or causing them to migrate to another new area. By simulating this phenomenon, an elimination–dispersal operation has been proposed. This operation occurs with a certain probability  $Ped$ . When the bacterial individual satisfies the probability  $Ped$ , then the individual of the bacterial dies and randomly generates a new individual anywhere in the solution space. This bacterium may be different from the original bacterial, which helps to jump out of the local optimal solution and promote the search for the global optimal solution.

## 2.3. Improved Bacterial Foraging Optimization (IBFO)

The IBFO strategy was constructed by combining opposition-based learning (OBL) with the original BFO algorithm. OBL was first proposed by Tizhoosh [37]. The OBL strategy generates corresponding opposite solutions for each initial candidate solution. From these two kinds of solutions (candidate solutions and corresponding opposite solutions), the solutions with relatively better fitness are selected as members of the initial populations. This will help to improve the convergence rate in the optimization process. The related mathematical concepts of the OBL strategy can be represented as follows.

Let  $x \in R$  be a real number defined on a certain interval:  $x \in [a, b]$ . The opposite number  $\tilde{x}$  is defined as follows:

$$x = a + b - x \quad (6)$$

Let  $P(x_1, x_2, \dots, x_n)$  be a point in a  $n$ -dimensional coordinate system with  $x_1, x_2, \dots, x_n \in R$  and  $x_i \in [a_i, b_i]$ . The opposite point  $P^*$  is completely defined by its coordinates  $x_1^*, x_2^*, \dots, x_n^*$  where

$$x_i^* = a_i + b_i - x_i \quad i=1, \dots, n \quad (7)$$

Let  $f(x)$  be the function in focus and  $g(\cdot)$  a proper evaluation function. If  $x \in [a, b]$  is an initial (random) guess and  $x^*$  is its opposite value, then in every iteration we calculate  $f(x)$  and  $f(x^*)$ . The learning continues with  $x$  if  $g(f(x)) \geq g(f(x^*))$ , otherwise with  $x^*$ .

The initial solutions of bacterial populations can be expressed as

$$x_i = x_{low} + rand(x_{up} - x_{low}) \quad i=1, \dots, S \quad (8)$$

where  $S$  is the swarm size of the population.

The corresponding opposite solutions  $x_i^*$  of bacterial populations based on opposition-based learning can be expressed as

$$x_i^* = x_{up} + x_{low} - x_i \quad (9)$$

### 3. Proposed IBFO-KELM Model

In this study, a novel evolutionary KELM model was developed using the IBFO strategy, as shown in Figure 1. The two key parameters of the proposed KELM model were automatically tuned by using the IBFO strategy to simulate the foraging behaviors of the BFO model and its interactions with the surrounding environment. The proposed model was comprised of two main procedures, including inner parameter optimization and outer performance evaluation. During the inner parameter optimization procedure, the penalty parameters  $C$  and kernel bandwidth  $\gamma$  of the KELM were determined dynamically using the IBFO technique via a five-fold cross validation (CV) analysis. Then, the obtained optimal parameter pair  $(C, \gamma)$  was inputted into the KELM prediction model in order to perform the classification task in the outer loop using a 10-fold CV strategy. The classification error rate was used as the fitness function.

$$fitness = \left( \sum_{i=1}^K testError_i \right) / k \quad (10)$$

where  $testError_i$  represents the average test error rates achieved by the KELM classifier via the five-fold CV during the inner parameter optimization procedure.

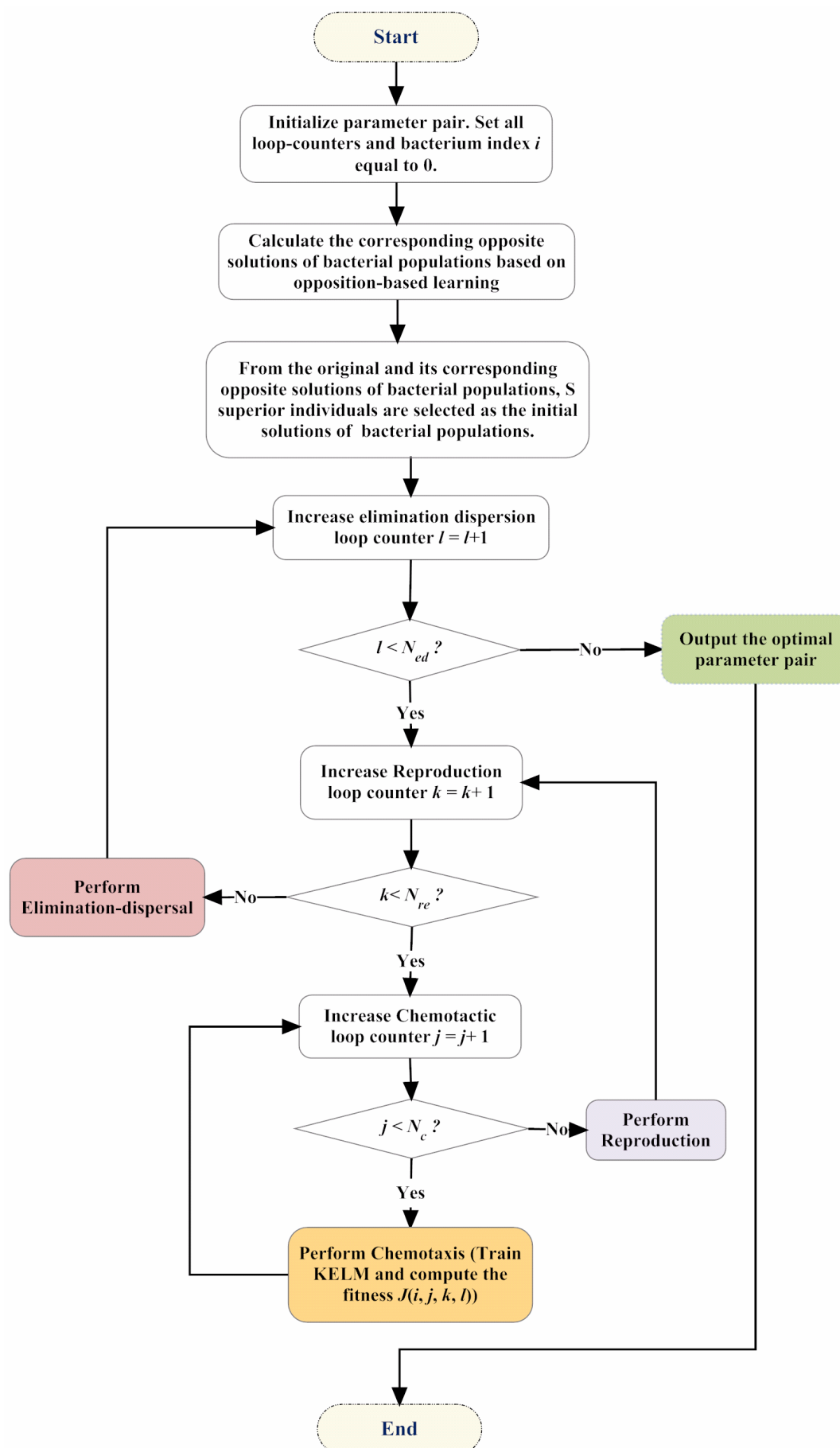


Figure 1. Flowchart of the improved bacterial foraging optimization strategy

The pseudo-code of the IBFO strategy is described in detail as follows in Algorithm 1:

---

**Algorithm 1.** Pseudo-code of the improved bacterial foraging optimization (IBFO) strategy.

---

**Begin**

Initialize dimension  $p$ , population  $S$ , chemotactic steps  $N_c$ , swimming length  $N_s$ , reproduction steps  $N_{re}$ , elimination-dispersal steps  $N_{ed}$ , elimination-dispersal probability  $P_{ed}$ , step size  $C(i)$ .

