IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 15, 2021, accepted February 16, 2021, date of publication February 19, 2021, date of current version March 2, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3060621

IDP: An Intelligent Data Prediction Scheme Based on Big Data and Smart Service for Soil Heavy Metal Content Prediction

FANG CHEN, CONG ZHANG, JUNJIE ZHANG, AND WENQI CAO
School of Mathematics and Computer Science, Wuhan Polytechnic University, Wuhan 430023, China

Corresponding author: Cong Zhang (hb_wh_zc@163.com)

This work was supported in part by the Major Technical Innovation Projects of Hubei Province under Grant 2018ABA099, in part by the National Science Fund for Youth of Hubei Province of China under Grant 2018CFB408, in part by the Natural Science Foundation of Hubei Province of China under Grant 2015CFA061, in part by the National Natural Science Foundation of China under Grant 61272278, and in part by the Research on Key Technologies of Intelligent Decision-Making for Food Big Data under Grant 2018A01038.

ABSTRACT In the application of regression prediction through big data technology, the error between the predicted value and the true value is often large. In order to reduce the error of data prediction, this paper proposes an Intelligent Data Prediction (IDP) scheme for Smart Service. It uses Least Squares Support Vector Machine (LSSVM) as the basic prediction model. Since there is no standard procedure for determining the main parameters of LSSVM, an improved Particle Swarm Optimization (MBPSO) algorithm is used to simultaneously optimize the parameters of LSSVM. The main disadvantage of PSO is precocity due to the disappearance of population diversity. Based on this, Improvement strategy of MBPSO aims to continuously generate ''More'' and ''Better'' particles. First, in order to avoid the early disappearance of particle diversity, MBPSO re-adjusted the inertia weight and learning factor. Secondly, a renewable access strategy is proposed to allow a part of the disappeared population to regenerate. Finally, the method of global optimal adjustment is introduced to help particles find the optimal flight direction. In order to verify the effectiveness of MBPSO, 9 test functions are used to test the algorithm performance. The results show that MBPSO's optimization speed, best and mean all perform best. Taking the farmland soil heavy metal data sets of Dongxihu District and Hannan District of Wuhan City as examples of application, the content of heavy metals Cr and Pb in the soil was predicted. The results show that the predicted value of IDP is closer to the actual value, and the three error index values are significantly lower than other models. Especially in the prediction of Pb content, compared with the LSSVM model, the prediction errors of the two regions are reduced by 25.67% and 20.70% respectively. We can conclude that the proposed IDP scheme has practical significance in data prediction.

INDEX TERMS Big data, smart service, intelligent data prediction (IDP), improved particle swarm optimization (MBPSO), least square support vector machine (LSSVM).

I. INTRODUCTION

In the era of big data information, the processing, analysis and prediction of data can help us solve a lot of problems [1]. Machine learning (ML) is an indispensable method for big data prediction. ML is mainly a computer learning a calculation method by learning rules from complex data [2]. There are already many ML methods for data prediction. They are: Logistic Regression [3], prototype-based objective function clustering method (K-Means) [4], a graphical method using probability analysis (Decision Trees) [5], using a graphical method of probability analysis (Random Forest) [6], Support Vector Machine (SVM) [7] and Artificial Neural Networks (ANN) [8] etc.

Many scholars use ML to build big data prediction models to solve practical problems. Some applications are examples. Based on OMI observations in previous years, the environmental SO2 concentration and its exposure risk in future years is inferred. The estimated ground-level SO2 concentrations were in strong agreement with the ground observations

The associate editor coordinating the review of this manuscript and approving it for publication was Fahmi Khalifa.

(R = 0.86, root-mean-square error = 10.49 μg/m3, relative prediction error = 19%). [9]. Coal intensity change is predicted during the process of CO2 sequestration in coal seams. The experimental results show that the correlation coefficient (R), Root Mean Square Error (RMSE), Mean Absolute Error (MAE), and Mean Absolute Percentage Error (MAPE) values of the training set are 0.9963, 0.6582, 0.4282, 0.1088, and the R, RMSE, MAE, and MAPE values of the test set are 0.9956, 0.7634, 0.5229, 0.1912, respectively [10]. Soil heavy metal concentration is predicted. After learning and training, the prediction accuracy of 5 elements: Hg, As, Cd, Cr and Pb, are 98.52%, 98.22%, 91.86%,90.70% and 88.31%. [11]. In these examples, how to establish a model for smart service and it is the focus to predict the unknown variables with the same characteristics according to known feature variables easier to obtain.

The Swarm Intelligence (SI) mainly simulates the swarm behavior of insects, animals, birds and fish. The application of SI is very extensive. These are some examples. The use of Swarm Intelligence can extend the life of wireless sensor networks [12]. SI is applied to dynamic system identification problems and license plate detection problems [13]. Usually, we select parameters based on prior experience or trial and error. But this method is awkward, time-consuming and labor-intensive, and it may not always find the ideal parameters of the model. SI can solve this problem perfectly. SI algorithms take the parameters needed to be optimized in ML as the optimization target, and take the result of ML as the fitness value of optimization. Common SI algorithms include Ant Colony Optimization (ACO) [14], Particle Swarm Optimization (PSO) [15], Gray Wolf Optimizer (GWO) [16], etc.

This paper proposes an Intelligent Data Prediction (IDP) scheme. IDP can provide a Smart Service for predicting the content of heavy metals in the soil. The program uses Least Squares Support Vector Machine (LSSVM) as the basic model, and chooses an improved Particle Swarm Optimization (MBPSO) to help LSSVM establish the learning and training process. MBPSO re-adjusts the inertia weight and learning factor of PSO to make it more conducive to the flight of particles in the optimization process. Considering the importance of population diversity to the optimization process, a renewable access strategy is proposed to restore the vitality of certain particles. Introduce the global optimal particle, redefine the particle position update formula, and help the particle find a better optimization direction. This prevents the disappearance of population diversity to a certain extent. MBPSO takes the penalty factor C and the parameter σ of LSSVM as optimization targets. Through the particle search flight process, LSSVM performs multiple data learning processes and adjusts its own parameters.

The main contributions in this paper are as follows: 1) An Intelligent Data Prediction (IDP) scheme is proposed that combines MBPSO and Least Squares Support Vector Machine (LSSVM); 2) Propose an improved Particle Swarm Optimization (MBPSO); 3) Perform performance comparison test of MBPSO algorithm; 4) Optimal system parameters are obtained by PSO; 5) Some experiments were carried out to predict the content of heavy metals in soil; 6) An acceptable match is noticed between measured and predicted values.

The rest of this paper is organized as follows. Section II introduces the work related to the research of big data prediction models. Section III introduces PSO and LSSVM. Section IV proposes MBPSO and explains IDP realization process. Section V first conducts a performance comparison test on MBPSO, and then uses the farmland soil heavy metals as data to conduct IDP for simulation, comparison and analysis. Section VI conducts the work content of this paper to sum up.

II. RELATED WORK

Big data forecasting technology provides a powerful way for data forecasting. In recent years, many scholars have proposed Neural Networks (NNs) to solve the problem of big data prediction. NNs usually have good mapping capabilities. During data training, they can train well, learn and adjust themselves. Some examples are shown as follows. In 2019, B. Wang, W. Kong, H. Guan and N. N. Xiong proposed a Long Short Term Memory Neural Network (LSTM) to predict the PM2.5 value [17]. In 2020, C. Chen, N. N. Xiong, X. Guo and J. Ren used RBF-NN to predict the number of wars and the transformation of the historical stage [18]. In 2020, Z. Wang, J. Huang, N. N. Xiong, X. Zhou, X. Lin and T. L. Ward proposed SqueezeNet to accurately identify vehicle category [19].

The biggest shortcoming of NNs is its own inexplicability. They have no ability to explain their own reasoning process and reasoning basis. When the data is not sufficient, NNs work poorly. When the data is sufficient, overfitting is prone to occur. They are not enough that the generalization performance of the model generated by training. If the generalization performance of the model is not good enough, the prediction bias of unknown variables will be larger.

In addition to NNs, there are many other methods used in the study of the data prediction. Some examples are shown. In 2012, Kita E, Harada M and Mizuno T proposed Bayesian for stock prediction [20]. In 2016, Varez, Siwabessy, Tran applied Random Forest to predict sponge species richness [21]. In 2020, Z. Sai, C. Lu, S. Jiang, L. Shan, C. James and N. N. Xiong applied Support Vector Machine to Energy Management Optimization of Open-Pit Mine Solar Photothermal-Photoelectric Membrane Distillation [22].

The Bayesian algorithm does not have a high demand for data prediction and can predict quickly. The disadvantage of Random Forest is that it is easy to fall into overfitting. Support Vector Machine (SVM) has a better effect on non-linear sample training, but the generalization performance of SVM has certain limitations in data prediction. To solve these problems, this paper proposes to use Least Squares Support Vector Machine (LSSVM) as the basic model. This method replaces inequality constraints with equality constraints on the basis of SVM. It can well solve the problem of data

prediction since LSSVM maps low-latitude data to the high latitude space. They are some example. In 2016, Xiang Song, Xu Li, Weicheng Tang applied LSSVM decision model to calculating model probabilities according to the operating state of the vehicle [23]. In 2018, Xu Li, Chen Wei and Chan Chingyao applied LSSVM to predict and compensate for the INS position errors [24]. In 2020, Chengbing Yu, Ziwei Xi and Yilin Lu applied LSSVM to color prediction for cotton fabrics [25].

In the data prediction application of LSSVM, its generalization performance is affected by the penalty factor C and nuclear parameter σ. Therefore, for data prediction results of LSSVM, it is very important to choose parameters. Normally, these two parameters need to be manually set by experience. This leads to insufficient calculation accuracy of the model, or the need to adjust the model multiple times to seek higher accuracy. If C or σ is too large or small, it will easily lead to changes in the predicted result. At present, some ways are used to optimize parameters of LSSVM, such as ACO [26], GWO [27], PSO [28], [29] and other algorithms.

Based on the complexity of the calculation of the kernel function of LSSVM, in order to speed up the learning speed of the model training process, PSO is chosen to optimize the parameters of LSSVM in this paper. The advantage of PSO is simple operation and easy implementation. The basic idea of PSO is: the potential solution of each optimization problem is a particle in the search space, and the characteristics of each particle can be represented by speed, position, and fitness value. PSO also has certain limitations. For example, during the optimization process, the particles fall into the local optimal position, and as the number of iterations increases, the diversity of the particles decreases. Locally optimal, there are often the disadvantages of slow convergence speed and low convergence accuracy in the later stages of evolution [30], [31]. Therefore, many scholars have made a lot of method improvements to the algorithm change strategy. Some examples are as follows: MPSO is an algorithm improved the inertial weight and speed update [32]; IPSO1 is an algorithm improved the speed update [33]; IPSO2 is an algorithm improved the inertia weight and adaptive learning factor [34]; R-dPSO is an algorithm the random-driven global Particle Swarm Optimization [35]. Based on the above research, A new improved PSO (MBPSO) is proposed in this paper.

