Mathematical Biosciences and Engineering

http://www.aimspress.com/journal/MBE

MBE, 18(4): 3646–3666.
DOI: 10.3934/mbe.2021182
Received: 07 March 2021
Accepted: 19 April 2021
Published: 26 April 2021

Research article

AI-HydSu: An advanced hybrid approach using support vector regression and particle swarm optimization for dissolved oxygen forecasting

Dashe Li[1,2], **Xueying Wang**[1,2,*], **Jiajun Sun**[1,2] **and Huanhai Yang**[1,2]

[1] School of Computer Science and Technology, Shandong Technology and Business University, Yantai 264005, China

[2] Co-innovation Center of Shandong Colleges and Universities: Future Intelligent Computing, Yantai 264005, China

* **Correspondence:** Email: 2018410036@sdtbu.edu.cn.

Abstract: Since the variations in the dissolved oxygen concentration are affected by many factors, the corresponding uncertainty is nonlinear and fuzzy. Therefore, the accurate prediction of dissolved oxygen concentrations has been a difficult problem in the fishing industry. To address this problem, a hybrid dissolved oxygen concentration prediction model (AI-HydSu) is proposed in this paper. First, to ensure the accuracy of the experimental results, the data are preprocessed by wavelet threshold denoising, and the advantages of the particle swarm optimization (PSO) algorithm are used to search the solution space and select the best parameters for the support vector regression (SVR) model. Second, the prediction model optimizes the invariant learning factors in the standard PSO algorithm by using nonlinear adaptive learning factors, thus effectively preventing the algorithm from falling to local optimal solutions and accelerating the algorithm's optimization search process. Third, the velocities and positions of the particles are updated by constantly updating the learning factors to finally obtain the optimal combination of SVR parameters. The algorithm not only performs searches for the penalty factor, kernel function parameters, and error parameters in SVR but also balances its global and local search abilities. A dissolved oxygen concentration prediction experiment demonstrates that the proposed model achieves high accuracy and a fast convergence rate.

Keywords: support vector regression; particle swarm optimization; dissolved oxygen; adaptive; hybrid forecasting model

1. Introduction

Dissolved oxygen (DO) is a factor of great importance to aquaculture. When the DO concentration is slightly below the critical level, the cultured aquatic animals will start to show a decrease in feeding, slow growth, an increase in the feed coefficient, and a decrease in shrimp shelling frequency. If the water is continuously low in oxygen for a long time, it reduces the resistance of the animals to environmental stressors and diseases, which can easily cause production losses. Therefore, the aquaculture industry needs to master the changing pattern in dissolved oxygen to reduce risks and increase the success rate of aquaculture.

In recent years, domestic and foreign scholars have made much progress in the accurate prediction of water quality parameters. The commonly used methods for water quality dissolved oxygen prediction mainly include linear regression [1, 2], time series prediction methods [3, 4], wavelet analysis prediction methods [5, 6], artificial neural network prediction methods [7, 8], etc. Olyaie et al. [9] used three different methods for DO prediction. The comparison of the estimation accuracy of various intelligent models showed that a support vector machine (SVM) was the most accurate DO estimation model compared to other models. Kisi et al. [10] proposed a new integration method, Bayes model averaging (BMA), for estimating hourly DO concentration. Raheli et al. [11] proposed a hybrid prediction model called an MLP-FFA model; this model is based on the firefly algorithm (FFA), which acts as a metaheuristic optimizer and integrates with a multilayer perceptron (MLP) with the ability to predict monthly water quality in the Langat River Basin. Masrur Ahmed [12] developed a feedforward neural network (FFNN) model and radial basis function (RBF) neural network (RBFNN) model to predict biochemical oxygen demand (BOD) and chemical oxygen demand (COD) in the Surma River. Keshtegar et al. [13] developed and compared two nonlinear mathematical modeling methods, the modified response surface methodology (MRSM) and MLP neural network (MLPNN), to simulate daily DO concentrations. Csábrági et al. [14] used four algorithmic models for the prediction of dissolved oxygen concentration. Experimental results showed that the nonlinear model had better prediction results than the linear model. Heddam et al. [15] used different extreme learning machine (ELM) models to predict DO concentrations. The experimental results showed that the ELM was more effective than an MLPNN and multiple linear regression (MLR) in simulating DO concentrations in riverine ecosystems.

The above water quality analysis and prediction models are mainly focused on shallow learning models based on artificial neural networks. The neural network model has a high self-learning and generalization ability compared with the traditional prediction model. It can not only solve complex nonlinear approximation problems but also has a good simulation and prediction effect on the development trend in water environments. Even so, there are still some defects in the experimental process. For example, the learning speed of the algorithm is too slow to make the training of the model efficient, and it easily falls into local minima during the training process, which makes the results inaccurate.