Calculate the corresponding opposite solutions of bacterial populations based on opposition-based learning. From the original and its corresponding opposite solutions of bacterial populations,  $S$  superior individuals are selected as the initial solutions of bacterial populations.

**for**  $ell = 1:N_{ed}$

**for**  $K = 1:N_{re}$

**for**  $j = 1:N_c$

$Intertime = Intertime + 1;$

**for**  $i = 1:s$

$J(i,j,K,ell) = fobj(P(:,i,j,K,ell));$

$J_{last} = J(i,j,K,ell);$

**Tumble according to Equation (5)**

$m = 0;$

**while**  $m < N_s$

$m = m + 1;$

**if**  $J(i,j+1,K,ell) < J_{last}$

$J_{last} = J(i,j+1,K,ell);$

**Tumble according to Equation (5)**

**else**

$m = N_s;$

**End**

**End**

**End**

**End**

**/\*Reproduction\*/**

$J_{health} = \text{sum}(J(:, :, K, ell), 2);$

$[J_{health}, \text{sortind}] = \text{sort}(J_{health});$

$P(:, :, 1, K+1, ell) = P(:, \text{sortind}, N_c + 1, K, ell);$

**for**  $i = 1:S_r$

$P(:, i + S_r, 1, K+1, ell) = P(:, i, 1, K+1, ell);$

**End**

**End**

**/\*Elimination-Dispersal\*/**

**for**  $m = 1:s$

**if**  $P_{ed} > \text{rand}$

**Reinitialize bacteria**  $m$

**End**

**End**

**End**

**End**

---

## 4. Experimental Design

### 4.1. Somatization Disorder Data Description

The data obtained from this research are all from the jurisdiction of Wenzhou Municipal Bureau of Justice. A total of 419 community correction objects were selected as the research objects. These research objects are mainly inmates with mild offences, little subjective viciousness, including inmates who were sentenced to public surveillance, have been suspended, with temporary sentence execution outside a prison and a ruling by parole personnel, etc. In this study, the symptom self-rating scale (SCL-90) was used to study psychological and physical symptoms, depression, hostility and psychosis in the recent week. Among them, somatization symptoms mainly reflect subjective bodily discomfort, with 12 attributes in all. The range of values for features F1–F12 is {1, 2, 3, 4, 5}, representing five different intensities of a given symptom. The range of values for the decision attribute F13 is {1, 2}, representing a severe state and the mild state, respectively. Table 1 gives a description of the 13 attributes.

**Table 1.** Description of the somatization disorder data.

Feature	Description
F1	Headache
F2	Dizzy or fainted
F3	Chest pain
F4	Low back pain
F5	Nausea or upset stomach
F6	Muscle soreness
F7	Having breathe difficulty
F8	A series of chills or fever
F9	Body tingling or prickling
F10	The throat is infarcted
F11	Feeling that part of the body is weak
F12	Feeling the weight of your hands or feet
F13	Somatization severity

### 4.2. Experimental Setup

The computational analysis was conducted on a Windows 7 operating system with an AMD Athlon 64 X2 Dual Core Processor 5000+ (2.6 GHz) (AMD company, Santa Clara, California) and 4GB of RAM. The IBFO-KELM, BFO-KELM, PSO-KELM, GA-KELM, and Grid-KELM models were implemented in the MATLAB platform. The technique used by Huang (available online: <http://www3.ntu.edu.sg/home/egbhuang>, 17 January 2018) was used for the KELM classification. The data was scaled into a range of [−1, 1] before each classification was conducted.

The search ranges of the penalty parameter  $C$  and kernel bandwidth  $\gamma$  in  $K(x, x_i) = \exp(-\gamma \|x - x_i\|^2)$  were defined as  $C \in \{2^{-5}, 2^{-3}, \dots, 2^{15}\}$  and  $\gamma \in \{2^{-15}, 2^{-13}, \dots, 2^5\}$ . A population swarm size of eight, chemotactic step number of five, swimming length of four, reproduction step number of five, elimination–dispersal event number of two, and elimination–dispersal probability of 0.25 were selected for BFO-KELM and IBFO-KELM. The chemotaxis step value was established through trial and error, as shown in the experimental results section. For PSO, the maximum velocity was set to about 60% of the dynamic range of the variable on each dimension for the continuous type of dimensions. The two acceleration coefficients  $c_1$  and  $c_2$  were set as 2.05, the inertia weight was set to one. In order to determine the validity and accuracy of the results, the  $k$ -fold CV [38] was used to evaluate the classification performance of the model. A nested stratified 10-fold CV, which has been widely used in previous research, was used for the purposes of this study [39]. The classification performance evaluation was conducted in the outer loop. Since a 10-fold CV was used in the outer loop, the classifiers were evaluated in one independent fold of data, and the other nine folds of data



were left for training. The parameter optimization process was performed in the inner loop. Since a five-fold CV was used in the inner loop, the IBFO-KELM searched for the optimal values of  $C$  and  $\gamma$  in the remaining nine folds of data. The nine folds of data were further split into one fold of data for the performance evaluation, and four folds of data were left for training. To evaluate the proposed method, commonly used evaluation criteria such as classification accuracy (ACC), sensitivity, specificity and Matthews correlation coefficients (MCC) were analyzed. .

## 5. Experimental Results and Discussion

### 5.1. Benchmark Function Validation

In order to test the performance of the proposed algorithm IBFO, nine multidimensional benchmark functions, as shown in Table 2, were used.

