Frontiers in Environmental Research
ISSN: 2960-026X
DOI: https://doi.org/10.61784/fer3030

INTELLIGENT PREDICTION OF WATER QUALITY PARAMETERS BASED ON IMPROVED PARTICLE SWARM OPTIMIZATION COUPLED WITH RANDOM FOREST MODEL

JinYi Luo
School of Mathematical Science, Chengdu University of Technology, Chengdu 610059, Sichuan, China.
Corresponding Email: ljinyi77@gmail.com

Abstract: In recent years, the problem of water pollution has become increasingly serious, and traditional water quality monitoring methods are difficult to meet the demand for high precision. Constructing an efficient and reliable water quality prediction model is of great significance for governance decisions. To this end, this paper proposes a random forest (IPSO-RF) water quality prediction model optimized based on the improved particle swarm algorithm. Firstly, for the problems of traditional particle swarm algorithm (PSO), such as easy premature convergence and insufficient global search ability, an improved particle swarm algorithm (IPSO) with nonlinear iteration incorporating inertia weights is proposed, and its optimization performance is verified based on typical test functions. Secondly, the IPSO algorithm is combined with random forest (RF) to optimize the key hyperparameters of RF (e.g., the number of decision trees, the minimum number of samples for node splitting, etc.) in order to improve the prediction accuracy and generalization ability of the model. Simulation experiments were carried out based on the water quality monitoring data of a watershed, and the results showed that compared with RF and standard PSO-RF and other models, the IPSO-RF model showed lower MSE in the prediction of key indicators such as dissolved oxygen, phosphorus content of water body and ammonia nitrogen amount of water body, which verified its superiority in the prediction of water quality. This study not only provides new ideas for the application of intelligent optimization algorithms in the field of water environment, but also provides more accurate technical support for water quality monitoring and pollution prevention and control of environmental protection departments.
Keywords: Random forest; IPSO-RF; Inertia weight; PSO

1 INTRODUCTION

As China's economy and society continue to develop rapidly, the problem of water supply and demand has become increasingly prominent, and the problem of water pollution has become one of the important factors limiting sustainable development. According to the Bulletin of China's Ecological and Environmental Conditions released by the Ministry of Ecology and Environment, about 20% of the surface water monitoring sections in the country will still fail to meet the Class III water quality standard in 2022. Traditional water quality monitoring methods have limitations in terms of high cost and poor timeliness, making it difficult to meet modern environmental governance needs. In this context, intelligent optimization algorithms have brought new technological breakthroughs in the field of water quality monitoring. And the first large-scale application of the intelligent early warning system for the cyanobacterial outbreak in Lake Taihu in 2016 marked the entry of China's water quality monitoring into the intelligent development stage. In recent years, the application of machine learning algorithms in water quality prediction has become more and more widespread, which greatly promotes the environmental protection department, water enterprises and scientific research institutions to pay more attention to the innovation and application of intelligent monitoring technology, and provides new technical support for water environment management.

At present, scholars both at home and abroad have been conducting research on the construction of water quality monitoring models and evaluating various water quality indicators. The research methods involve machine learning methods and artificial intelligence. Taher Rajaee et al. used methods including artificial neural network (ANN), genetic programming (GP), fuzzy logic (FL), support vector machine (SVM), hybrid neural-fuzzy (NF), hybrid ANN-ARIMA, and hybrid genetic algorithm-neural network (GA-NN) to model and predict river water quality parameters such as dissolved oxygen (DO) and suspended sediment [1]. Mosleh Hmoud Al-Adhaileh et al. used feedforward neural network (FFNN) and K-nearest neighbor for water quality classification, and proposed advanced artificial intelligence methods that can assist water treatment and management. The ANFIS model accurately predicted WQI [2]. Yingyi Chen et al. conducted extensive investigations and analyses on water quality prediction based on ANN from three aspects: feedforward, feedback, and hybrid architectures [3]. Lu et al. used extreme gradient boosting (XGBoost) and random forest (RF) to predict six water quality indicators such as water temperature, dissolved oxygen, pH value, specific conductance, turbidity, and fluorescent dissolved organic matter [4].Compared with the traditional water quality monitoring methods, machine learning techniques show better performance and prediction accuracy in constructing water quality prediction models; among the many machine learning algorithms, the random forest (RF) model derived from decision trees has been widely used in water quality classification and regression prediction due to its excellent generalization ability and robustness. However, it should be pointed out that traditional machine learning methods are often limited by parameter setting issues, such as the selection of kernel function and penalty coefficient