The previous research results show that the basic principle of improved PSO is to generate more different populations, or to fly to the better position. Based on the two ideas of more populations and better position, MBPSO is proposed. It is improved as follows. A new formula is designed to randomly generate inertial weights. The learning factor is adaptive collaborative adjustment with it within a certain range. Out-of-bounds particles are re-generated randomly within the space. particles tend to fly to the better position. Various improvements have enabled the particles update process to generate more optimizable populations, and all particles fly to better positions.

An Intelligent Data Prediction (IDP) scheme is proposed in this paper. IDP combines LSSVM and MBPSO to collaborate for smart service, so that it can perform high-dimensional spatial mapping of data, and can be generalized well. Within a certain range, IDP can accurately predict unknown variables with the same data characteristics.

III. SYSTEM MODEL AND DEFINITION

A. PATICLE SWARM OPTIMIZATION (PSO)

PSO was first proposed by Eberhart and Kennedy in the mid-1990s. It is an advanced Intelligent Swarm algorithm [36]. Each particle of PSO, such as $X_i = \{x_{i,1}, x_{i,2}, \cdots, x_{i,d}$, has a memory, which can track the particle in the last iteration. The best position pb_i and the global best position gb are track through last generation to update yourself by flying speed $V_i = \{v_{i,1}, v_{i,2}, \cdots, v_{i,d}$. The formula for updating the velocity and position of the particle is as follows.

$$v_{i,d}^{k+1} = v_{i,d}^{k} + c_1 r_1 \left(pb_{i,d}^{k} - x_{i,d}^{k}\right) + c_2 r_2 \left(gb^{k} - x_{i,d}^{k}\right); \quad (1)$$

$$x_{i,d}^{k+1} = x_{i,d}^{k} + v_{i,d}^{k}. \quad (2)$$

where m is the population size and $i = 1, 2, \cdots, m$; D is the particle dimension and $d = 1, 2, \cdots, D$; K is the maximum evolutionary algebra and $k = 1, 2, \cdots K$; r_1 and r_2 are random variables that obey uniform distribution $U(0, 1)$; c_1 and c_2 are learning factors.

Shi [37] introduced a linearly decreasing strategy of inertia weight based on PSO, which is called the basic PSO, and the strategy formula is as follows.

$$w = w_{max} - \frac{w_{max} - w_{min}}{K} \times k. \quad (3)$$

where w is the inertia weight, w_{max} and w_{min} are the maximum and minimum values of the inertia weight; the speed update formula is as follows.

$$v_{i,d}^{k+1} = w v_{i,d}^{k} + c_1 r_1 \left(pb_{i,d}^{k} - x_{i,d}^{k}\right) + c_2 r_2 \left(gb^{k} - x_{i,d}^{k}\right). \quad (4)$$

where k and K are the current number of evolutionary iterations and the maximum number of iterations.

B. LEAST SQUARE SUPPORT VECTOR MACHINE (LSSVM)

SVM regression technology is essentially an intelligent learning algorithm that receives input values or input vectors through a learning model. It uses a Machine Learning method similar to Neural Networks. It is also an intelligent learning algorithm that obtains output values through regression functions. The main application schematic diagram is shown in Figure 1.

LSSVM replaces inequality constraints with equality constraints on the basis of SVM, avoiding the problem of quadratic regression, and has higher calculation accuracy and efficiency [38]. Suppose the training data is $\{x_i, y_i\}, i = 1, 2, \ldots, n$ and $x_i \in R^n$ is the n-dimensional input vector, and $y_i \in R$ is the output vector. Set a non-linear function $\varphi(\cdot)$, let:

$$\psi(x) = (\varphi(x_1), \varphi(x_2), \cdots, \varphi(x_n)). \quad (5)$$

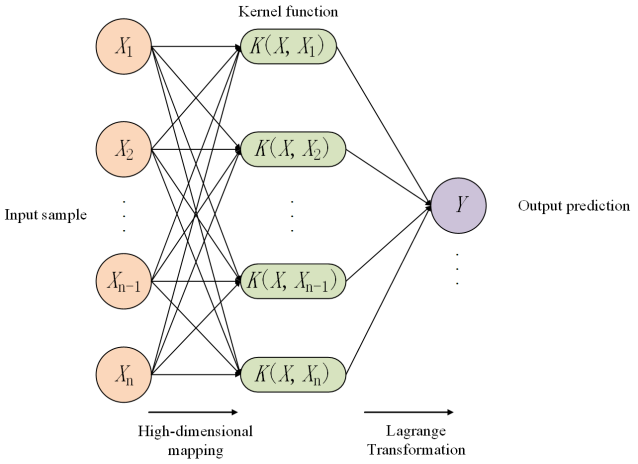

FIGURE 1. Support vector machine model.

Through this function, the training data is mapped to the high-dimensional feature space. Then, perform non-linear regression estimation on the processed sample data.

The objective function of LSSVM can be expressed as follows.

$$f(x) = q^T \cdot \varphi(x) + b. \tag{6}$$

where $\varphi(x)$ is a non-linear function, q^T is a weight vector; b is a paranoid vector. Then, use the structural risk minimization principle to determine the parameters q and b of the objective function. The structural risk expression adopted is as follows.

$$R = \frac{1}{2}C \cdot R_{emp} + \frac{1}{2}\| q \|^2. \tag{7}$$

where C is the penalty factor and R_{emp} is a loss function.

LSSVM usually uses a quadratic loss function, and its expression is as follows.

$$R_{emp} = \sum_{i=1}^{n} \xi_i^2. \tag{8}$$

where ξ_i is the error generated after predicted data of SVM.

Add constraints as follows.

$$y_i - q^T \varphi(x_i) = b + \xi_i. \tag{9}$$

Lagrange solution equation of the minimization function is as follows.

$$L(q, b, \xi, \lambda) = \frac{1}{2}\| q \|^2 + \frac{1}{2}C\sum_{i=1}^{n}\xi_i^2 - \sum_{i=1}^{n}\lambda_i\left[q^T\phi(x_i) + b + \xi_i - y_i\right]. \tag{10}$$

where $\lambda_i (i = 1, 2, \ldots, n)$ is Lagrange multiplier. Then, the partial derivatives of ξ, b, q, and λ in Lagrangian function are respectively obtained, and each partial derivative is equal to 0 as follows.

$$\begin{cases} \frac{\partial L}{\partial \xi} = 0 \Longrightarrow \lambda_i = C\xi_i \& \\ \frac{\partial L}{\partial b} = 0 \Longrightarrow \sum_{i=1}^{n}\lambda_i = 0 \\ \frac{\partial L}{\partial q} = 0 \Longrightarrow q = \sum_{i=1}^{n}\lambda_i\phi(x_i) \\ \frac{\partial L}{\partial \lambda} = 0 \Longrightarrow q^T\varphi(x_i) + b + \xi_i - y_i = 0 \end{cases}. \tag{11}$$

Eliminate weight coefficient q and the error variable ξ_i in equation (11), and convert to solve equation as follows.

$$\begin{bmatrix} 0 & Q^T \\ Q & K + C^{-1}I \end{bmatrix}\begin{bmatrix} b \\ \lambda \end{bmatrix} = \begin{bmatrix} 0 \\ Y \end{bmatrix}. \tag{12}$$

where: I is the unit matrix of order l; $Q = [1, 1, \cdots, 1]^T$; $\lambda = [\lambda_1, \lambda_2, \cdots, \lambda_n]^T$; $Y = [y_1, y_2, \cdots y_n]^T$. K is the kernel function matrix, whose elements $K_{ij} = \phi(x_i)^T\phi(x_j)$; $i, j = 1, 2, \cdots, n$; its regression function is as follows.

$$y_i = \sum_{i=1}^{l}\lambda_i K(x, x_i) + b. \tag{13}$$

Gaussian Radial Basis Function (RBF) has good local feature extraction capabilities and smoothing characteristics. Due to the characteristics of non-linear data, RBF is used in LSSVM in this paper, and the kernel function is as follows.

$$K(x_i, x_j) = \exp\left(-\frac{\| x_i - x_j \|^2}{2\sigma^2}\right). \tag{14}$$

where σ is nuclear parameter.

LSSVM regression function is as follows.

$$y(x) = \sum_{i=1}^{l}\lambda_i \exp\left(-\frac{\| x_i - x_j \|^2}{2\sigma^2}\right) + b. \tag{15}$$

LSSVM training and generalization ability is directly affected by the penalty coefficient C and the kernel parameter σ. Therefore, it is extremely important to adjust the values of these two parameters when applying LSSVM. However, in actual problems, these two parameters are not necessarily related. In the past, empirical estimation methods or trial and error methods are usually used for parameter selection. Both methods are very time-consuming and laborious, and may cause large deviations.

IV. OUR PROPOSED SCHEME OF IDP

An improved PSO (MBPSO) is proposed in this paper, aiming to find "More" and "Better" particles. MBPSO is combined with LSSVM for smart service to form an Intelligent Data Prediction (IDP) scheme. The scenario application of IDP is shown as Figure 2. In dataset1, LSSVM performs model training on known input and output variables, and adjusts its fitness value in conjunction with MBPSO. After repeated training, save the trained model as IDP. The known input variables in dataset2 are easier to obtain in practical applications, and the output variables are often difficult to obtain. Some variables difficult to obtain in actual sampling can be more accurately predicted by IDP.

A. IMPROVED PSO (MBPSO)

PSO is an evolutionary calculation method based on the intelligent collective behavior of certain animals. It is easy to implement, and few parameters need to be adjusted. In the basic PSO optimization process, it is prone to stagnation. The reason for the continuous reduction of the particle population is: some particles fly outside the feasible region; some particles track the previous generation of particles during the

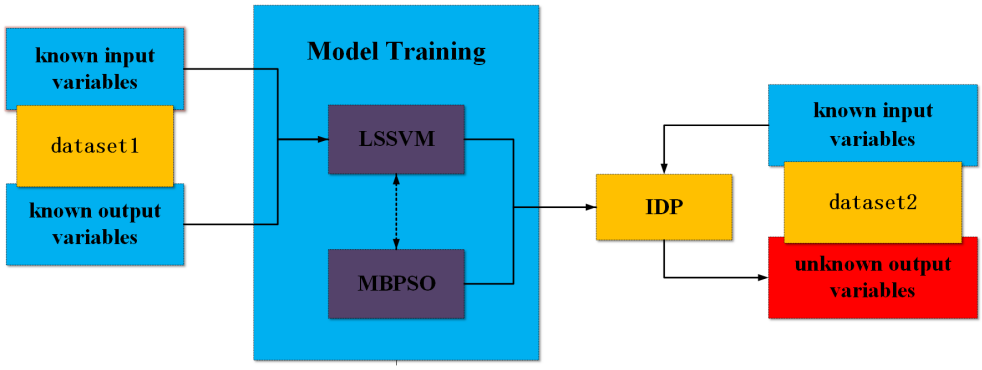

FIGURE 2. Scene application diagram of IDP.

update process, and disappear in the process of replacement of better particles. Therefore, in the particle update process, it is very important to improve the basic PSO by generating more different particle populations. The performance of PSO largely depends on the parameter selection strategy. Therefore, the cooperation between parameters determines whether it can promote particle optimization. The parameter generation strategy and value range setting are one of the keys to improve PSO. To generate more populations and fly to a better position as the idea, an improved Particle Swarm Optimization (MBPSO) is proposed.