**Table 2.** Benchmark functions.

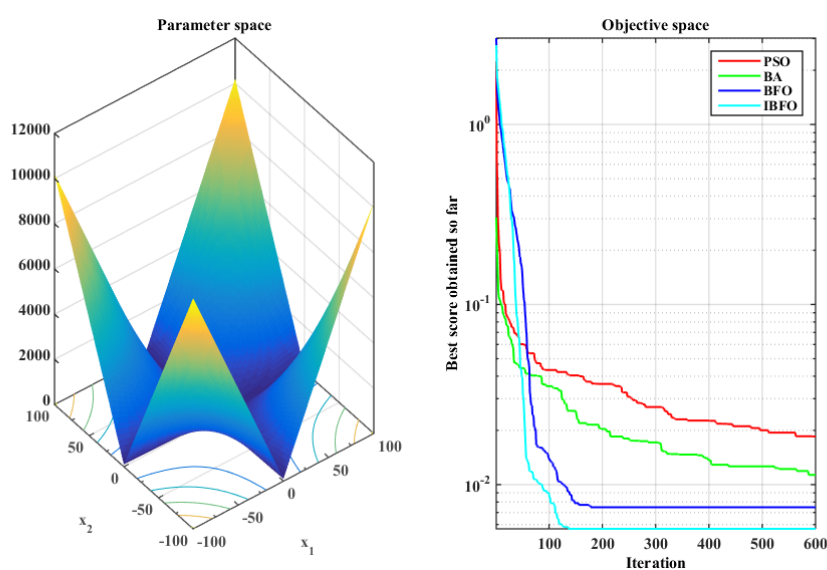
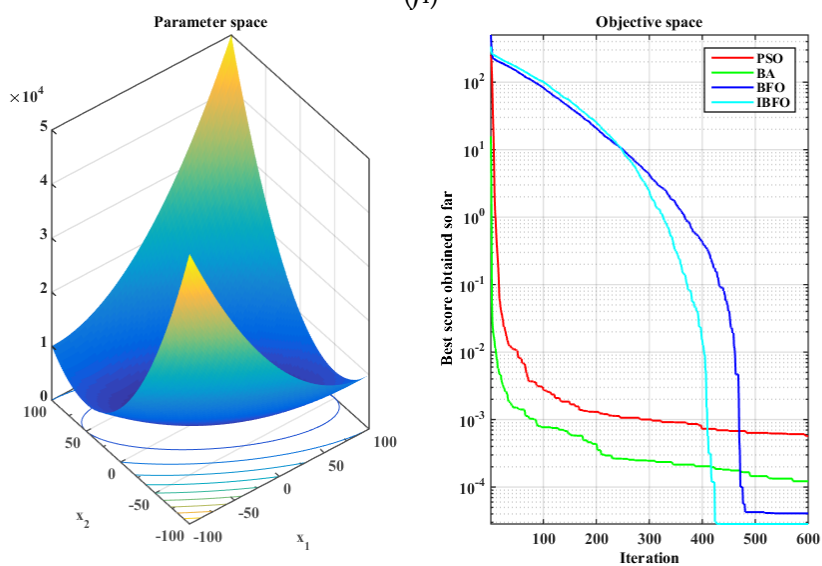
Function	Range	Minimum
$f_1(x) = \sum_{i=1}^n  x_i  + \prod_{i=1}^n  x_i $	$[-10, 10]$	0
$f_2(x) = \sum_{i=1}^n (\sum_{j=1}^i x_j)^2$	$[-100, 100]$	0
$f_3(x) = \max_i \{ x_i , 1 \leq i \leq n\}$	$[-100, 100]$	0
$f_4(x) = \sum_{i=1}^n ([x_i + 0.5])^2$	$[-100, 100]$	0
$f_5(x) = \sum_{i=1}^n ix_i^4 + \text{random}[0, 1]$	$[-1.28, 1.28]$	0
$f_6(x) = \sum_{i=1}^n -x_i \sin(\sqrt{ x_i })$	$[-500, 500]$	$-418.9829 \times 5$
$f_7(x) = (\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^2 (x_i - a_{ij})^6})^{-1}$	$[-65, 65]$	1
$f_8(x) = \sum_{i=1}^{11} [a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_3 + x_4}]^2$	$[-5, 5]$	0.00030
$f_9(x) = -\sum_{i=1}^{10} [(x - a_i)(x - a_i)^T + c_i]^{-1}$	$[0, 10]$	-10.5363

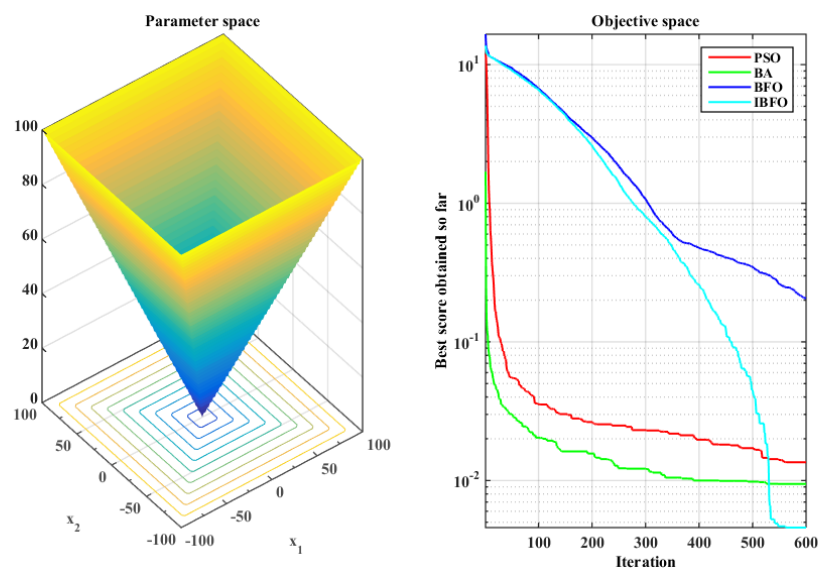
This paper selected nine multidimensional benchmark functions for the experiment to illustrate the effectiveness of the proposed IBFO algorithm. The nine benchmark functions are made up of five unimodal ( $f_1$ – $f_5$ ) benchmark functions, one multimodal ( $f_6$ ) benchmark functions and three fixed-dimension multimodal ( $f_7$ – $f_9$ ) benchmark functions. Moreover, the performance of the IBFO is also compared with the PSO, Bat algorithm (BA) and conventional BFO. As mentioned in the literature, the proposed IBFO has superior results for these benchmark functions compared to the PSO, BA and original BFO. A total of 30 independent runs of the four algorithms were performed on each benchmark function. The maximum number of iterations and population size for all algorithms were set to 600 and 50, respectively. We recorded the average (Ave) and standard deviation (Std) for each benchmark problem.

Table 3 shows the detailed statistical results for the nine multidimensional benchmark functions when dimension = 2 for different algorithms (PSO, BA, BFO, IBFO), respectively. Based on the results in Table 3, IBFO can provide very competitive results. According to the statistical result of  $f_1$ – $f_6$ , it can be seen that IBFO has better searching ability and is more stable, which means that it also has better robustness. According to the performance of  $f_1$ – $f_9$  in Figure 2, it is obvious that although the proposed IBFO can converge slowly in the early stages in some functions, the final result is optimal as the number of iterations increases.