setting of support vector regression (SVR), which can significantly affect the model performance and easily lead to problems such as overfitting or underfitting. The emergence of population intelligent optimization algorithms, on the other hand, provides an effective technical way to solve the parameter optimization problems of traditional machine learning models.

In this paper, based on the systematic analysis of related research at home and abroad, an improved particle swarm algorithm (IPSO) incorporating inertia weight nonlinear iteration is proposed and applied to the optimization study of water quality prediction model. The research content of the whole paper mainly includes the following aspects: firstly, introducing the basic principles of random forest (RF) model; secondly, elaborating the basic principles of the standard particle swarm algorithm, proposing the improvement strategy based on the inertia weight nonlinear iteration for the problems such as easy to fall into the local optimization, and choosing the two test functions of Sphere and Rastrigin to verify the performance of the IPSO algorithm, through the comparison experiment with the PSO algorithm, and through the optimization of water quality forecasting model. The comparison experiment with PSO algorithm confirms the superiority of the improved algorithm, constructs the IPSO-RF integrated prediction model, applies it to the task of water quality prediction, and finally puts forward the targeted optimization suggestions for water quality monitoring based on the experimental results.

2 DESCRIPTION OF APPLICATION METHODS

2.1 Particle Swarm Algorithm

Particle swarm optimization algorithm (PSO) belongs to an important branch of population intelligence optimization algorithms, and together with ant colony algorithms and artificial fish swarm algorithms, it constitutes a research hotspot in the field of intelligent computing [5]. The algorithm was first proposed by Kennedy and Eberhart in 1995, and its design was inspired by the observation study of bird group foraging behavior. In nature, an efficient strategy for birds to search for food is to congregate to the area of the individuals in the group that are closest to the food source.The PSO algorithm is an optimization method developed to simulate this characteristic of intelligent behavior of biological groups.Each particle in the algorithm characterizes a possible solution to the problem and evaluates the degree of its superiority or inferiority by means of a fitness function. During the search process, the particles continuously adjust their speed and direction according to their own historical optimal position and the optimal position of the group, thus realizing intelligent exploration and optimization in the solution space. In order to calculate the fitness value corresponding to the position of each particle according to the objective function, it is assumed that in a D-dimensional search space, a population consisting of n particles $\mathrm{X} = (\mathrm{X}_1, \mathrm{X}_2, \ldots \mathrm{X}_\mathrm{n})$, where the i-th particle is denoted as a D-dimensional vector $X_i = (X_{i1}, X_{i2}, \ldots, X_{iD})^T$, represents the position of the ith particle in the D-dimensional search space, which is a potential solution to the problem. The velocity of the ith particle is $V_i = (V_{i1}, V_{i2}, \ldots, V_{iD})^T$, Its individual extremes are $P_i = (P_{i1}, P_{i2}, \ldots, P_{iD})^T$, And the population extremes of the population are $P_g = (P_{g1}, P_{g2}, \ldots, P_{gD})^T$.

During each iteration, the particle updates its velocity and position through the individual poles and the population poles, i.e：

$$V_{id}^{k+1} = \omega V_{id}^{k} + c_1 r_1 (P_{id}^{k} - X_{id}^{k}) + c_2 r_2 \left(P_{gd}^{k} - X_{id}^{k}\right) \tag{1}$$

$$X_{id}^{k+1} = X_{id}^{k} + V_{id}^{k+1} \tag{2}$$

where ω is the inertia weight; $\mathrm{d} = 1,2,3,\ldots \mathrm{D}; \mathrm{i} = 1,2,3,\ldots,\mathrm{n}$; k is the current iteration number; V_{id} is the particle velocity; and c_1 and c_2 is a non-negative constant called acceleration factor; and r_1 and r_2 is a random number distributed in the interval $[0，1]$. In order to prevent blind search of particles, it is generally recommended to limit the position and velocity to a certain interval as $[-X_{max}，X_{max}]$、$[-X_{max}，X_{max}]$.