1) RANDOM INERTIA WEIGHT

The flying speed of the particles is to the search step length of the particles in space, which is related to the convergence of the algorithm. The larger the value of w, the larger the particle flight search range, which can jump out of the local optimum; the smaller the value of w, the smaller the particle flight search range, which has a stronger local search capability and can speed up convergence [39].

If the particles reach a better position after updating less frequently, and the value of the inertia weight is too large, it is not conducive to local search. If the particles have not reached a better position after more updates, and the value of the inertia weight is too small, it is not conducive for them to fly to a better position. Both of these situations will reduce the diversity of the particles and tend to converge prematurely. The paper [40] pointed out: in the early stage, random inertia weights can prevent particles from converging prematurely due to excessive inertia weights; in the later stage, it can reduce the possibility that the inertia weight is too small to cause the optimization process to stagnate and fall into the local optimum. The paper [35] shows that the random inertia weight falling between $[-1, 1]$ can balance local and global search capabilities of algorithm. According to the analysis of inertia weight and the optimization process of PSO, a random inertia weight strategy is proposed in this paper as follows.

$$w = w_{min} + \theta \times (w_{max} - w_{min}) \tag{16}$$

where $w_{min} = -1$, $w_{max} = 1$, $w \in (-1, 1)$, θ is a random number uniformly between [0,1].

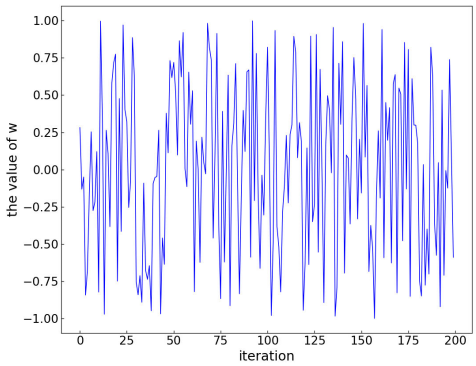

FIGURE 3. The value of the random inertia weight for 200 iterations.

Figure 3 shows the value of the inertia weight is randomly selected between $[-1, 1]$ during 200 iterations. The random value of the inertia weight helps to generate more new populations during the particle update process.

2) ADAPTIVE LEARNING FACTOR

The learning factor c_1 represents the ability of particles to approach the individual's best position pb, giving the particles the ability to self-summarize. The learning factor c_2 represents the ability of particles to approach the global best position gb, giving particles the ability to learn to the global optimal particle. Through adjustment, the magnitude relationship between the learning factor c_1 and c_2 can balance the ability of global search in the early stage and local convergence in the later stage [41].

During the particles update process, the learning factor c_1 decreases and c_2 increases to help the particles fly to a better position. The paper [35] shows that when w is randomly selected on $[-1, 1]$, the learning facto is adjusted adaptively on [0.25,1], which can better perform optimization search. A non-linear adjustment strategy is proposed. With the update of the particle swarm, the learning factor c_1 decreases from 1 to 0.25, and c_2 increases from 0.25 to 1. In the early stage of particles flight search, the larger c_1 and the smallerc_2 make the particles fly to the global optimal solution better; in the later stage of particles flight search, the smaller c_1 and the larger c_2 can make the particles fly in

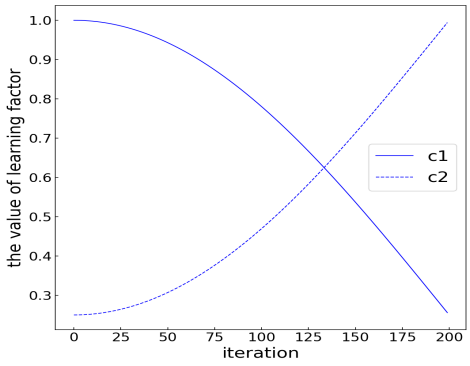

FIGURE 4. The value of the random inertia weight for 200 iterations.

the local optimal direction, which is beneficial to accelerate the convergence of the particles. The learning factor adjustment strategy is as follows.

$$c_1 = 0.75 \times \cos\left(0.5 \times \pi \times \frac{k}{K}\right) + 0.25, \tag{17}$$

$$c_2 = -0.75 \times \cos\left(0.5 \times \pi \times \frac{k}{K}\right) + 1. \tag{18}$$

Figure 4 shows the values of c_1, c_2 during 200 iterations.

3) A RENEWABLE ACCESS STRATEGY

In the process of particle flight optimization, it is inevitable that some out-of-boundary particles will be generated. These particles are beyond the monitoring area. There are usually three treatment methods: the first is to replace the out-of-boundary particles with boundary particles for adjustment, that is, ''absorption boundary''; the speed of the particles in this dimension is unchanged, and the direction is reversed, that is, the ''reflection boundary''; the third is to eliminate the particles and neither participate in the fitness value calculation nor update, that is, the ''invisible boundary''. Paper [42] indicated that the process of particle renewal will occur the disappearance of population diversity.

In this paper, a renewable access strategy is used to adjust the out-of-boundary particles, so that the out-of-boundary particles are randomly generated within flight range, and the more population is generated, making it more universal and diverse. The formula is as follows.

$$x_{i,d}^{k} = x_{min} + h \times (x_{max} - x_{min}). \tag{19}$$

where h is a uniform random number between (0,1), x_{max} a and x_{min} are the maximum and minimum ranges of particles that can fly. According to this update formula, the regenerated particles will be located in the optimized space, and the probability of each space position is equal.

4) PARTICLE POSITION UPDATE ADJUSTMENT FORMULA

The velocity update formula is as follows.

$$v_{i,d}^{k+1} = wv_{i,d}^{k} + c_1 r_1 \left(pb_{i,d}^{k} - x_{i,d}^{k}\right) + c_2 r_2 \left(gb^{k} - x_{i,d}^{k}\right) \tag{20}$$

Algorithm 1 Framework of MBPS

```
Initialize all individuals' positions within the search space;
for each individual do
Calculate fitness value;
 Update parameters according to formula (16), (17) and (18;
 Update the position of the out-of-bounds particle in search space according to the formula (19;
 Update the speed and position of particles according to the formula (20) and (21;
  if fitness > e the
    if num < maxgen the
Repeat the above process;
     else
       End;
   end if
  end fo
```

From the particle position update formula (2), it can be seen that in the basic PSO, the particle position update is only related to the current position and flight speed. When the particle is in a poor position and the flight speed is not ideal, the updated particle have poor fitness, so that the convergence of the group is not ideal. Based on this, introducing the global optimum into the position update formula can randomly guide particles to fly to the global optimal position, increasing the possibility of particles flying to a better position. The new position update strategy is as follows.

$$x_{i,d}^{k+1} = t \times (\alpha \times gb^{k} + v_{i,d}^{k}) + (1 - t) \times \left(x_{i,d}^{k} + v_{i,d}^{k}\right). \tag{21}$$

where t = 0.5, α is a uniform random number between (0,1). Set the value of t to 0.5 to balance the influence of the current position and the best position on the updated particle position, thereby achieving a better balance between global optimization and local optimization.

B. IDP REALIZATION PROCESS

Framework of MBPSO is described as Algorithm 1: Algorithm calculation error e is the smallest error the algorithm wants to achieve; The current iteration number is num; The maximum number of iterations is *maxgen*.

It is the flowchart of IDP as Figure 5. The steps to use this model to predict unknown data are as follows

1) Collect raw data, normalize the data, and divide the data into training set and test set. The training set is used for learning and the test set is used for prediction;

2) The penalty factor C and the kernel parameter σ of the LSSVM model are taken as the positions of the individuals in the MBPSO population. Initialize the population. Individuals in the population correspond to a set of two-dimensional coordinates C and σ;

3) Set the initial parameters of MBPSO;

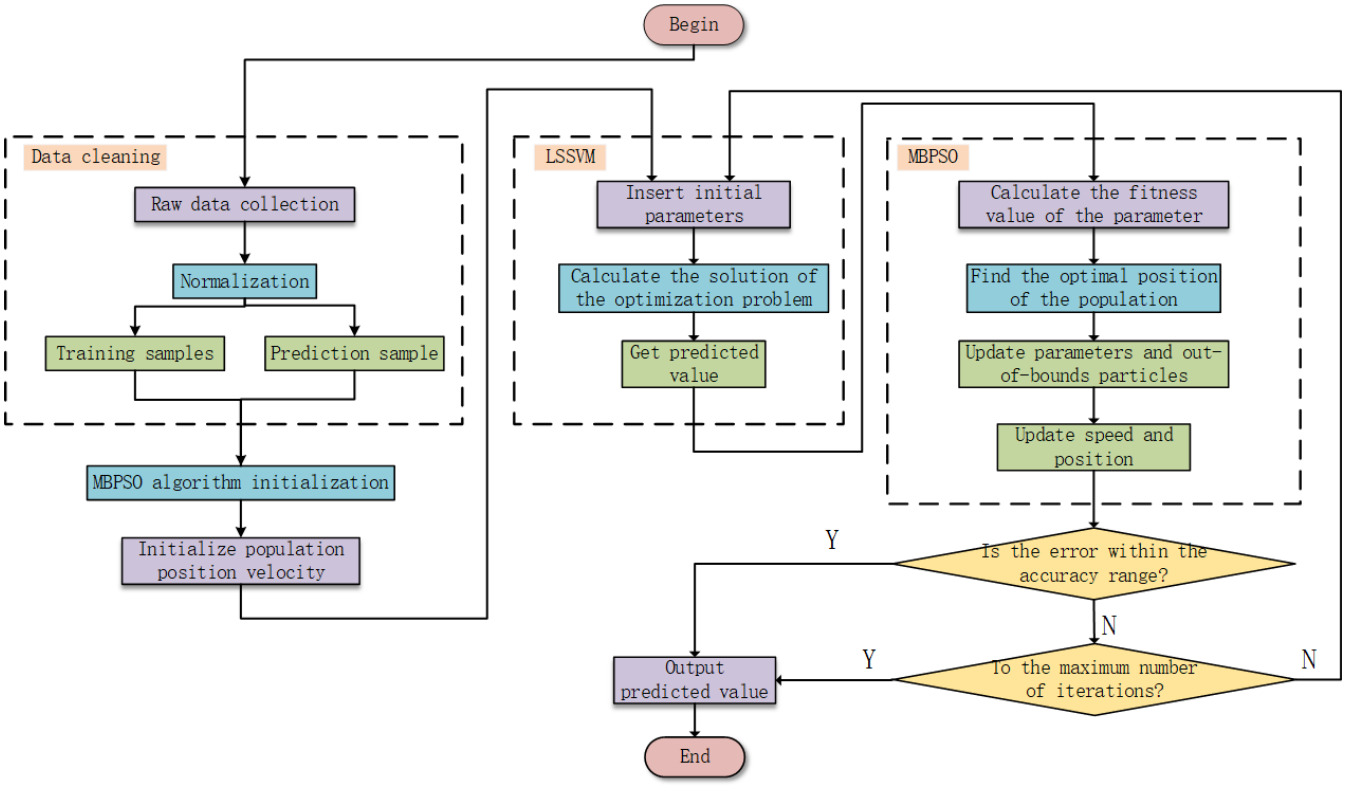

FIGURE 5. The flowchart of IDP.