A support vector machine (SVM) [16] is a class of generalized linear classifiers that performs binary classification of data in a supervised manner whose decision boundary is the maximum margin hyperplane solved for the learned samples. Its minimization of actual risk by seeking structural risk minimization can solve both classification and regression problems. Extending SVMs from classification problems to regression problems yields support vector regression (SVR). At this time, the standard SVM algorithm is also known as support vector classification (SVC). The hyperplane decision boundary in SVC is the SVR model, and this paper used the SVR method to predict DO dissolution.

Luo et al. [17] proposed a hybrid prediction method combining the discrete Fourier transform

(DFT) with SVR. Ahmad et al. [18] proposed a novel SVR model for predicting the splice strength of unconstrained beam samples. Zhang et al. [19] proposed a wind power prediction model based on a combination of particle swarm optimization (PSO)-SVR and gray theory. Dodangeh et al. [20] proposed the group method of data handling (GMDH), which is based on SVR for meta-optimization and data processing. Xiang et al. [21] proposed a combinatorial model for extracting information based on ensemble empirical mode decomposition (EEMD), which employs various supervised learning methods for different components of the input data. Panahi et al. [22] proposed the modified simulated annealing (MSA) algorithm to optimize the SVR prediction model with an improved annealing schedule and perturbation range. All of the abovementioned papers have optimized SVR to some extent, but there are limitations in data application, and some of the models are computationally complex.

The penalty coefficient C and the kernel function parameter σ in the SVR model have a significant impact on the model. The penalty coefficient C reflects the degree to which the algorithm penalizes sample data that exceed the accuracy ε, and its value affects the complexity and stability of the model. When C is too small, the penalty for the sample data exceeding the accuracy ε is small, and the training error becomes larger. When C is too large, the learning accuracy increases accordingly, but the generalizability of the model becomes poor. When σ is too large, the model is easily underfit, and the prediction accuracy subsequently decreases. When σ is too small, the model is easily overfit and the training time increases, while the demand for the number of samples increases. Therefore, choosing the appropriate C, σ, and ε greatly improves the prediction accuracy of the model. The most commonly used parameter selection method is grid search, where the best combination of parameters is obtained through continuous trial and error. However, this method also has disadvantages. It can find the global optimal solution when the set interval is large enough, and the step size is small enough. However, this inevitably generates a large number of unnecessary and invalid computations, which in turn leads to an exponential increase in the computation time.

Therefore, combining the SVR model with the PSO algorithm has the advantage of searching the solution space and is computationally simple. This paper proposed and constructed a DO prediction model based on the nonlinear adaptive learning factor of the PSO algorithm fused with an SVM (AI-HydSu). The experimental results showed that 1) the proposed AI-HydSu model performs well in a small nonlinear sample of DO data and has a faster convergence speed and higher accuracy than other similar algorithms; 2) the AI-HydSu method can effectively solve the problem of long model operation time caused by SVR using the grid search method to select the model structure parameters; 3) compared with other similar algorithms, the model proposed in this paper has a better generalizability and has a better prediction effect for water quality parameter prediction in different waters.

The other chapters of this paper are organized as follows: Chapter 2 of this paper introduces the related methods; Chapter 3 introduces the algorithm and constructs the model in detail; Chapter 4 shows the experimental results of the proposed algorithm and conducts a comparative analysis; Chapter 5 summarizes the main ideas of this paper and introduces the next work.

2. Materials and methods

2.1. Support vector regression

SVR uses the decision boundary of the optimal hyperplane in SVC to build a regression model. Suppose the set of load data is (x_i, y_i), where x_i is the sample input, y_i is the sample output, and $\phi(x_i)$ denotes the feature vector of x_i after mapping to the high-dimensional feature space. Its corresponding optimal hyperplane equation is:

$$f(x_i) = \omega^T \phi(x_i) + b \tag{2.1}$$

where ω is the normal vector and b is the displacement term.

For the sample (x_i, y_i), traditional regression models usually calculate the loss directly based on the difference between the model output $f(x_i)$ and the true output y_i; the loss is 0 if and only if $f(x_i)$ is exactly the same as y_i. In contrast, SVR assumes that we can tolerate an error of at most ε between $f(x_i)$ and y_i. The loss is calculated only when the absolute value of the difference between $f(x_i)$ and y_i is greater than ε.