Basic steps:

Step 1. Parameter setting:

General order $c_1 = 1.49445，c_2 = 1.49445$

Step 2. Population initialization:

Randomly initialize particle position and particle velocity, and calculate particle fitness value according to fitness function.

Step 3. Finding initial extreme value.

Finding individual extreme value and population extreme value according to the initial particle fitness value

Step 4. Iterative optimization

Update the particle position and velocity according to Eqs (1) and (2), and update the individual extreme value and population extreme value according to the fitness value of the new particles.

Step 5. Results analysis

2.2 The Random Forest Model

Random Forest (RF) is a machine learning algorithm based on decision tree integration, which was first proposed by statistician Leo Breiman in 2001. As a typical integrated learning method, the algorithm significantly improves the stability and accuracy of the model by constructing multiple decision trees and synthesizing their predictions, effectively solving the problems of weak generalization ability and large fluctuations in prediction of a single classifier [6]. The core implementation process of the algorithm mainly includes the following key links:

1. Random sampling with put-back from the original training set containing N samples, i.e., Bootstrap sampling technique is used to generate multiple training subsets with the same capacity;
2. In the feature selection stage, for the samples possessing M-dimensional features, each time the node splitting is performed, not all the features are considered, but only the features composed from the m features selected randomly are taken into account. Instead of considering all the features, the optimal split features are determined from a subset of m randomly selected features;
3. Each decision tree adopts the full-growth strategy without pruning, in order to ensure the diversity of the base learner, it is necessary to make each decision tree adopt the full-growth strategy without pruning;
4. A large number of decision trees are constructed through the iterative process described above, and a complete Random Forest system is formed;
5. The prediction results of all decision trees are aggregated by the set strategy. Aggregate the prediction results of all decision trees.

Through the synergistic effect of “three randomnesses” (sample randomness, feature randomness, split randomness), this integration mechanism not only significantly enhances the model's generalization ability and overfitting resistance, but also improves the model's prediction accuracy, which makes it an effective tool for dealing with complex classification and regression problems. Similar to most machine learning algorithms, the performance of the random forest model is highly dependent on the setting of key parameters. In Python's scikit-learn implementation, the following three core parameters have a decisive impact on model performance: n_estimators, max_features, and max_depth. The settings of these three hyperparameters significantly affect the model's prediction accuracy and computational efficiency.

Insufficient n_estimators leads to an underfitted model. Excessive n_estimators will increase unnecessary computational overhead, and the choice of max_features and max_depth directly affects the degree of exploration of the feature space, and unreasonable settings will reduce the model generalization ability. In the subsequent study, we will focus on optimizing these two key parameters, and find the optimal parameter combinations through intelligent optimization algorithms to improve the performance of the Random Forest model in water quality prediction tasks.

3 DATA COLLECTION

The water quality monitoring data in this study were mainly obtained from the public database of the National Environmental Quality Monitoring Network for Surface Water (http://www.cnemc.cn), and the monthly monitoring data from key river basins in China were selected as the study samples. The dataset contains seven key water quality indicators, including pH, dissolved oxygen (O2), permanganate index (KMn), ammonia nitrogen (NH4N), total phosphorus (P), total nitrogen (N), and flow direction (Dir).

In the data preprocessing stage, the following quality control measures were used:

1. Missing value processing: for the missing data in the monitoring network, the standard deviation weighted interpolation method based on time series was used. Specifically, the inter-month standard deviation of the historical data of each monitoring station is first calculated, and then the standard deviation is used as the threshold to weight the missing values for interpolation to ensure data continuity.
2. Outlier detection and processing: Physical extreme value calibration for pH (normal range 6-9), dissolved oxygen (>2mg/L) and other indicators
3. Data standardization: Improvement of data normality for NH4N, P and other concentration indicators, and all characteristic variables were finally processed to the [0,1] interval using Min-Max standardization [7].