4) Calculate the predicted value and compare it with the real value.

5) Calculate the fitness value of MBPSO corresponding to the penalty factor C and the kernel parameter σ;

6) Update the parameters of MBPSO.

7) Update the random position of the out-of-bounds particle optimization range.

8) Update particle velocity and position.

9) Seek the optimal position of the updated population, and apply the parameters corresponding to the optimal position to the prediction calculation of LSSVM.

10) Update the fitness values of all individuals in the population.

11) Compare the fitness value of the current population with the previous one, and perform individual better processing. That is, only update individuals with better fitness.

12) Repeat steps 6-11.

13) If the desired error value is reached, or the maximum number of iterations is reached, the update is stopped.

14) Save the last trained model as an IDP for prediction of unknown data.

Intelligent Data Prediction (IDP) uses MBPSO to optimize the parameters of LSSVM as the learning process. The two parameters to be optimized by LSSVM are the optimization goals of MBPSO. Each time MBPSO performs an optimization process, LSSVM will learn from the known data and calculate a prediction. The error MSE between the predicted value and the true value is the optimization target of MBPSO. After many trainings, the model can get the desired parameters. Save the trained LSSVM model named IDP. IDP can be used to predict unknown data. MBPSO and LSSVM collaborate for smart service.

Algorithm 2 Framework of ID

```
The penalty factor C and the nuclear parameter σ are
regarded as individuals in the MBPSO population,
which are the optimization goals of MBPSO;
for each individual do
Perform an LSSVM calculation;
  Calculate fitness value;
Better particle replacement treatment;
if meet the optimization conditions then
 End;
else
 Repeat steps 2-5;
end if
```

Framework of IDP is described as Algorithm 2.

V. PERFORMANCE ANALYSIS

A. VERIFICATION OF THE PROPOSED ALGORITHM

In order to verify the superiority of the optimization performance of MBPSO, a variety of PSO are selected for comparison. They are: the basic PSO, the algorithm for improving

TABLE 1. Parameter setting of six algorithms.

Algorithms	w	c_1	$c2$	m	D	K
basic PSO	0.9~0.4	2	2	50	30	100
MPSO	1~0	-	-	50	30	100
IPSO1	-	2.5	2.5	50	30	100
IPSO2	0.9~0.4	2.5~0.5	0.5~2.5	50	30	100
R-dPSO	[0,1]	1~0.25	0.25~1	50	30	100
MBPSO	[−1,1]	1~0.25	0.25~1	50	30	100

TABLE 2. Formulas of 9 functions.

function	formula		domain	D
Ackley	$f_1(x) = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n}x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos 2\pi x_i\right) + 20 + e$	(23)	[-32,32]	30
Rosenbrock	$f_2(x) = \sum_{i=1}^{n-1}[100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	(24)	[-30,30]	30
Sum of Different Power	$f_3(x) = \sum_{i=1}^{n}\lvert x_i\rvert^{(i+1)}$	(25)	[-1,1]	30
Axis parallel Hyperelliposid	$f_4(x) = \sum_{i=1}^{n} i x_i^2$	(26)	[-5.12,5.12]	30
Sphere	$f_5(x) = \sum_{i=1}^{n} x_i^2$	(27)	[-100,100]	30
Step	$f_6(x) = \sum_{i=1}^{n}(\lvert x_i + 0.5\rvert)^2$	(28)	[-100,100]	30
Griewank	$f_7(x) = \frac{1}{4000}\sum_{i=1}^{n}x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	(29)	[-600,600]	30
Schwefel's 2.22	$f_8(x) = \sum_{i=1}^{n}\lvert x_i\rvert + \prod_{i=1}^{n}\lvert x_i\rvert$	(30)	[-10,10]	30
Rastrigin	$f_9(\mathrm{x}) = \sum_{i=1}^{n}[x_i^2 - 10\cos(2\pi x_i) + 10]$	(31)	[-5.12,5.12]	30

the inertia weight and the speed update method named as MPSO [32], the algorithm for improving the speed update method named as IPSO1 [33], improved inertial weight and adaptive learning factor algorithm named as IPSO2 [34] and random-driven global PSO named as R-dPSO [35]. Use nine test functions to test the performance of these six algorithms. The parameter settings of the six algorithms are shown in Table 1. These nine functions are as Table 2.

In this paper, *fitness* is Mean Square Error (MSE) between the true value and the predicted value, and the calculation formula is as follows.

$$fitness = MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i^{'} - y_i)^2. \quad (22)$$

Where y_i is the true value, and y_i' is the predicted value.

The optimization results of the nine functions for the best fitness are shown in Figure 6, Figure 7, and Figure 8, respectively. The advantages and disadvantages of the six algorithms are ranked according to the best fitness. If the best fitness is the same, the average fitness is compared for ranking. The smaller the fitness value, the better performance in optimization and the higher the ranking. The best fitness, average fitness and ranking of nine functions are shown in Table 3, Table 4, and Table 5 for six algorithms.

These nine functions can test the optimization ability and speed of the algorithm. According to the above three optimization result figures and three optimization result tables,

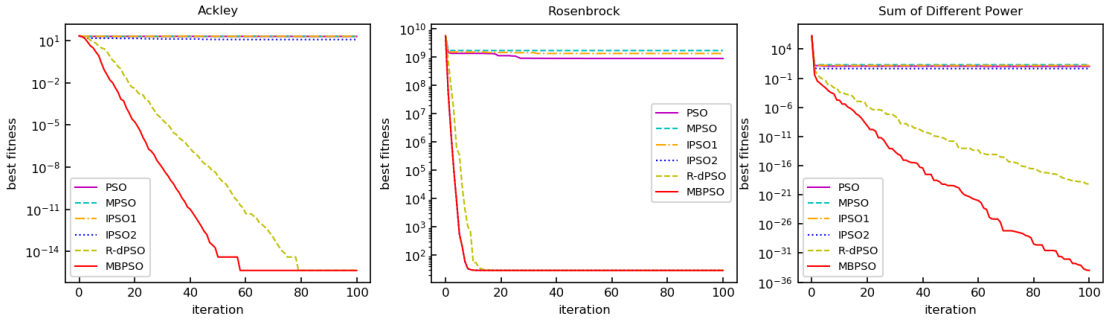

FIGURE 6. The change of best fitness value under f_1, f_2 and f_3.

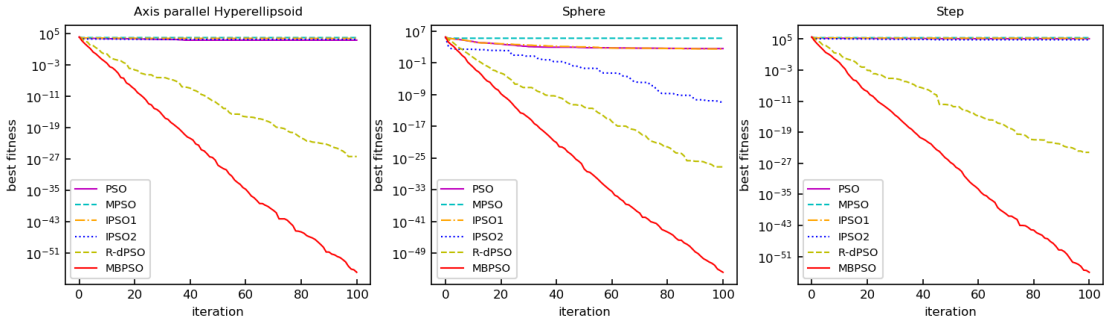

FIGURE 7. The change of best fitness value under f_4, f_5 and f_6.

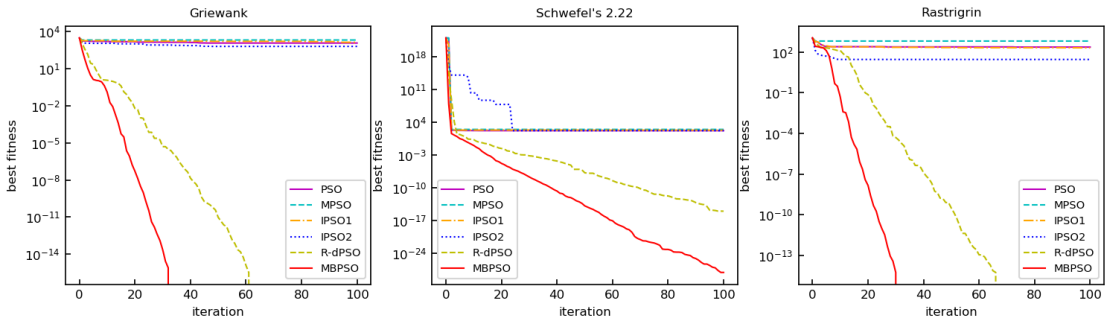

FIGURE 8. The change of best fitness value under f_7, f_8 and f_9.

TABLE 3. Result comparison on f_1, f_2 and f_3.

	Algorithm	Best	Mean	Rank	Algorithm	Best	Mean	Rank
f_1	basic PSO	1.98E+01	1.99E+01	4	IPSO2	1.18E+01	1.28E+01	3
	MPSO	2.00E+01	2.00E+01	5	R-dPSO	4.44E-16	1.15E+00	2
	IPSO1	2.00E+01	2.00E+01	5	MBPSO	4.44E-16	7.20E-01	**1**
f_2	basic PSO	8.96E+08	1.04E+09	4	IPSO2	3.71E+08	5.52E+08	3
	MPSO	1.71E+09	1.75E+09	6	R-dPSO	2.87E+01	6.41E+01	2
	IPSO1	1.34E+09	1.43E+09	5	MBPSO	2.87E+01	5.62E+07	**1**
f_3	basic PSO	1.10E+01	1.80E+04	4	IPSO2	4.51E+00	1.80E+04	3
	MPSO	2.05E+01	1.80E+04	6	R-dPSO	7.30E-20	1.80E+04	2
	IPSO1	1.52E+01	1.80E+04	5	MBPSO	1.06E-34	1.80E+04	**1**

it can be seen that the performance of MBPSO is far better than the basic PSO, MPSO, IPSO1, IPSO2. The four algorithms of basic PSO, MPSO, IPSO1, IPSO2 are easy to fall into the local optimal value, and the population diversity decreases with the increase of the number of iterations. Therefore, the iteration process stagnates and the

TABLE 4. Result comparison on f_4, f_5 and f_6.