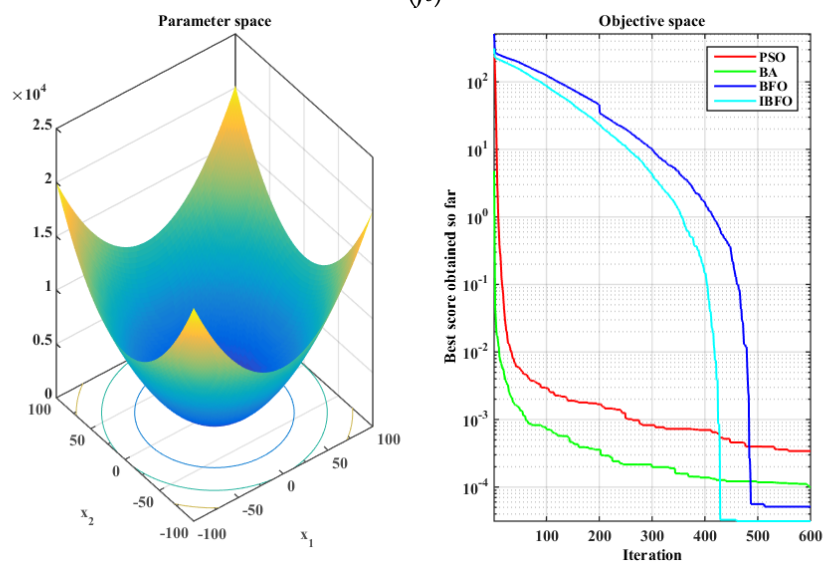
**Table 3.** Results of testing nine multidimensional benchmark functions.

	Methods							
	PSO		BA		BFO		IBFO	
	Ave	Std	Ave	Std	Ave	Std	Ave	Std
$f_1$	0.0185	0.0097	0.0113	0.0053	0.0075	0.0042	0.0057	0.0031
$f_2$	0.0006	0.0005	0.0001	0.0001	$4.07 \times 10^{-5}$	$4.76 \times 10^{-5}$	$2.83 \times 10^{-5}$	$4.83 \times 10^{-5}$
$f_3$	0.0135	0.0079	0.0095	0.0045	0.2072	1.1059	0.0046	0.0029
$f_4$	0.0003	0.0003	$9.88 \times 10^{-5}$	0.0001	$5.11 \times 10^{-5}$	0.0001	$3.13 \times 10^{-5}$	$3.76 \times 10^{-5}$
$f_5$	0.0070	0.0045	0.0020	0.0016	0.00045	0.0003	0.0004	0.0003
$f_6$	-793.55	65.4056	-763.56	77.9863	-720.08	76.2664	-797.41	56.1899
$f_7$	1.9873	1.5529	2.8100	1.9357	1.7915	1.0204	1.3947	0.8472
$f_8$	0.0012	0.0002	0.0081	0.0095	0.0007	0.0002	0.0007	0.0002
$f_9$	-5.1904	1.9864	-5.8112	3.1206	-10.3323	0.9752	-10.5104	0.0135

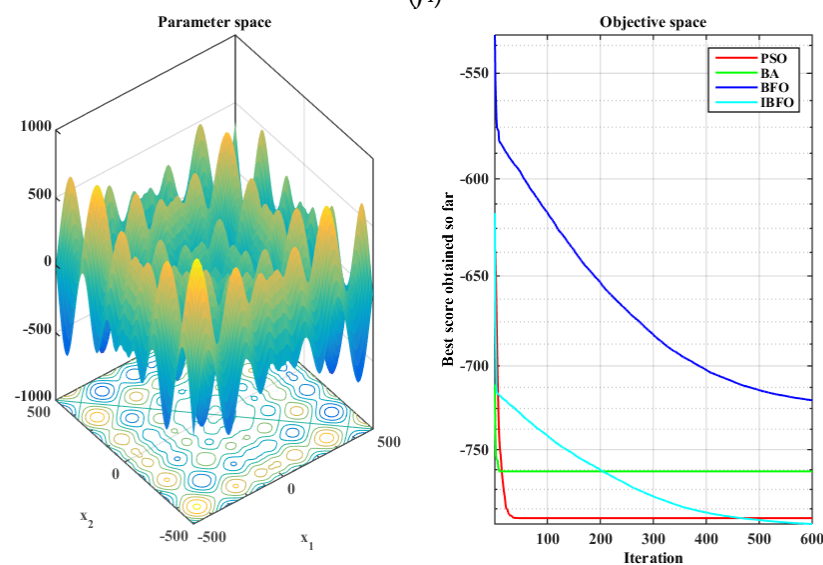
(f<sub>1</sub>)(f<sub>2</sub>)



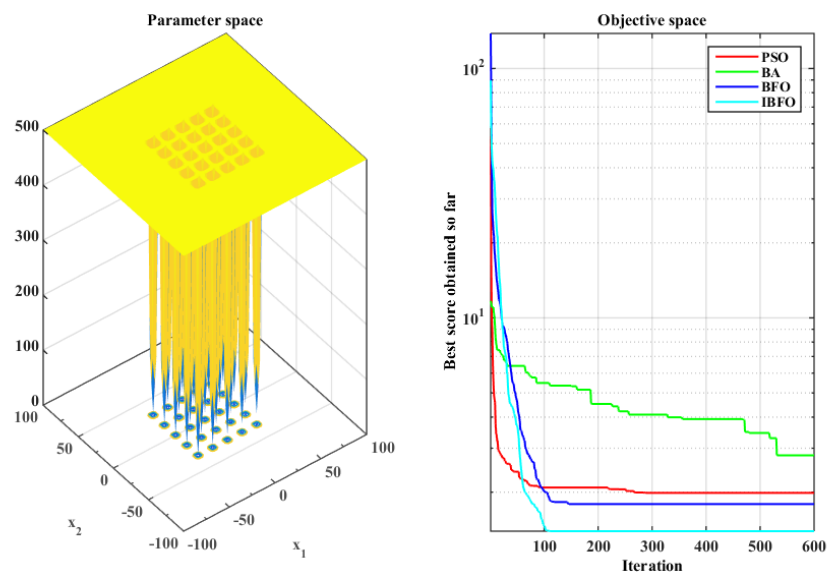
( $f_3$ )



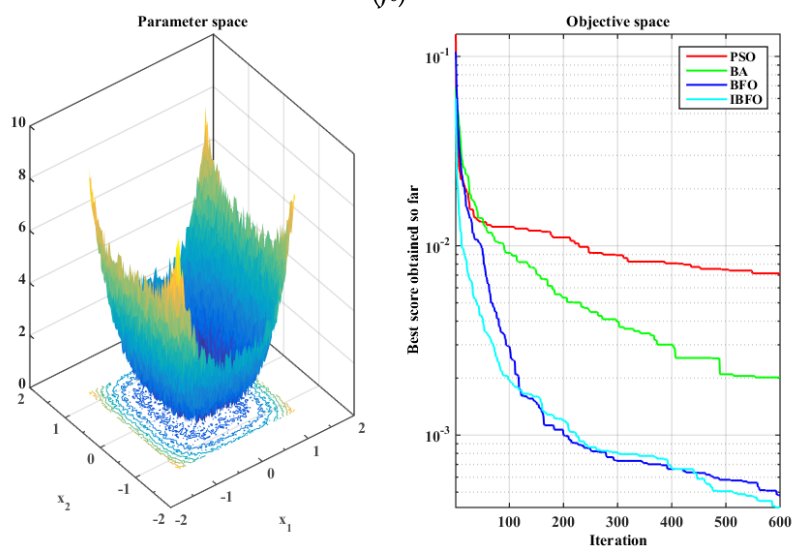
( $f_4$ )