After pre-processing, it provides a reliable data base for subsequent modeling, and Table 1 shows some of the data:

Table 1 Treated Water Quality Data (Partial)

pH	O2	KMn	NH4N	P	N	Dir	WR
8.39	13.12	6.03	0.114	0.095	2.25	23.3	4
8.52	12.94	5.65	0.105	0.081	2.3	14.6	3
8.39	12.3	5.68	0.106	0.082	2.12	17.5	3
8.49	12.71	5.66	0.17	0.098	2.08	10.9	3
8.54	12.14	5.68	0.099	0.078	2.06	15.8	3
8.53	12.26	5.78	0.101	0.084	2.09	22.1	3
8.5	12.14	5.67	0.106	0.091	2.08	20.9	3
8.51	13.18	5.39	0.106	0.079	1.95	18.5	3
8.51	12.7	5.8	0.125	0.114	2.17	30.6	3
8.52	12.3	5.64	0.122	0.109	2.18	28.7	3
8.5	11.86	6.21	0.139	0.167	2.47	59	4
8.53	12.86	5.47	0.108	0.09	2.04	22.4	3
8.5	12.53	5.49	0.106	0.1	2.07	28.7	3

8.5	12.15	5.62	0.11	0.112	2.3	28.3	3

4 RESULTS

4.1 Introduction of Inertia Weights

The inertia weight reflects the degree of inheritance of the particle to the previous velocity [8].Shi.Y firstly incorporated the inertia weight into the PSO algorithm, which was analyzed to show that larger inertia weights are more favorable for the global search, while smaller inertia weights are more advantageous for the local search. In order to strike a balance between the global and local search capabilities of the algorithm more effectively, Shi.Y proposed linear decreasing inertia weight (LDIW), i.e:

$$\omega(\mathrm{k}) = \omega_{\mathrm{start}}(\omega_{\mathrm{start}} - \omega_{\mathrm{end}})(\mathrm{T_{max}} - \mathrm{k})/\mathrm{T_{max}} \tag{3}$$

where, ω_{start} is the initial inertia weight; ω_{end} is the inertia weight when iterating to the maximum number of iterations; k is the current number of iterations; and$\mathrm{T_{max}}$ is the maximum number of iterations.
Common inertia weight selection:

$$\omega(\mathrm{k}) = \omega_{\mathrm{start}} - (\omega_{\mathrm{start}} - \omega_{\mathrm{end}})\left(\frac{\mathrm{k}}{\mathrm{T_{max}}}\right)^2 \tag{4}$$

$$\omega(\mathrm{k}) = \omega_{\mathrm{start}} + (\omega_{\mathrm{start}} - \omega_{\mathrm{end}})\left[\frac{2\mathrm{k}}{\mathrm{T_{max}}} - \left(\frac{\mathrm{k}}{\mathrm{T_{max}}}\right)^2\right] \tag{5}$$

$$\omega(\mathrm{k}) = \omega_{\mathrm{end}}\left(\frac{\omega_{\mathrm{start}}}{\omega_{\mathrm{end}}}\right)^{1/(1+\mathrm{ck}/\mathrm{T_{max}})} \tag{6}$$

4.2 Test Functions

We chose two of the 23 Benchmark benchmark functions, the Sphere function and the Rastrigin function, to introduce the mechanism of variable inertia weights to improve the particle swarm algorithm to obtain the IPSO on the basis of the particle swarm algorithm with fixed inertia weights at the base.
Sphere test function:

$$\mathrm{f(x)} = \sum_{\mathrm{i=1}}^{\mathrm{n}} \mathrm{x_i}^2 \tag{7}$$

Rastrigin test function:

$$\mathrm{f(x)} = 10\mathrm{n} + \sum_{\mathrm{i=1}}^{\mathrm{n}} \left(\mathrm{x_i^2} - 10\cos(2\pi \mathrm{x_i}) + 10\right) \tag{8}$$