	Algorithm	Best	Mean	Rank	Algorithm	Best	Mean	Rank
f_4	basic PSO	1.86E+03	2.75E+03	3	IPSO2	2.86E+03	3.12E+03	4
	MPSO	8.21E+03	8.24E+03	6	R-dPSO	4.41E-27	1.70E+02	2
	IPSO1	5.18E+03	5.56E+03	5	MBPSO	1.42E-56	1.28E+02	1
f_5	basic PSO	4.22E+02	1.31E+04	5	IPSO2	1.41E-11	3.40E+03	3
	MPSO	1.74E+05	1.74E+05	6	R-dPSO	5.68E-28	4.66E+03	2
	IPSO1	3.46E+02	1.36E+04	4	MBPSO	1.38E-54	3.59E+03	1
f_6	basic PSO	1.30E+05	1.45E+05	4	IPSO2	6.91E+04	8.64E+04	3
	MPSO	2.29E+05	2.29E+05	6	R-dPSO	6.61E-25	5.14E+03	2
	IPSO1	1.49E+05	1.81E+05	5	MBPSO	8.49E-56	3.77E+03	1

TABLE 5. Result comparison on f_7, f_8 and f_9.

	Algorithm	Best	Mean	Rank	Algorithm	Best	Mean	Rank
f_7	basic PSO	1.17E+03	1.31E+03	4	IPSO2	6.23E+02	7.79E+02	3
	MPSO	2.06E+03	2.07E+03	6	R-dPSO	0.00E+00	4.64E+01	2
	IPSO1	1.35E+03	1.63E+03	5	MBPSO	0.00E+00	3.40E+01	1
f_8	basic PSO	1.50E+02	1.67E+20	4	IPSO2	1.50E+02	8.84E+19	3
	MPSO	2.76E+02	1.77E+20	6	R-dPSO	1.00E-15	8.84E+19	2
	IPSO1	1.82E+02	9.20E+19	5	MBPSO	8.70E-29	8.84E+19	1
f_9	basic PSO	2.37E+02	2.61E+02	5	IPSO2	2.90E+01	4.24E+01	3
	MPSO	6.60E+02	6.65E+02	6	R-dPSO	0.00E+00	3.50E+01	2
	IPSO1	2.10E+02	2.44E+02	4	MBPSO	0.00E+00	2.24E+02	1

TABLE 6. Average optimal value sort.

z	basic PSO	MPSO	IPSO1	IPSO2	R-dPSO	MBPSO
Sum of rank	37	53	43	17	18	9
Final rank	4	6	5	3	2	1

optimization performance is poor. R-dPSO and MBPSO increase the abundance of the population as much as possible, and expand the optimization space of the particles to a certain extent. In addition to superior optimization potential, MBPSO is slightly better than R-dPSO in convergence speed and convergence accuracy.

Table 6 calculates the sum of rank and final rank about the performance of six PSO algorithms in nine test function results. The optimal value ranking is an important evaluation index for algorithm optimization performance. MBPSO ranks first under the test of each function, and has the best global search performance for nine functions.

B. EXAMPLE OF IDP PREDICTION MODEL

In China, rice and vegetables are the two very important agricultural products. Soil plays a central role in food safety by supplying possible components at the root of the food chain [43]. More than 60% of China's population eat rice, and rice safety is the top priority of food security. If you eat rice and vegetables contaminated with metal Cr, it will increase the incidence of cancer [44]. In 2014, Norton *et al.* reported that white rice from China showed significantly higher Pb concentrations ($0.064\mu gg^{-1}, n = 88$) than that from other countries ($0.004 - 0.033\mu gg^{-1}, n = 572$) [45]. In summary, the prediction of the content of heavy metals Cr and Pb in soil is of great significance to the research on the safety of agricultural products of rice and vegetables.

In order to verify the feasibility and superiority of IDP, two datasets are selected for experiments. The first dataset comes from the farmland soil heavy metal dataset of Dongxihu District, Wuhan City, which is dataset1, and the second dataset comes from the farmland soil heavy metal dataset of Hannan District, Wuhan City, which is dataset2. Cr and Pb in the two datasets are used as prediction objects. The longitude, latitude, sampling depth, and crop type codes of the sample are used as the input variables of the model, and the heavy metal content is used as the output variable of the model. The first group has 82 sets of data, of which 62 groups are used as training data, and 20 are used as testing data. The second group has 53 sets of data, of which 40 groups are used as training data and 13 are used as testing data. Among them, according to the different types of crops, the testing

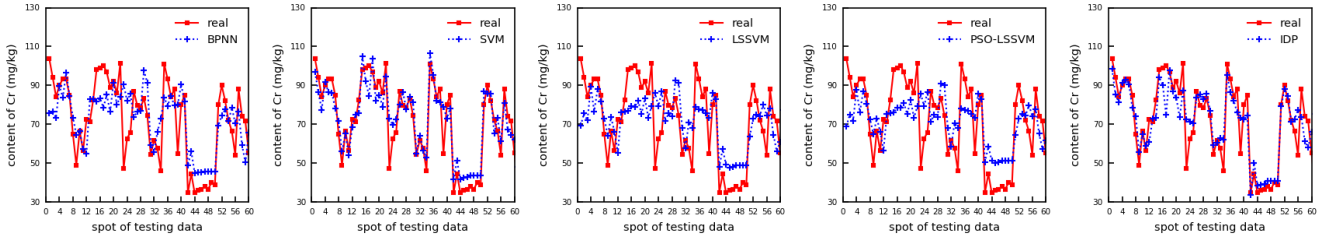

FIGURE 9. Metal Cr prediction results of BPNN, SVM, LSSVM, PSO-LSSVM and IDP in training data of dataset 1.

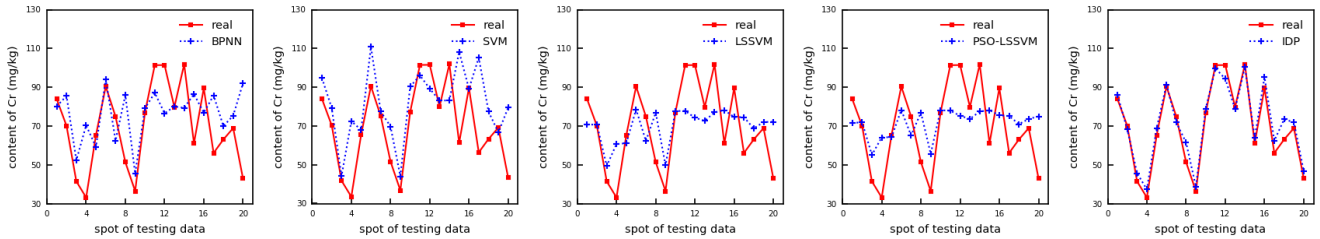

FIGURE 10. Metal Cr prediction results of BPNN, SVM, LSSVM, PSO-LSSVM and IDP in testing data of dataset 1.

TABLE 7. Parameter settings.

Model	Parameter
BPNN	learning rate=0.01, iteration=50
SVM	C=25,σ=10
LSSVM	C=25,σ=10
PSO-LSSVM	M=30, D=2, iteration=50, $C \in (0,50)$,$\sigma \in (0,20)$
IDP	M=30, D=2, iteration=50,$C \in (0,50)$,$\sigma \in (0,20)$

data is selected uniformly from the dataset from low to high heavy metal content, and the unselected data is used as the training data. Under the two datasets, first train the model by training data, and then use the trained model as IDP to predict the heavy metal content of the testing data. Before the experiment, the experimental data should be normalized first, and the maximum and minimum values are selected as the normalization method.

At present, common and proven predictive models for predicting the content of heavy metals in soil are BPNN [46] and SVM [47]. Choose these two models and LSSVM, PSO-LSSVM as the comparative experimental model of IDP. To ensure the fairness of the experiment, all experiments are carried out in the same environment. The SVM core function is the RBF function. Except for SVM and LSSVM, others the number of iterations of the model is 50. The iterative process is the learning and training of the model. The number of Neural Network nodes in the input layer, hidden layer, and output layer of BPNN are 4, 8, 1, and the learning rate of Network is 0.01. SVM and LSSVM use the same parameter. PSO-LSSVM and IDP use the same initial population to optimize model parameters. The parameter settings of the five models are shown in Table 7.

The relevant experimental results of the five models for the heavy metals Cr and Pb in data1 are given. The simulation fitting results of the training data are shown in Figure 9 and Figure 11 respectively. The simulation prediction results of the testing data are shown in Figure 10 and Figure 12 respectively. The error of the related experimental results is shown in Figure 13. The error is the absolute value of the difference between the real value and the predicted value.

Observing Figure 9, for the fitting of the heavy metal Cr training data, we can see that the fitting result of SVM and IDP is better than the other three models, and the overall fitting effect of SVM is better than that of IDP. It can be seen in combination with Figure10 that the prediction result of IDP on the testing set is significantly better than SVM. Although SVM has a good fitting result on the training data, the generalization ability of the trained model is poorer than IDP, so the prediction error of the testing data is significantly higher than IDP. LSSVM and PSO-LSSVM have similarity to the experimental results of training data and testing data, indicating optimized parameters of PSO-LSSVM are similar to the initial parameters of LSSVM, or have similar effects on the data prediction.

Observing Figure 11 and Figure 12, we can see that IDP fitting and prediction results are excellent. Although the fitting result of SVM training data is relatively good, its prediction result on the testing set is not ideal, indicating that the generalization performance of SVM is not ideal. In comparison, LSSVM has more stable prediction performance on the training data and testing data. BPNN is due to its own limitations, and the simulation results of the training data and testing data are not ideal.

Observing the error of the two heavy metal training data and testing data of dataset1 in Figure 13, we can see that, overall, IDP has the highest prediction accuracy.