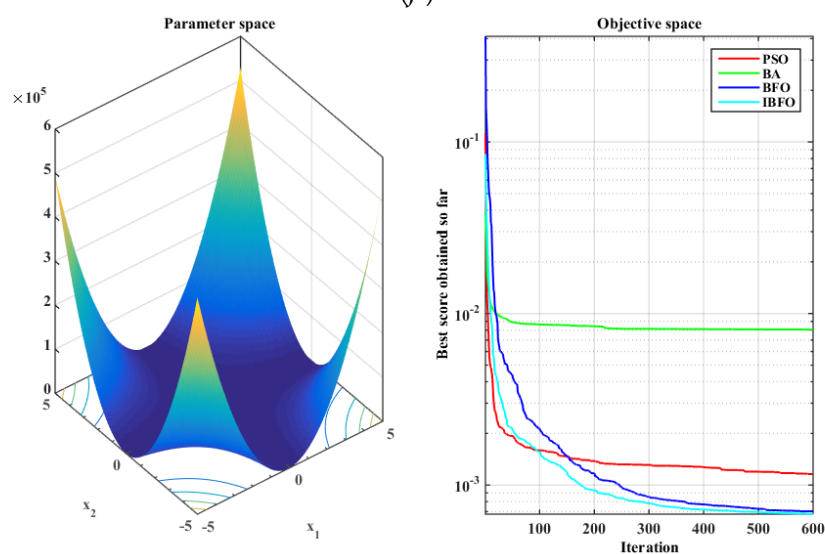
( $f_5$ )



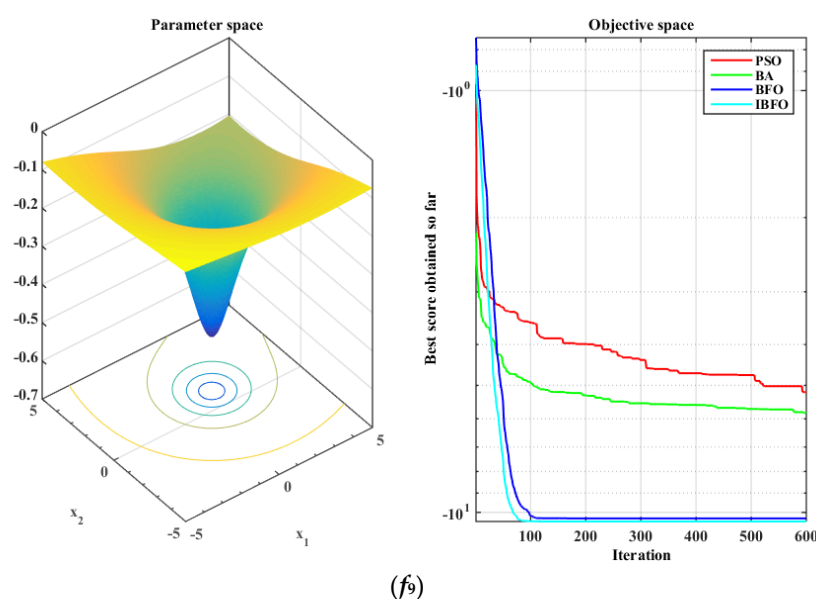
(f<sub>6</sub>)



(f<sub>7</sub>)



(f<sub>8</sub>)



**Figure 2.** Convergence curves of the nine multidimensional benchmark functions for the four algorithms when dimension = 2.

## 5.2. Results of the Somatization Disorder Diagnosis

Previous studies have shown that the chemotaxis step size of the bacterial optimization algorithm plays an important role in the search ability of bacteria. Thus this paper first analyzed and studied the influence of this on the performance of IBFO-KELM. Table 4 shows that the IBFO-KELM obtained the classification results in the psychological correction data under different chemotaxis step sizes. The data shown in the table mainly consist of mean and variance. As can be seen from the table, when the chemotaxis step size value is 0.5, the IBFO-KELM model achieved the best results with 96.97% classification accuracy, 0.9243 MCC, 97.29% sensitivity and 96.00% specificity. In addition, the variance of the model on the ACC index is the smallest when the value is 0.15, which indicates that the model has the most stable performance at this value. Therefore, in the follow-up experiment, we will take the chemotaxis step size value of 0.15 as the experimental parameter.

**Table 4.** The classification results of IBFO-KELM (kernel extreme learning machine) with different chemotaxis step size.

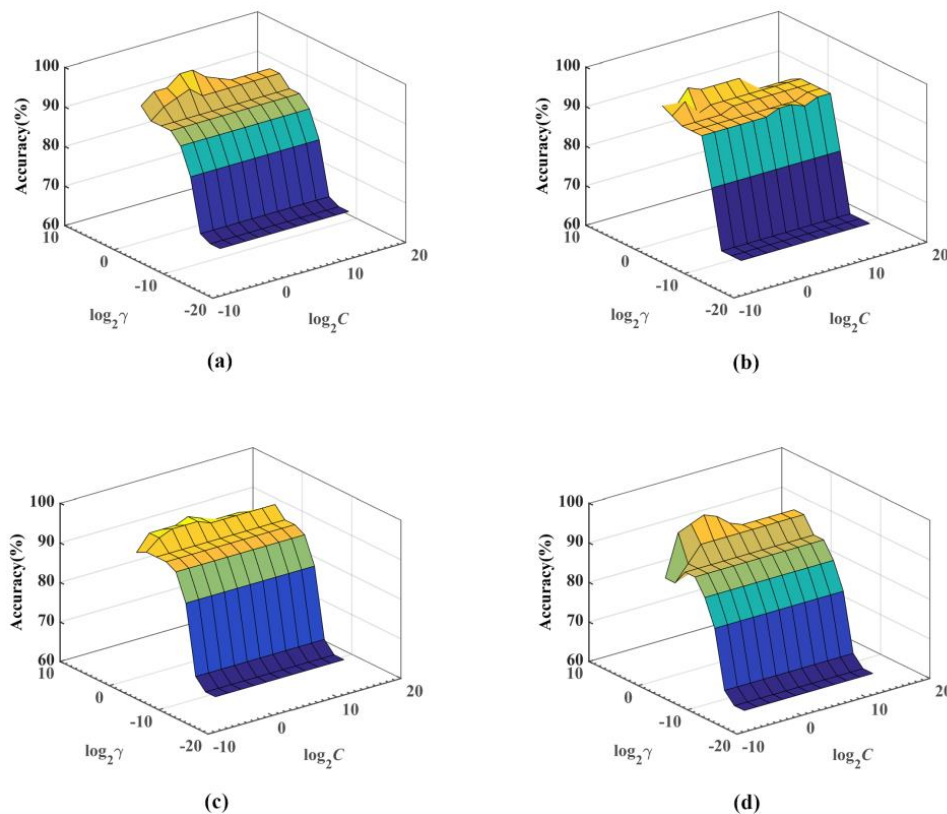
Step Size	IBFO-KELM			
	ACC	MCC	Sensitivity	Specificity
0.05	0.9213 (0.0389)	0.8227 (0.0903)	0.9679(0.035)	0.8286 (0.0768)
0.1	0.9402 (0.0362)	0.8653 (0.0824)	0.9713 (0.0282)	0.8786 (0.0678)
0.15	0.9697 (0.0351)	0.9243 (0.0907)	0.9729 (0.0351)	0.9600 (0.0843)
0.2	0.9476 (0.0512)	0.8850 (0.1089)	0.9679 (0.0544)	0.9071 (0.0828)
0.25	0.9378 (0.0427)	0.8614 (0.0965)	0.9786 (0.0301)	0.8571 (0.1117)
0.3	0.9211 (0.0377)	0.8241 (0.0836)	0.9675 (0.0365)	0.8286 (0.1075)