4.3 Comparative Algorithm Performance Evaluation

In the comparison of the two algorithms PSO and IPSO, I chose the Sphere test function as well as the Rastrigin test function from the 23 Benchmark benchmark functions [9]. And used the above three inertia weights for the performance comparison of PSO and IPSO, respectively, so that the resultant mean and variance obtained by running them independently for 10 times are as follows:

Table 2 PSO and IPSO Performance Comparison Data

Func		PSO	IPSO（Linear）	IPSO（Expotential）	IPSO（Quadratic）
Sphere	Mean	6.3262e-01	1.2184E-10	9.7546E-14	9.372E-09
	Var	1.8646e-01	1.3233E-19	5.196E-26	1.1378E-16
Rastrigin	Mean	309.3884	124.5944	129.7789	133.0352
	Var	426.1845	916.549	1050.3886	996.97

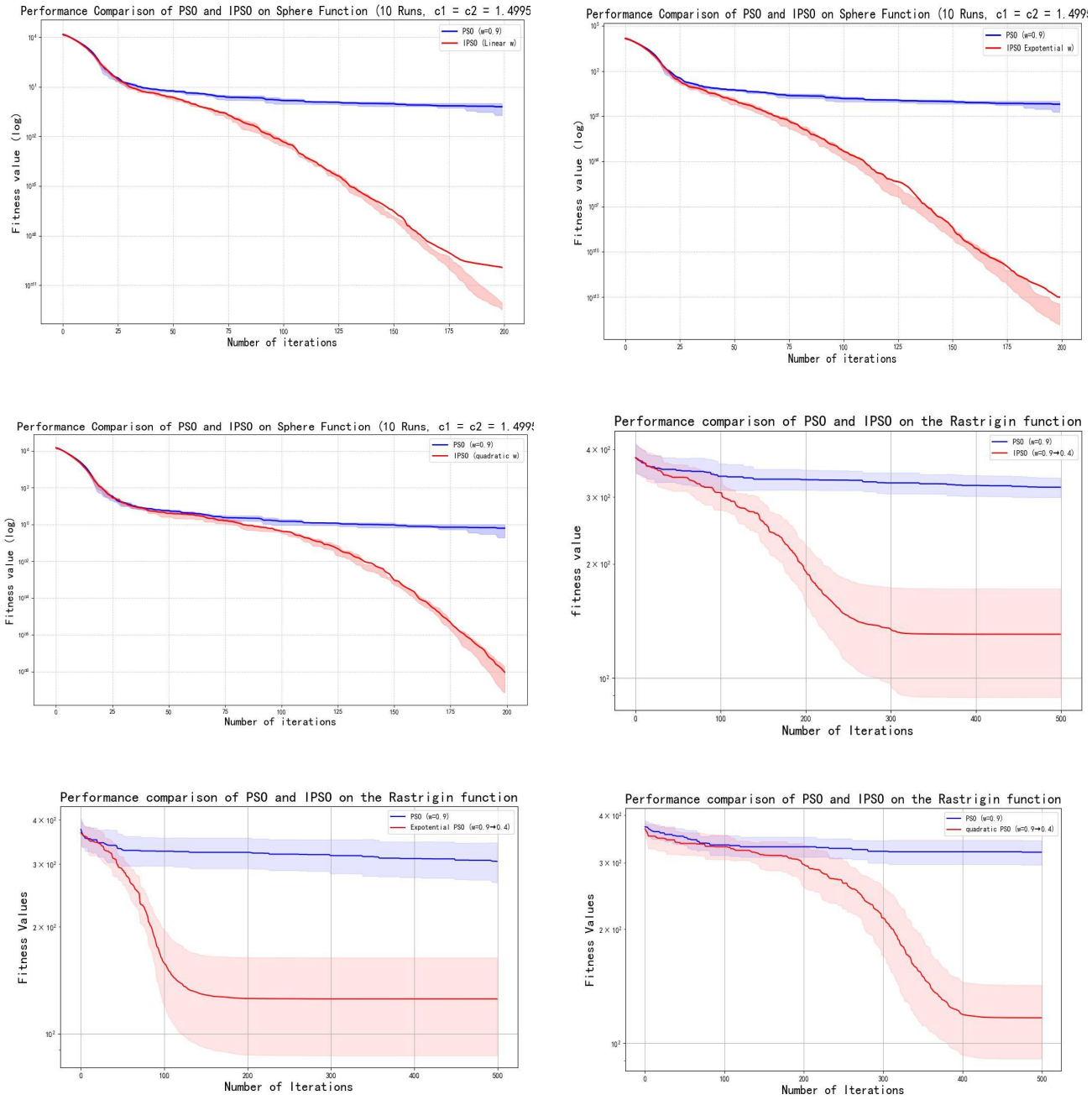

Figure 1 PSO and IPSO Performance Comparison Data

Based on the above six comparison graphs, it is clearly found that IPSO outperforms PSO regardless of the inertia weights and test functions used. The results of testing the performance of the IPSO algorithm using the Sphere test function as well as the Rastrigin test function are shown in Figure 1 as well as in Table 2, Figure1 presents the iterative curves of PSO and IPSO under Linear, Expotential, and Quadrati inertia weights, respectively, and Table 2 records the mean and the variance.

Based on the data and iterative curves in Table 2 and Figure1, it can be seen that the IPSO algorithm outperforms PSO in both the Sphere test function as well as the Rastrigin test function, with the mean values of IPSO (Linear) being 1.2184E-10, 124.5944, and the mean values of IPSO (Expotential) being 9.7546E-14, 129.7789, and the mean values of IPSO (Quadratic) are 9.372E-09, 133.0352, respectively, which are all better than the mean values of PSO, 6.3262e-01, 309.3884, and the minimum value of IPSO is improved by several orders of magnitude compared to PSO.

In summary, the performance of the IPSO algorithm using nonlinear iteration with inertia weights is improved compared to the PSO algorithm, proving that the improvements made on the traditional basis are effective.