The relevant experimental results of the five models on the heavy metals Cr and Pb in dataset2 is given. The fitting results of the training data are shown in Figure 14 and Figure 16, respectively. The prediction results of the testing data are

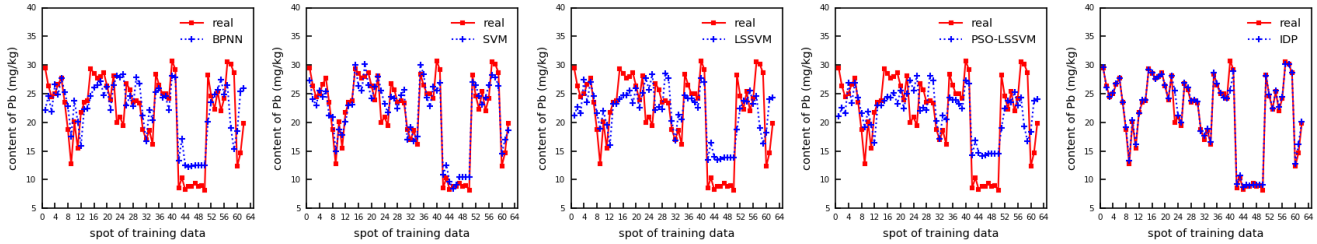

FIGURE 11. Metal Pb prediction results of BPNN, SVM, LSSVM, PSO-LSSVM and IDP in training data of dataset 1.

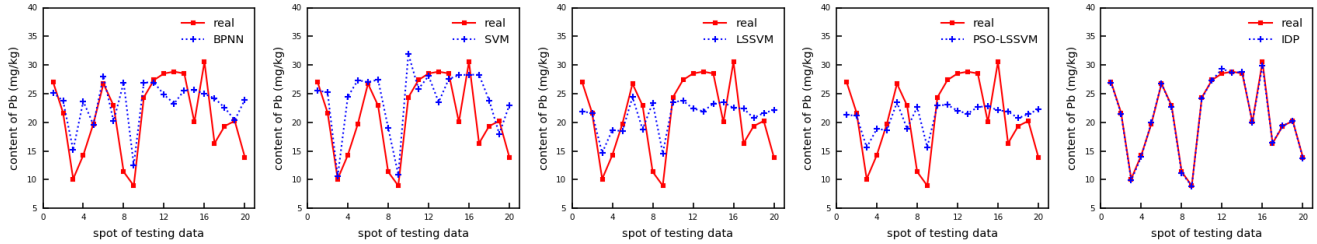

FIGURE 12. Metal Pb prediction results of BPNN, SVM, LSSVM, PSO-LSSVM and IDP in training data of dataset 1.

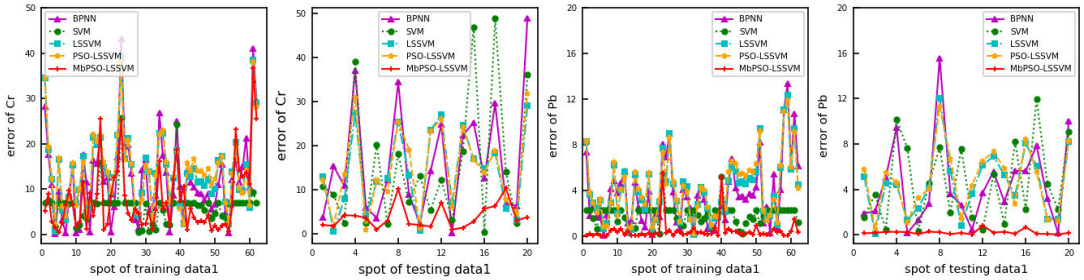

FIGURE 13. The absolute value of the error between the predicted value and the true value of dataset1.

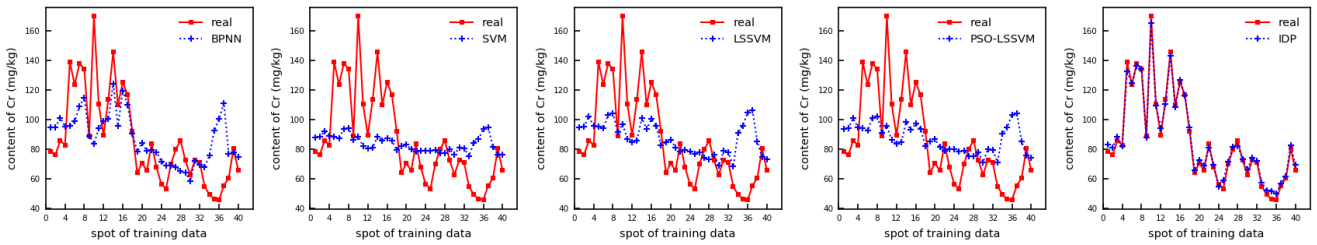

FIGURE 14. Metal Cr prediction results of BPNN, SVM, LSSVM, PSO-LSSVM and IDP in training data of dataset 2.

shown in Figure 15 and Figure 17. The error of the related experimental results is shown in Figure 18.

Observing Figure 14 and Figure 15, we can see that in addition to IDP, the other four models are not satisfactory in fitting and predicting the content of Cr in data2. The reason for BPNN may be the insufficient performance of the model itself. SVM, LSSVM and PSO-LSSVM are affected by model parameters. The fitting trend of each spot of the three models is similar, and the error of each spot is similar. The performance of parameters after learning and training of PSO-LSSVM is similar to that of the initial parameters artificially set by LSSVM. The results prove that PSO is not ideal about parameter optimization of LSSVM, and it is easy to fall into local optimal values due to itself limitation. The fitting and prediction results of IDP are relatively ideal, which proves that IDP not only has excellent parameter optimization performance, but also has strong generalization ability and good prediction effect.

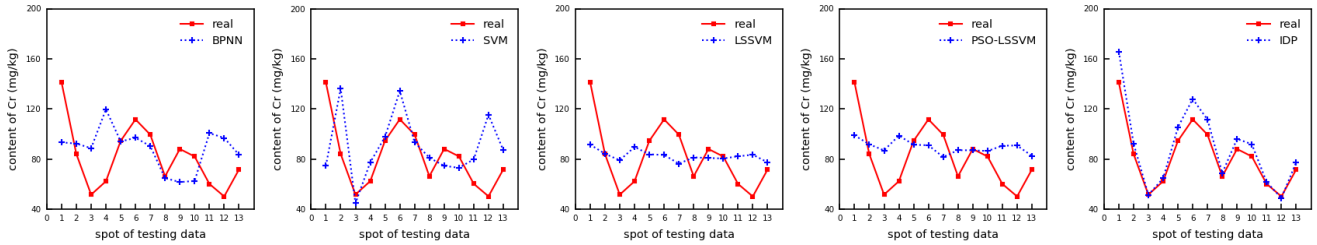

FIGURE 15. Metal Cr prediction results of BPNN, SVM, LSSVM, PSO-LSSVM and IDP in testing data of dataset 2.

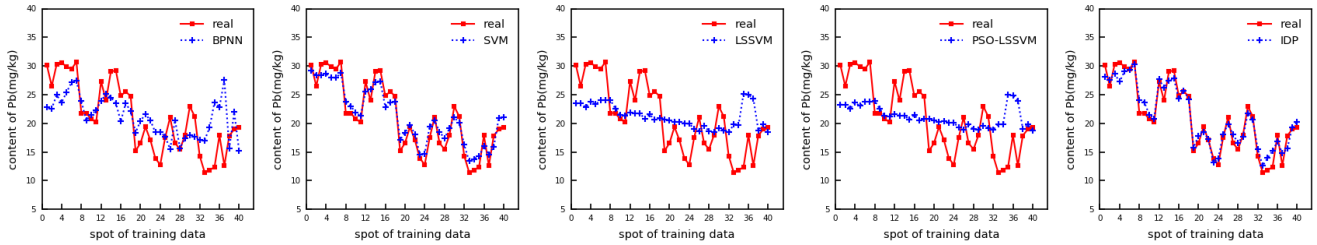

FIGURE 16. Metal Pb prediction results of BPNN, SVM, LSSVM, PSO-LSSVM and IDP in training data of dataset 2.

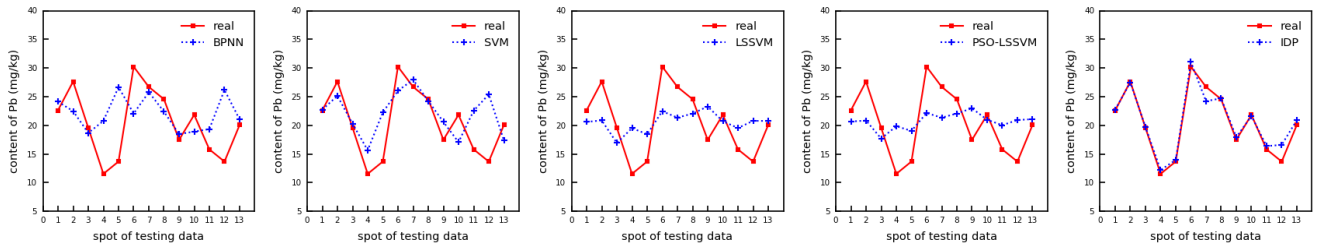

FIGURE 17. Metal Pb prediction results of BPNN, SVM, LSSVM, PSO-LSSVM and IDP in testing data of dataset 2.

Observing Figure 16 and Figure 17, we can see that the training fitting results of SVM and IDP are relatively ideal, but the predicted value of SVM on the testing data is quite different from the true value. This is the reason for the unstable generalization of SVM. The fitting and prediction results of LSSVM and PSO-LSSVM are relatively stable. The generalization performance of the model mainly depends on the quality of the trained model. The error of BPNN prediction is large, and the prediction performance of the model is average. Observing the error of the simulation results of the two heavy metal training data and testing data of data2, Figure 18 shows that, overall, the IDP model has the highest prediction accuracy.

Root Mean Square Error (RMSE), Mean Absolute Error (MAE) and Mean Absolute Percentage Error (MAPE) are three groups of error values as the evaluation index of the data prediction performance. The smaller the evaluation index value is, the higher the corresponding model have data prediction accuracy. The three sets of error formulas are as follows.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i' - y_i)^2}, \tag{33}$$

$$MAE = \frac{1}{n}\left(\sum_{i=1}^{n}\left|y_i' - y_i\right|\right), \tag{34}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i' - y_i|}{y_i} \times 100\%. \tag{35}$$

Wherey_i is the true value of the metal content at each spot, and y_i' is the predicted value of the metal at each spot.

Table 8 and Table 9 are the three sets of error values for fitting and prediction of dataset1 and dataset2 about the heavy metal Cr and Pb. The smaller the error value, the better the prediction performance of the model. The prediction errors of the two models of LSSVM and PSO-LSSVM are relatively close, indicating that the model parameters of PSO-LSSVM and LSSVM have similar effects on the prediction performance. The fitting and prediction performance of BPNN is average. In the simulation experiment of dataset1, the fitting error of SVM to training set is smaller than IDP, but the prediction error of the testing set is larger than IDP, indicating that the generalization performance of SVM is not ideal. From the overall analysis of the data in the table, it can be seen that IDP has the best fitting effect on the training data, and its generalization ability is very stable, and the prediction error on the testing set is the smallest. On the whole, IDP has

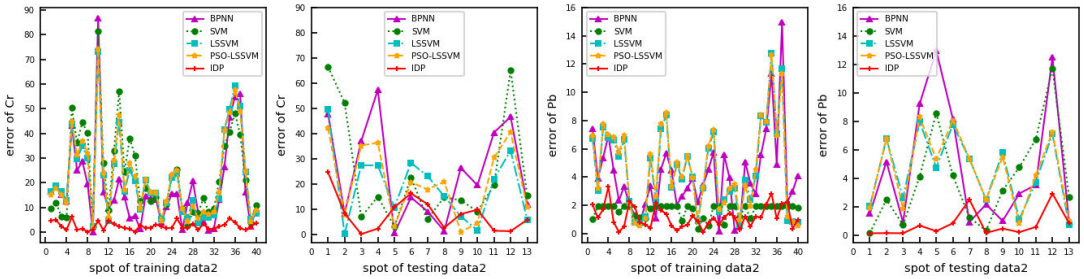

FIGURE 18. The absolute value of the error between the predicted value and the true value of dataset2.