In order to verify the effectiveness of the proposed method, we proposed a comparison study with four other efficient machine learning models, namely Grid-KELM, BFO-KELM, GA-KELM and PSO-KELM. The comparison of the five methods in terms of the average ACC, MCC, sensitivity, and specificity is shown in Table 5. As shown, the IBFO-KELM had the highest ACC among the five methods. IBFO-KELM yielded average results of 96.97% ACC, 0.9243 MCC, 97.29% sensitivity, and 96.00% specificity. BFO-KELM yielded an ACC of 93.29%, 0.8280 MCC, sensitivity of 95.86%, and specificity of 85.50%. The PSO-KELM yielded an ACC of 92.82%, 0.8056 MCC, sensitivity of 96.57%, and specificity of 80.50%. The GA-KELM yielded an ACC of 91.76%, 0.7775 MCC, sensitivity of 95.95%, and specificity of 80.00%. The Grid-KELM yielded the worst results with an ACC of 90.76%,

0.7592 MCC, sensitivity of 94.48%, and specificity of 79.00%. Figure 3 displays the surface of the training classification accuracies achieved by the KELM method for one fold via the grid search strategy, where the  $x$ -axis and  $y$ -axis are denoted by  $\log_2 C$  and  $\log_2 \gamma$ , respectively. In addition, each mesh node in the plane  $(x, y)$  of the training accuracy represents a combination of the parameters, and the  $z$ -axis denotes the training accuracy value obtained with each combination of the parameters.

**Table 5.** Classification performance obtained by the five methods in terms of ACC (classification accuracy), MCC (Matthew’s correlation coefficient), sensitivity, and specificity.

Method	Metrics			
	ACC	MCC	Sensitivity	Specificity
IBFO-KELM	$0.9697 \pm 0.0351$	$0.9243 \pm 0.0907$	$0.9729 \pm 0.0351$	$0.9600 \pm 0.0843$
BFO-KELM	$0.9329 \pm 0.0362$	$0.8280 \pm 0.0850$	$0.9586 \pm 0.0491$	$0.8550 \pm 0.1012$
PSO-KELM	$0.9282 \pm 0.0266$	$0.8056 \pm 0.0813$	$0.9657 \pm 0.0362$	$0.8050 \pm 0.1383$
GA-KELM	$0.9176 \pm 0.0662$	$0.7775 \pm 0.1879$	$0.9595 \pm 0.0349$	$0.8000 \pm 0.2494$
Grid-KELM	$0.9076 \pm 0.0679$	$0.7592 \pm 0.1637$	$0.9448 \pm 0.0724$	$0.7900 \pm 0.1647$



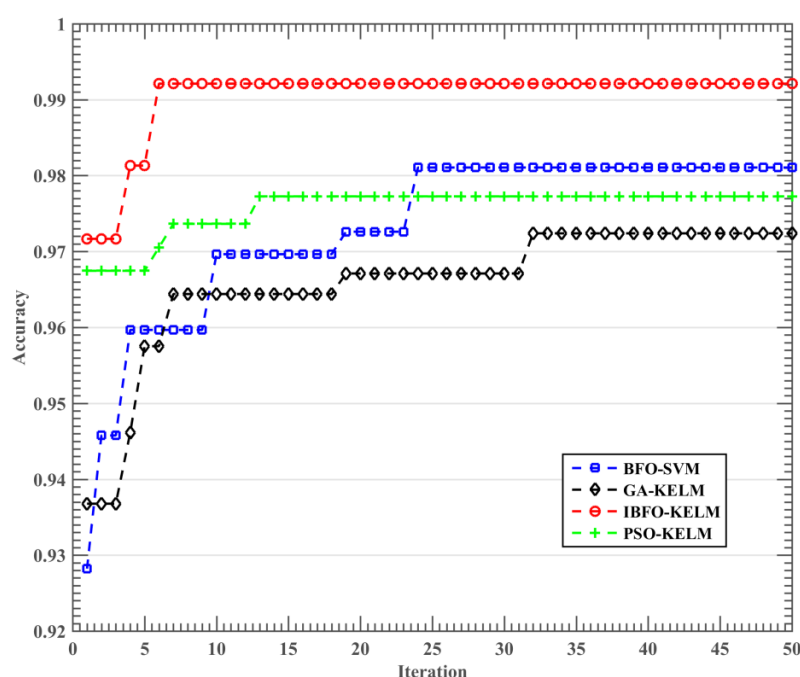
**Figure 3.** Training accuracy surfaces of KELM and KELM with parameters obtained by the grid-search based KELM (Grid-KELM) for several folds. (a) Training accuracy surface of fold 2 for KELM; (b) Training accuracy surface of fold 4 for KELM; (c) Training accuracy surface of fold 6 for KELM; (d) Training accuracy surface of fold 8 for KELM.

In order to reveal subtle differences in the classification performance of the methods, the non-parametric statistical test Wilcoxon’s rank-sum was used to check whether or not the improvement achieved by the IBFO-KELM was statistically significant. A  $p$ -value of less than 0.05 indicates statistical significance in the experiment. Table 6 shows that IBFO-KELM has significantly better results than the other four methods in terms of ACC and MCC. In Table 6, significant values that are greater than 0.05 are shown in bold.

**Table 6.**  $p$ -values of the Wilcoxon test of IBFO-KELM results versus other four methods ( $p > 0.05$  are shown in bold).