4.4 Construction of Coupled Models

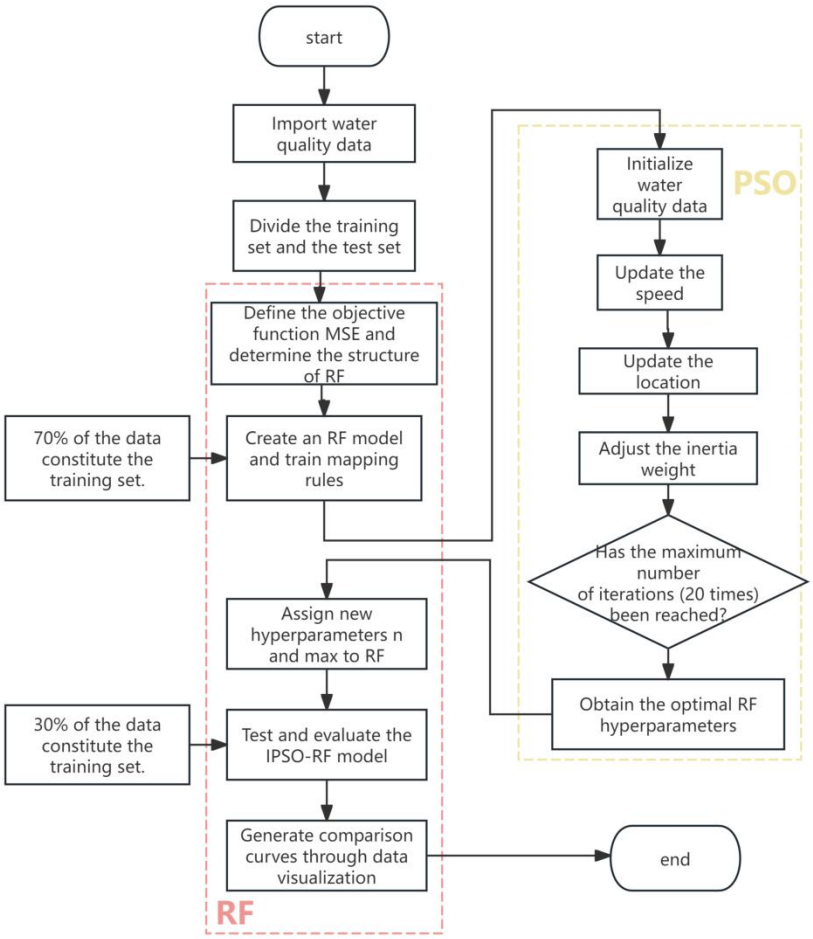

Figure 2 Flowchart of the Coupled Model

The logical structure of the IPSO-RF integrated algorithm constructed is shown in Figure 2. The specific steps are as follows:

Step 1: Import water quality data and normalize it;

Step 2: Divide the training set and test set. Use 70% of the data as the training set for subsequent training of the model, and 30% of the data as the test set for model evaluation;

Step 3: Determine the RF structure based on the input data, then create the basic RF model and train it;

Step 4: Implement the algorithm using the inertia-weight nonlinear iterative IPSO algorithm. According to the settings in the previous text, set relevant parameters and encode the three hyperparameters of RF, namely n_estimators, max_features, and max_depth;

Step 5: Use the mean square error of the sample data as the fitness function, and continuously iterate according to the IPSO algorithm process to update the optimal position and search for the optimal hyperparameter combination;

Step 6: Determine whether the maximum iteration times have been reached. If so, output the best hyperparameter combination to the random forest model and end the iteration. If not, return to Step 5;

Step 7: Assign the optimal n_estimators, max_features, and max_depth parameters of the RF model, construct a new IPSO-RF water quality prediction model, test whether the model accuracy meets the requirements, and output the classification results if it does. Otherwise, return to Step 3.

5 EXPERIMENTAL RESULTS AND DISCUSSION

In this section, the water quality prediction model based on the IPSO-RF algorithm is constructed according to the above, and in order to verify that the model has a better prediction performance, it is validated against the PSO algorithm by comparing the mean square error (MSE) of the two algorithms [10].

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{9}$$

It is further demonstrated that the IPSO-RF model has better performance.

Among them, the parameter settings of the IPSO and PSO algorithms are consistent with those described above, and the population size and the number of iterations is uniformly set to 20 generations. The final model construction was carried out.

Among them, the final optimization results of the IPSO-RF integrated model for the parameters of random forest n and max are: n_estimators=16, max_depth=10, max_features=7; the specific visualization image of the mean square error is shown in Figure 3, and it can be seen that the MSE of PSO is converging to 0.021707, while the MSE of IPSO It is obvious that the mean square error of IPSO is smaller than that of PSO and its performance is more superior. Finally, it is proved that the IPSO-RF model constructed in this paper has certain theoretical and practical significance, and can make more accurate prediction of water quality composition.