TABLE 8. The calculated value of error indicators in dataset 1.

Dataset1	Training data	RMSE (mg/kg)	MAE (mg/kg)	MAPE (%)	Testing data	RMSE (mg/kg)	MAE (mg/kg)	MAPE (%)
Cr	BPNN	15.14	12.15	19.76	BPNN	20.84	16.29	29.42
	SVM	**7.99**	**6.90**	**10.89**	SVM	21.62	15.67	28.12
	LSSVM	15.85	13.70	22.00	LSSVM	17.01	14.39	24.06
	PSO-LSSVM	17.81	14.03	22.72	PSO-LSSVM	16.08	15.08	26.16
	IDP	10.55	7.56	12.17	IDP	**4.71**	**3.86**	**6.46**
Pb	BPNN	4.60	3.63	20.73	BPNN	5.84	4.37	28.31
	SVM	2.09	1.89	9.77	SVM	5.80	4.59	26.40
	LSSVM	4.78	3.97	22.83	LSSVM	5.39	4.56	26.70
	PSO-LSSVM	4.87	4.11	24.09	PSO-LSSVM	5.56	4.79	27.95
	IDP	**1.07**	**0.45**	**2.49**	IDP	**0.08**	**0.21**	**1.03**

TABLE 9. The calculated value of error indicators in dataset 2.

Dataset2	Training data	RMSE (mg/kg)	MAE (mg/kg)	MAPE (%)	Testing data	RMSE (mg/kg)	MAE (mg/kg)	MAPE (%)
Cr	BPNN	24.93	17.88	23.47	BPNN	30.82	24.70	35.59
	SVM	28.80	22.81	27.82	SVM	32.02	23.84	31.43
	LSSVM	27.31	21.67	28.09	LSSVM	23.71	19.37	25.81
	PSO-LSSVM	27.76	22.29	28.71	PSO-LSSVM	25.29	20.89	30.01
	IDP	**2.94**	**2.51**	**3.39**	IDP	**10.43**	**7.97**	**8.27**
Pb	BPNN	4.93	4.05	22.43	BPNN	6.41	4.76	29.37
	SVM	1.70	1.63	8.52	SVM	5.11	3.91	24.08
	LSSVM	5.44	4.53	24.67	LSSVM	5.10	4.48	24.94
	PSO-LSSVM	5.51	4.61	25.04	PSO-LSSVM	5.22	4.45	25.46
	IDP	**1.39**	**1.17**	**6.22**	IDP	**1.15**	**0.76**	**4.24**

the best performance in predicting soil heavy metal content for smart service.

In addition to the above comparison, Table 10 also provides the training time required for each group of experiments. It can be seen from the data in the table that the training time of SVM and BPNN is similar, but SVM only performs one non-linear calculation, while BPNN has 50 parameter optimization processes. It can be seen that the non-linear calculation of SVM is better than that of BPNN. It may be because the calculation process of SVM is more complicated. The running time of LSSVM is much longer than that of BPNN and SVM. The reason is that there are complex kernel calculations inside LSSVM. The running time of POS-LSSVM and IDP is much longer than the previous models, and the training time required for the two models is similar. Both models are optimized by the process of parameter optimization, and each optimization requires complex kernel calculations, so the required running time is longer. Combining the analysis of Table 8 and Table 9, it can be seen that the optimization process of IDP is more effective, the final fitting and prediction result of model training has smaller errors, and the model performs stronger generalization performance when tested.

TABLE 10. The runtime of five models for training.

Data	Cr	Runtime(s)	Pb	Runtime(s)	Data	Cr	Runtime(s)	Pb	Runtime(s)
	BPNN	0.11	BPNN	0.09		BPNN	0.02	BPNN	0.04
	SVM	0.17	SVM	0.18		SVM	0.13	SVM	0.17
data1	LSSVM	0.49	LSSVM	0.39	data2	LSSVM	0.19	LSSVM	0.20
	PSO-LSSVM	150.83	PSO-LSSVM	117.59		PSO-LSSVM	48.35	PSO-LSSVM	62.05
	IDP	106.23	IDP	110.72		IDP	47.93	IDP	53.26

TABLE 11. Model final parameters of LSSVM, PSO-LSSVM, IDP.

Data	Cr	C	σ	num	Pb	C	σ	num
	LSSVM	25.00	10.00	-	LSSVM	25.00	10.00	-
dataset1	PSO-LSSVM	27.44	14.30	1	PSO-LSSVM	27.44	14.30	1
	IDP	16.51	1.29	50	IDP	30.02	0.35	47
	LSSVM	25.00	10.00	-	LSSVM	25.00	10.00	-
dataset2	PSO-LSSVM	27.44	14.30	1	PSO-LSSVM	27.44	14.30	1
	IDP	17.36	0.50	50	IDP	29.32	1.26	46

In Table 11, c, σ are the final fitting model parameters of LSSVM, PSO, and IDP, and num is the number of times required to find the best parameter. The parameters of LSSVM are obtained based on experience and there is no optimization process. The num of PSO-LSSVM are all 1, which means that the iteration process has stagnated after the model obtains parameters from 10 initial populations for the first time. The model has the same initial populations, so PSO-LSSVM optimized the same parameters for the four groups of experiments. IDP better avoids the shortcomings of PSO-LSSVM easy to fall into the local optimum. According to the different optimization processes of each group of experiments, the parameters of each model are finally different, and the generalization performance of IDP after being trained is better.

VI. CONCLUSION AND FUTURE WORK

In this paper, big data prediction methods are used as research objects to predict the content of metal in soil. IDP is proposed to fit and predict data through the collaboration of MBPSO and LSSVM for smart service. In order to verify the feasibility and superiority of IDP, this paper uses two datasets of farmland soil heavy metal as experimental objects. Through the prediction of the heavy metal content, the errors are compared to judge the performance of model learning, generalization and prediction. The experimental results of IDP comparing BP, SVM, LSSVM and PSO-LSSVM show: In dataset1, the prediction accuracy of Cr is increased by 22.9%, 21.66%, 17.60%, 19.70%, and the prediction accuracy of Pb is increased by 27.28%,25.37%, 25.67%, 26.92%, respectively; In dataset2, the prediction accuracy of Cr increased by 27.32%, 23.16%, 17.54%, 21.74%, and the prediction accuracy of Pb increased by 25.13%, 19.84%, 20.70%, 21.22 %, respectively.

Experimental results show that under the intelligent service model combining MBPSO and LSSVM, IDP learns the known data repeatedly through the optimization process of MBPSO. The training results can provide good parameters for the model. IDP accurately predicts data through complex core calculations in the model. The model saved by training has excellent prediction results for new knowledge.

Although IDP has high prediction accuracy in this study, there is still some supplementary work to be improved. In the later stage of optimization, although the results of the local search are still changing, it has a small impact on the IDP prediction accuracy. Whether it is necessary to optimize the later stage requires further discussion.

IDP has complex core calculations, so when the amount of data is very large, the requirements for corresponding computing equipment will be very high. The LSSVM inside IDP is sensitive to outliers. Even if there are a few outliers in the data used for experiments, the experimental error will be large.

In future work, for the model proposed in this article, we can consider continuing to improve the PSO to improve the optimization speed and accuracy of the model. For example, other intelligent algorithms can be combined to improve PSO. Or, by optimizing the structure of the IDP, the prediction time is reduced and the prediction error is reduced. For example, intelligently monitoring the number of iterations can not only ensure the optimization of the required parameters, but also avoid unnecessary optimization time. You can

also consider combining other swarm intelligence algorithms and machine learning algorithms to build models. You can also consider trying other data sets to expand the scope of model applications. In short, improving the generalization performance and prediction accuracy of prediction models is the focus of future work.

REFERENCES

[1] G. Bello-Orgaz, J. J. Jung, and D. Camacho, "Social big data: Recent achievements and new challenges," *Inf. Fusion*, vol. 28, pp. 45–59, Mar. 2016.

[2] A. R. Lima, A. J. Cannon, and W. W. Hsieh, "Nonlinear regression in environmental sciences using extreme learning machines: A comparative evaluation," *Environ. Model. Softw.*, vol. 73, pp. 175–188, Nov. 2015.

[3] J. Liao, L. Tang, G. Shao, X. Su, D. Chen, and T. Xu, "Incorporation of extended neighborhood mechanisms and its impact on urban land-use cellular automata simulations," *Environ. Model. Softw.*, vol. 75, pp. 163–175, Jan. 2016.

[4] N. Nidheesh, K. A. A. Nazeer, and P. M. Ameer, "An enhanced deterministic K-means clustering algorithm for cancer subtype prediction from gene expression data," *Comput. Biol. Med.*, vol. 91, pp. 213–221, Dec. 2017.

[5] C. Bockstaller, S. Beauchet, V. Manneville, B. Amiaud, and R. Botreau, "A tool to design fuzzy decision trees for sustainability assessment," *Environ. Model. Softw.*, vol. 97, pp. 130–144, Nov. 2017.

[6] B. J. Robson and A. Mousquès, "Can we predict citation counts of environmental modelling papers? Fourteen bibliographic and categorical variables predict less than 30% of the variability in citation counts," *Environ. Model. Softw.*, vol. 75, pp. 94–104, Jan. 2016.

[7] Y. Feng, Y. Yang, X. Huang, S. Mehrkanoon, and J. A. K. Suykens, "Robust support vector machines for classification with nonconvex and smooth losses," *Neural Comput.*, vol. 28, no. 6, pp. 1217–1247, 2016.

[8] Ş. Yalpır, "Enhancement of parcel valuation with adaptive artificial neural network modeling," *Artif. Intell. Rev.*, vol. 49, no. 3, pp. 393–405, Mar. 2018.

[9] X. Zhang, Z. Wang, M. Cheng, X. Wu, N. Zhan, and J. Xu, "Long-term ambient SO_2 concentration and its exposure risk across China inferred from OMI observations from 2005 to 2018," *Atmos. Res.*, vol. 247, Jan. 2020, Art. no. 105150.