Method	$p$ -Value			
	ACC	MCC	Sensitivity	Specificity
BFO-KELM	0.03	0.03	0.24	0.06
PSO-KELM	0.02	0.02	0.41	0.02
GA-KELM	0.03	0.04	0.61	0.08
Grid-KELM	0.02	0.02	0.23	0.02

In order to describe the convergence of the proposed IBFO algorithm, we also recorded the trend that the accuracy (of the various KELM models based on swarm intelligence algorithms in the experiment) changed with population iteration. It is found from Figure 4 that the IBFO-KELM model can quickly converge to the best accuracy in the sixth iteration during the process of training, suggesting that the IBFO algorithm has a strong global search ability to avoid the algorithm being trapped prematurely in the local optimum. The main reason for this is that the opposite learning strategy plays a role in regulating the diversity of the population so as to accelerate the convergence of the whole population to the optimal solution. The original BFO-KELM model requires 24 iterations to converge to the optimal solution, and the obtained solution is less than that of IBFO. The PSO-KELM model converged faster than BFO, but the fitness value was less than BFO. The GA-KELM model obtained the maximum number of iteration times for the optimal solution, and the obtained solution was the lowest in the four methods, mainly because the GA algorithm in the current data has poor searching ability.



**Figure 4.** Relationship between the iteration and training accuracy of improved bacterial foraging optimization based KELM (IBFO-KELM), bacterial foraging optimization (BFO-KELM), particle swarm optimization based KELM (PSO-KELM) and genetic algorithm (GA-KELM).

In order to show the effectiveness of the proposed method, we also carried out the experiment using the other classifiers including support vector machines (SVM), random forest (RF) and Naive Bayes (NB) for comparison. The grid search method was also used to search for the two values for the radial basis function (RBF) kernel of SVM. The number of trees (ntree) and variables (mtry) were chosen from the range of ntree  $\in [50, 500]$  and mtry  $\in [1, 6]$  with a step of one, and the optimal results was taken for comparison. The detailed results of the above three methods are shown in Table 7. We



found that RF achieved the best results among the three methods, while NB performed the worst. SVM ranked between NB and RF. Comparing Tables 5 and 7, we find that KELM can achieve better results than SVM and that the proposed IBFO-KELM can achieve much better results than all the other methods.

**Table 7.** Detailed results for random forest (RF), Naïve Bayes (NB) and (SVM).

Method	Metrics			
	ACC	MCC	Sensitivity	Specificity
NB	0.9182 ± 0.0543	0.7766 ± 0.1495	0.9800 ± 0.0322	0.7350 ± 0.1634
SVM	0.8971 ± 0.0735	0.7122 ± 0.2148	0.9595 ± 0.0469	0.7100 ± 0.2601
RF	0.9382 ± 0.0405	0.8337 ± 0.1133	0.9652 ± 0.0367	0.8500 ± 0.1414

## 6. Conclusions and Future Work

In this study, a kind of kernel learning machine model (IBFO-KELM) based on an oppositional bacterial foraging optimization algorithm is proposed to predict the severity of somatization disorder in community correction personnel. The main innovation point of this article is the improved bacterial foraging optimization algorithm (IBFO) proposed on the basis of opposition-based learning strategy. Compared with the original BFO algorithm, this method can not only obtain a solution of the higher quality, but also with a faster convergence speed. Based on this method, KELM can obtain better parameters and higher prediction performance. The experimental results show that the IBFO-KELM model has better performance in terms of ACC, MCC, sensitivity and specificity than the other four KELM models.

Therefore, we can certainly come to the conclusion that the proposed intelligent prediction model can well predict the severity of somatization disorder in community correction personnel and provide psychological workers or doctors with meaningful information for clinical decisions and reference. In future work, we will continue to improve the methodology presented in this study, enabling it to be used to identify key factors that affect the physical impairment of community correctional personnel. In addition, we consider collecting more data samples, further improving the performance of the whole model, and introducing the proposed models to other similar mental disease diagnoses.

**Acknowledgments:** This research is funded by the National Natural Science Foundation of China (61602206, 61702376), Zhejiang Provincial Natural Science Foundation of China (LY17F020012, LY18F020022), Science and Technology Plan Project of Wenzhou of China (ZG2017019).

**Author Contributions:** Huiling Chen and Xinen Lv conceived and designed the experiments; Qian Zhang performed the experiments; Gang Wang and Xujie Li analyzed the data; Hui Huang contributed reagents/materials/analysis tools; Huiling Chen wrote the paper.

**Conflicts of Interest:** The authors declare that there is no conflict of interests regarding the publication of article.

## References

1. Holli, M.M.; Sammallahiti, P.R.; Aalberg, V.A. A finnish validation study of the SCL-90. *Acta Psychiatr. Scand.* **1998**, *97*, 42–46.
2. Derogatis, L.R.; Lipman, R.S.; Covi, L. SCL-90: An outpatient psychiatric rating scale—Preliminary report. *Psychopharmacol. Bull.* **1973**, *9*, 13–28.
3. Huang, G.-B.; Zhou, H.; Ding, X.; Zhang, R. Extreme learning machine for regression and multiclass classification. *IEEE Trans. Syst. Man Cybern. Part B* **2012**, *42*, 513–529.
4. Pal, M.; Maxwell, A.E.; Warner, T.A. Kernel-based extreme learning machine for remote-sensing image classification. *Remote Sens. Lett.* **2013**, *4*, 853–862.
5. Chen, C.; Li, W.; Su, H.; Liu, K. Spectral-spatial classification of hyperspectral image based on kernel extreme learning machine. *Remote Sens.* **2014**, *6*, 5795–5814.
6. Garea, A.S.; Heras, D.B.; Argüello, F. Gpu classification of remote-sensing images using kernel ELM and extended morphological profiles. *Int. J. Remote Sens.* **2016**, *37*, 5918–5935.