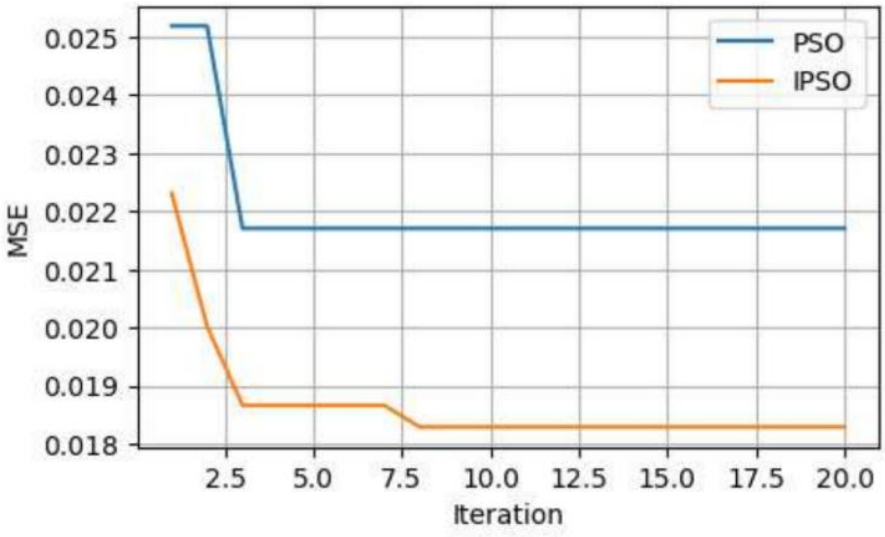

Figure 3 Water Quality Data IPSO and PSO Performance Comparison

6 CONCLUSIONS AND IMPLICATIONS

This study focuses on the construction of an intelligent water quality prediction model, demonstrating the advantages of machine learning models enhanced by intelligent optimization algorithms in the field of environmental monitoring. By improving the standard Particle Swarm Optimization (PSO) algorithm, we propose an IPSO algorithm that incorporates nonlinear iterative inertia weight adjustment. The optimization performance of IPSO and PSO is compared using two benchmark test functions, with experimental results confirming the superior convergence speed and accuracy of the IPSO algorithm. Furthermore, we integrate the IPSO algorithm with Random Forest (RF) to construct an IPSO-RF ensemble prediction model, which is applied to the prediction and analysis of key water quality indicators.

The findings of this study provide an effective technical solution for intelligent water quality monitoring, contributing to more accurate and stable environmental data analysis. The proposed model can assist in early warning systems for water pollution, support decision-making in water resource management, and promote sustainable environmental protection practices.

Innovation:

1. Algorithm Improvement: The introduction of nonlinear adaptive inertia weight in IPSO enhances optimization efficiency, addressing the limitations of standard PSO in local optima escape and convergence speed.
2. Model Integration: The novel combination of IPSO and RF improves prediction accuracy by optimizing hyperparameters and feature selection, offering a robust approach for water quality forecasting.
3. Practical Application: The IPSO-RF model demonstrates strong generalization capability in real-world water quality monitoring scenarios, providing a reliable tool for environmental big data analysis.

COMPETING INTERESTS

The authors have no relevant financial or non-financial interests to disclose.

REFERENCES

[1] Rajaee T, Khani S, Ravansalar M. Artificial intelligence-based single and hybrid models for prediction of water quality in rivers: A review. Chemometrics and Intelligent Laboratory Systems, 2020, 200: 103978.
[2] Hmoud Al-Adhaileh M, Waselallah Alsaade F. Modelling and prediction of water quality by using artificial intelligence. Sustainability, 2021, 13(8): 4259.
[3] Chen Y, Song L, Liu Y, et al. A review of the artificial neural network models for water quality prediction. Applied Sciences, 2020, 10(17): 5776.
[4] Lu H, Ma X. Hybrid decision tree-based machine learning models for short-term water quality prediction. Chemosphere, 2020, 249: 126169.

[5] Jain M, Saihjpal V, Singh N, et al. An overview of variants and advancements of PSO algorithm. Applied Sciences, 2022, 12(17): 8392.
[6] Genuer R, Poggi J M, Genuer R, et al. Random forests. Springer International Publishing, 2020.
[7] Deepa B, Ramesh K. Epileptic seizure detection using deep learning through min max scaler normalization. Int. J. Health Sci, 2022, 6: 10981-10996.
[8] Choudhary S, Sugumaran S, Belazi A, et al. Linearly decreasing inertia weight PSO and improved weight factor-based clustering algorithm for wireless sensor networks. Journal of ambient intelligence and humanized computing, 2023: 1-19.
[9] Chen T, Chen X, Chen W, et al. Learning to optimize: A primer and a benchmark. Journal of Machine Learning Research, 2022, 23(189): 1-59.
[10] Chicco D, Warrens M J, Jurman G. The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation. Peerj computer science, 2021, 7: e623.