[10] H. Yan, J. Zhang, N. Zhou, and M. Li, "Application of hybrid artificial intelligence model to predict coal strength alteration during CO_2 geological sequestration in coal seams," *Sci. Total Environ.*, vol. 711, Apr. 2020, Art. no. 135029.

[11] F. Ning, C. Yunxia, P. Yue, L. Weidi, and Z. Yongqing, "Prediction and evaluation of soil heavy metal concentration around coal-fired power plant based on BP neural network," *Environ. Sci. Technol.*, vol. 31, no. 2, pp. 52–56, 2018.

[12] S. Mehrjoo, A. Sarrafzadeh, and M. Mehrjoo, "Swarm intelligent compressive routing in wireless sensor networks," *Comput. Intell.*, vol. 31, no. 3, pp. 513–531, Aug. 2015.

[13] C. Karakuzu, F. Karakaya, and M. A. Çavuşlu, "FPGA implementation of neuro-fuzzy system with improved PSO learning," *Neural Netw.*, vol. 79, pp. 128–140, Jul. 2016.

[14] M. S. Rahman, M. K. Rahman, M. Kaykobad, and M. S. Rahman, "IsGPT: An optimized model to identify sub-golgi protein types using SVM and random forest based feature selection," *Artif. Intell. Med.*, vol. 84, pp. 90–100, Jan. 2018.

[15] K. K. Bhattacharjee and S. P. Sarmah, "Modified swarm intelligence based techniques for the knapsack problem," *Int. J. Speech Technol.*, vol. 46, no. 1, pp. 158–179, Jan. 2017.

[16] S. Mirjalili, "How effective is the grey wolf optimizer in training multi-layer perceptrons," *Int. J. Speech Technol.*, vol. 43, no. 1, pp. 150–161, Jul. 2015.

[17] B. Wang, W. Kong, H. Guan, and N. N. Xiong, "Air quality forecasting based on gated recurrent long short term memory model in Internet of Things," *IEEE Access*, vol. 7, pp. 69524–69534, 2019.

[18] C. Chen, N. N. Xiong, X. Guo, and J. Ren, "The system identification and prediction of the social earthquakes burst in human society," *IEEE Access*, vol. 8, pp. 103848–103859, 2020.

[19] Z. Wang, J. Huang, N. N. Xiong, X. Zhou, X. Lin, and T. L. Ward, "A robust vehicle detection scheme for intelligent traffic surveillance systems in smart cities," *IEEE Access*, vol. 8, pp. 139299–139312, 2020.

[20] E. Kita, M. Harada, and T. Mizuno, "Application of Bayesian network to stock price prediction," *Artif. Intell. Res.*, vol. 1, no. 2, p. 171, Sep. 2012.

[21] J. Li, B. Alvarez, J. Siwabessy, M. Tran, Z. Huang, R. Przeslawski, L. Radke, F. Howard, and S. Nichol, "Application of random forest, generalised linear model and their hybrid methods with geostatistical techniques to count data: Predicting sponge species richness," *Environ. Model. Softw.*, vol. 97, pp. 112–129, Nov. 2017.

[22] Z. Sai, C. Lu, S. Jiang, L. Shan, C. James, and N. N. Xiong, "Energy management optimization of open-pit mine solar photothermal-photoelectric membrane distillation using a support vector machine and a non-dominated genetic algorithm," *IEEE Access*, vol. 8, pp. 155766–155782, 2020.

[23] X. Song, X. Li, W. Tang, and W. Zhang, "A fusion strategy for reliable vehicle positioning utilizing RFID and in-vehicle sensors," *Inf. Fusion*, vol. 31, pp. 76–86, Sep. 2016.

[24] X. Li, W. Chen, C. Chan, B. Li, and X. Song, "Multi-sensor fusion methodology for enhanced land vehicle positioning," *Inf. Fusion*, vol. 46, pp. 51–62, Mar. 2019.

[25] C. Yu, Z. Xi, Y. Lu, K. Tao, and Z. Yi, "LSSVM-based color prediction for cotton fabrics with reactive pad-dry-pad-steam dyeing," *Chemometric Intell. Lab. Syst.*, vol. 199, Apr. 2020, Art. no. 103956.

[26] L. Wen, X.-M. Liang, Z.-Q. Long, and Z.-H. Li, "Parameters selection for LSSVM based on modified ant colony optimization in short-term load forecasting," *J. Central South Univ., Sci. Technol.*, vol. 42, no. 11, pp. 3408–3414, 2011.

[27] B. Zeng, J. Guo, W. Zhu, Z. Xiao, F. Yuan, and S. Huang, "A transformer fault diagnosis model based on hybrid grey wolf optimizer and LS-SVM," *Energies*, vol. 12, no. 21, p. 4170, Nov. 2019.

[28] H. Xu and G. Chen, "An intelligent fault identification method of rolling bearings based on LSSVM optimized by improved PSO," *Mech. Syst. Signal Process.*, vol. 35, nos. 1–2, pp. 167–175, Feb. 2013.

[29] A. Bemani, Q. Xiong, A. Baghban, S. Habibzadeh, A. H. Mohammadi, and M. H. Doranehgard, "Modeling of cetane number of biodiesel from fatty acid methyl ester (FAME) information using GA-, PSO-, and HGAPSO-LSSVM models," *Renew. Energy*, vol. 150, pp. 924–934, May 2020.

[30] U. Kirchmaier, S. Hawe, and K. Diepold, "Dynamical information fusion of heterogeneous sensors for 3D tracking using particle swarm optimization," *Inf. Fusion*, vol. 12, no. 4, pp. 275–283, Oct. 2011.

[31] L. Kang, R.-S. Chen, N. Xiong, Y.-C. Chen, Y.-X. Hu, and C.-M. Chen, "Selecting hyper-parameters of Gaussian process regression based on non-inertial particle swarm optimization in Internet of Things," *IEEE Access*, vol. 7, pp. 59504–59513, 2019.

[32] J. Fangfang and Z. Xin, "Solving the working space boundary of a planar parallel mechanism based on improved particle swarm optimization," *Mech. Transmiss.*, vol. 43, no. 12, pp. 28–33, 2019.

[33] S. Yu and Y. Ni, "An improved particle swarm optimization back propogation neural network algorithm for psychological stress identification," *Sci. Technol. Eng.*, vol. 20, no. 4, pp. 1467–1472, 2020.

[34] M. Yu, "Research on fault diagnosis method based in improved particle swarm optimization algorithm for SVM," *Mech. Eng. Autom.*, vol. 6, no. 6, pp. 14–15, 2020.

[35] Z. Hu, D. Z. Zou, and X. Zhang, "R-dPSO algorithm and its application in ATO control strategy," *Comput. Eng. Appl.*, vol. 54, no. 24, pp. 212–220, 2018.

[36] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proc. 6th Int. Symp. Micro Mach. Hum. Sci.*, Piscataway, NJ, USA, 1995, pp. 39–43.

[37] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *Proc. Int. Conf. Neural Netw.*, Perth, WA, Australia, May 1995, pp. 69–73.

[38] J. A. K. Suykens and J. Vandewalle, "Least squares support vector machine classifiers," *Neural Process. Lett.*, vol. 9, pp. 293–300, Jun. 1999.

[39] X.-L. Gu, M. Huang, and X. Liang, "A discrete particle swarm optimization algorithm with adaptive inertia weight for solving multiobjective flexible job-shop scheduling problem," *IEEE Access*, vol. 8, pp. 33125–33136, 2020.

[40] L. Zhang, Y. Tang, C. Hua, and X. Guan, "A new particle swarm optimization algorithm with adaptive inertia weight based on Bayesian techniques," *Appl. Soft Comput.*, vol. 28, pp. 138–149, Mar. 2015.

[41] Y. Lu, M. Liang, Z. Ye, and L. Cao, "Improved particle swarm optimization algorithm and its application in text feature selection," *Appl. Soft Comput.*, vol. 35, pp. 629–636, Oct. 2015.

[42] T. Wu, X. Chen, and Y. Yan, "Study of on-ramp PI controller based on dural group QPSO with different well centers algorithm," *Math. Problems Eng.*, vol. 2015, pp. 1–10, Jan. 2015.

[43] G. Tóth, T. Hermann, M. R. Da Silva, and L. Montanarella, "Heavy metals in agricultural soils of the European union with implications for food safety," *Environ. Int.*, vol. 88, pp. 299–309, Mar. 2016.
[44] J. R. Peralta-Videa, M. L. Lopez, M. Narayan, G. Saupe, and J. Gardea-Torresdey, "The biochemistry of environmental heavy metal uptake by plants: Implications for the food chain," *Int. J. Biochem. Cell Biol.*, vol. 41, nos. 8–9, pp. 1665–1677, Aug. 2009.
[45] G. J. Norton, P. N. Williams, E. E. Adomako, A. H. Price, Y. Zhu, F.-J. Zhao, S. McGrath, C. M. Deacon, A. Villada, A. Sommella, Y. Lu, L. Ming, P. M. C. S. De Silva, H. Brammer, T. Dasgupta, M. R. Islam, and A. A. Meharg, "Lead in rice: Analysis of baseline lead levels in market and field collected rice grains," *Sci. Total Environ.*, vols. 485–486, pp. 428–434, Jul. 2014.
[46] Z. Huang, Y. Ding, and J. Wang, "Special prediction of regional soil heavy metals content based on multiple model optimization," *J. Ecol. Rural Environ.*, no. 3, pp. 308–317, 2020.
[47] Z.-R. Yuan, L.-F. Wei, Y.-X. Zhang, M. Yu, and X.-R. Yan, "Hyperspectral inversion and analysis of heavy metal arsenic content in farmland soil based on optimizing CARS combined with PSO-SVM algorithm," *Spectrosc. Spectral Anal.*, vol. 40, no. 2, pp. 241–247, 2020.

FANG CHEN received the bachelor's degree from Huanggang Normal University, in 2018. She is currently pursuing the master's degree with the School of Mathematics and Computer Science, Wuhan Polytechnic University. Her research interests include artificial intelligence technology and its applications.

CONG ZHANG received the bachelor's degree in automation engineering from the Huazhong University of Science and Technology, in 1993, the master's degree in computer application technology from the Wuhan University of Technology, in 1999, and the Ph.D. degree in computer application technology from Wuhan University, in 2010. He is currently a Professor with the School of Mathematics and Computer Science, Wuhan Polytechnic University. His research interests include multimedia signal processing, multimedia communication system theory and application, pattern recognition, artificial intelligence, and big data technology in multimedia.

JUNJIE ZHANG received the B.S. degree in computer science from Wuhan Polytechnic University, China, in 2018, where he is currently pursuing the master's degree. His research interests include social learning, machine learning, recommender systems, and data mining.

WENQI CAO received the B.S. degree in computer science from Wuhan Polytechnic University, China, in 2018, where he is currently pursuing the master's degree. His research interests include artificial intelligence technology and its application.