7. Liu, T.; Hu, L.; Ma, C.; Wang, Z.Y.; Chen, H.L. A fast approach for detection of erythematous diseases based on extreme learning machine with maximum relevance minimum redundancy feature selection. *Int. J. Syst. Sci.* **2015**, *46*, 919–931.
8. Ma, C.; Ouyang, J.; Chen, H.-L.; Zhao, X.-H. An efficient diagnosis system for parkinson's disease using kernel-based extreme learning machine with subtractive clustering features weighting approach. *Comput. Math. Methods Med.* **2014**, *2014*, 985789.
9. Li, Q.; Chen, H.; Huang, H.; Zhao, X.; Cai, Z.; Tong, C.; Liu, W.; Tian, X. An enhanced grey wolf optimization based feature selection wrapped kernel extreme learning machine for medical diagnosis. *Comput. Math. Methods Med.* **2017**, *2017*, 15.
10. Chen, H.-L.; Wang, G.; Ma, C.; Cai, Z.-N.; Liu, W.-B.; Wang, S.-J. An efficient hybrid kernel extreme learning machine approach for early diagnosis of parkinson's disease. *Neurocomputing* **2016**, *184*, 131–144.
11. Wang, M.; Chen, H.; Yang, B.; Zhao, X.; Hu, L.; Cai, Z.; Huang, H.; Tong, C. Toward an optimal kernel extreme learning machine using a chaotic moth-flame optimization strategy with applications in medical diagnoses. *Neurocomputing* **2017**, *267*, 69–84.
12. Zhao, D.; Huang, C.; Wei, Y.; Yu, F.; Wang, M.; Chen, H. An effective computational model for bankruptcy prediction using kernel extreme learning machine approach. *Comput. Econ.* **2017**, *49*, 325–341.
13. Wang, M.; Chen, H.; Li, H.; Cai, Z.; Zhao, X.; Tong, C.; Li, J.; Xu, X. Grey wolf optimization evolving kernel extreme learning machine: Application to bankruptcy prediction. *Eng. Appl. Artif. Intell.* **2017**, *63*, 54–68.
14. Deng, W.-Y.; Zheng, Q.-H.; Wang, Z.-M. Cross-person activity recognition using reduced kernel extreme learning machine. *Neural Netw.* **2014**, *53*, 1–7.
15. Liu, B.; Tang, L.; Wang, J.; Li, A.; Hao, Y. 2-D defect profile reconstruction from ultrasonic guided wave signals based on qga-kernelized elm. *Neurocomputing* **2014**, *128*, 217–223.
16. Zhao, X.; Li, D.; Yang, B.; Liu, S.; Pan, Z.; Chen, H. An efficient and effective automatic recognition system for online recognition of foreign fibers in cotton. *IEEE Access* **2016**, *4*, 8465–8475.
17. Uçar, A.; Özalp, R. Efficient android electronic nose design for recognition and perception of fruit odors using kernel extreme learning machines. *Chemom. Intell. Lab. Syst.* **2017**, *166*, 69–80.
18. Peng, C.; Yan, J.; Duan, S.; Wang, L.; Jia, P.; Zhang, S. Enhancing electronic nose performance based on a novel qpso-kelm model. *Sensors* **2016**, *16*, 520.
19. Zeng, Y.; Xu, X.; Shen, D.; Fang, Y.; Xiao, Z. Traffic sign recognition using kernel extreme learning machines with deep perceptual features. *IEEE Trans. Intell. Transp. Syst.* **2017**, *18*, 1647–1653.
20. Li, J.; Li, D.C. Wind power time series prediction using optimized kernel extreme learning machine method. *Acta Phys. Sin.* **2016**, *65*, 130501.
21. Avci, D.; Dogantekin, A. An expert diagnosis system for parkinson disease based on genetic algorithm-wavelet kernel-extreme learning machine. *Park. Dis.* **2016**, *2016*, doi:10.1155/2016/5264743.
22. Veronese, N.; Luchini, C.; Nottegar, A.; Kaneko, T.; Sergi, G.; Manzato, E.; Solmi, M.; Scarpa, A. Prognostic impact of extra-nodal extension in thyroid cancer: A meta-analysis. *J. Surg. Oncol.* **2015**, *112*, 828–833.
23. Lu, H.; Du, B.; Liu, J.; Xia, H.; Yeap, W.K. A kernel extreme learning machine algorithm based on improved particle swarm optimization. *Memet. Comput.* **2017**, *9*, 121–128.
24. Passino, K.M. Biomimicry of bacterial foraging for distributed optimization and control. *IEEE Control Syst. Mag.* **2002**, *22*, 52–67.
25. Majhi, R.; Panda, G.; Majhi, B.; Sahoo, G. Efficient prediction of stock market indices using adaptive bacterial foraging optimization (ABFO) and bfo based techniques. *Expert Syst. Appl.* **2009**, *36*, 10097–10104.
26. Dasgupta, S.; Das, S.; Biswas, A.; Abraham, A. Automatic circle detection on digital images with an adaptive bacterial foraging algorithm. *Soft Comput.* **2010**, *14*, 1151–1164.
27. Mishra, S. A hybrid least square-fuzzy bacterial foraging strategy for harmonic estimation. *IEEE Trans. Evol. Comput.* **2005**, *9*, 61–73.
28. Mishra, S.; Bhende, C.N. Bacterial foraging technique-based optimized active power filter for load compensation. *IEEE Trans. Power Deliv.* **2007**, *22*, 457–465.
29. Ulagammai, M.; Venkatesh, P.; Kannan, P.S.; Prasad Padhy, N. Application of bacterial foraging technique trained artificial and wavelet neural networks in load forecasting. *Neurocomputing* **2007**, *70*, 2659–2667.
30. Wu, Q.; Mao, J.F.; Wei, C.F.; Fu, S.; Law, R.; Ding, L.; Yu, B.T.; Jia, B.; Yang, C.H. Hybrid BF-PSO and fuzzy support vector machine for diagnosis of fatigue status using emg signal features. *Neurocomputing* **2016**, *173*, 483–500.

31. Yang, C.; Ji, J.; Liu, J.; Liu, J.; Yin, B. Structural learning of bayesian networks by bacterial foraging optimization. *Int. J. Approx. Reason.* **2016**, *69*, 147–167.
32. Sivarani, T.S.; Joseph Jawhar, S.; Agees Kumar, C.; Prem Kumar, K. Novel bacterial foraging-based anfis for speed control of matrix converter-fed industrial bldc motors operated under low speed and high torque. *Neural Comput. Appl.* **2016**, 1–24. doi:10.1007/s00521-016-2652-6.
33. Cai, Z.; Gu, J.; Chen, H.L. A new hybrid intelligent framework for predicting parkinson's disease. *IEEE Access* **2017**, *5*, 17188–17200.
34. Zhao, W.; Wang, L. An effective bacterial foraging optimizer for global optimization. *Inf. Sci.* **2016**, *329*, 719–735.
35. Panda, R.; Naik, M.K. A novel adaptive crossover bacterial foraging optimization algorithm for linear discriminant analysis based face recognition. *Appl. Soft Comput. J.* **2015**, *30*, 722–736.
36. Nasir, A.N.K.; Tokhi, M.O.; Ghani, N.M.A. Novel adaptive bacterial foraging algorithms for global optimisation with application to modelling of a trs. *Expert Syst. Appl.* **2015**, *42*, 1513–1530.
37. Tizhoosh, H.R. Opposition-based learning: A new scheme for machine intelligence. In Proceedings of the International Conference on Computational Intelligence for Modelling, Control and Automation, CIMCA 2005 and International Conference on Intelligent Agents, Web Technologies and Internet, Vienna, Austria, 28–30 November 2005; pp. 695–701.
38. Salzberg, S.L. On comparing classifiers: Pitfalls to avoid and a recommended approach. *Data Min. Knowl. Discov.* **1997**, *1*, 317–328.
39. Statnikov, A.; Tsamardinos, I.; Dosbayev, Y.; Aliferis, C.F. GEMS: A system for automated cancer diagnosis and biomarker discovery from microarray gene expression data. *Int. J. Med. Inform.* **2005**, *74*, 491–503.



© 2018 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).