The essence of the SVR model training process is to find the optimal ω and b such that $f(x_i)$ approximates y_i, resulting in a convex optimization function:

$$\min_{\omega,b} \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{m}\left(\zeta_i + \overset{\wedge}{\zeta_i}\right) \tag{2.2}$$

where C denotes the penalty coefficient, which serves to balance the correlation between the structural risk $\frac{1}{2}\|\omega\|^2$ and empirical risk $\sum_{i=1}^{m}\left(\zeta_i + \overset{\wedge}{\zeta_i}\right)$; ζ_i and $\overset{\wedge}{\zeta_i}$ denote the relaxation variables located on both sides of the hyperplane.

The constraints corresponding to this convex optimization function are:

$$s.t\begin{cases} f(x_i) - y_i \leq \varepsilon + \zeta_i \\ y_i - f(x_i) \leq \varepsilon + \zeta \\ \zeta_i, \overset{\wedge}{\zeta_i} \geq 0, i = 1, 2..., m \\ C > 0 \end{cases} \tag{2.3}$$

where the relaxation factor ε represents the deviation value of $f(x_i)$ and y_i. Then, Lagrange multipliers a_i and a_i^* are introduced:

$$
\begin{aligned}
L\left(\omega, b, a, a_i^*, \xi, \overset{\wedge}{\xi}, \mu, \overset{\wedge}{\mu}\right) = \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{m}\left(\zeta_i + \overset{\wedge}{\zeta}_i\right) \\
- \sum_{i=1}^{m}\mu_i\xi_i - \sum_{i=1}^{m}\overset{\wedge}{\mu}_i\overset{\wedge}{\xi}_i \\
+ \sum_{i=1}^{m} a_i\left(f\left(x_i\right) - y_i - \varepsilon - \zeta_i\right) \\
+ \sum_{i=1}^{m} a_i^*\left(y_i - f\left(x_i\right) - \varepsilon - \overset{\wedge}{\zeta}_i\right)
\end{aligned}
\tag{2.4}
$$

To solve this constrained convex optimization function, ω and the SVR function $f(x)$ can be obtained as follows:

$$\omega = \sum_{i=1}^{m}\left(a_i^* - a_i\right)\phi\left(x_i\right) \tag{2.5}$$

$$
\begin{cases}
f(x) = \sum\limits_{i=1}^{m}\left(a_i^* - a_i\right)K\left(x_i, x\right) + b \\
K\left(x_i, x\right) = \exp(-\sigma\|x - x_i\|^2)
\end{cases}
\tag{2.6}
$$

where σ represents the kernel parameter of the kernel function $K\left(x_i, x\right)$. The resulting kernel function $K\left(x_i, x\right)$ can improve the model's ability to deal with nonlinear regression problems. The RBF kernel can effectively improve the fitting effect and prediction performance of the model, so it is often used as the kernel function to optimize the SVR model.

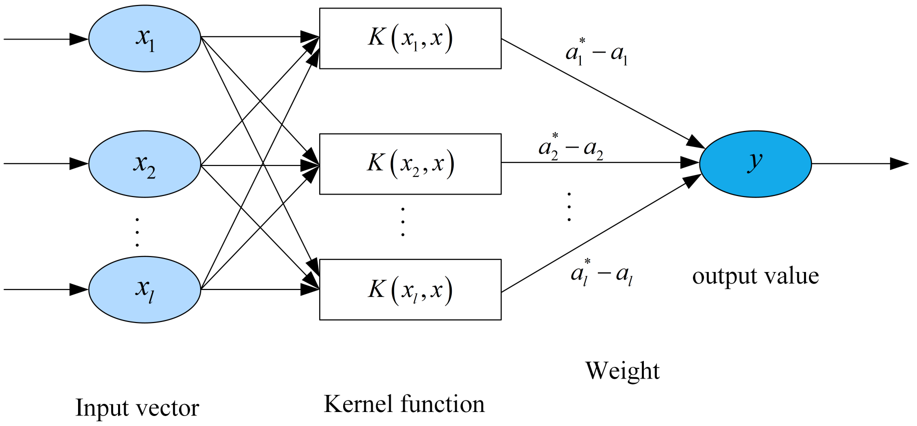

Figure 1. SVR model.

2.2. *Particle swarm optimization*

Eberhart and Kennedy first proposed PSO in 1995 [23]. The algorithm was modified for Hepper's model of a simulated bird swarm (fish school) to enable particles to fly to the solution space and land at the best solution, resulting in the PSO algorithm. The PSO algorithm has many advantages, such as simplicity, ease of implementation, no gradient information required, few parameters, and in particular, its natural real number encoding feature is particularly suitable for dealing with real optimization problems while having a deep intelligent background. Therefore, PSO is suitable for both scientific research and engineering applications.

If a particle is used to simulate the abovementioned bird individuals, then each particle can be regarded as a searching individual in the N-dimensional search space. The current position of the particle is a candidate solution of the corresponding optimization problem, and the flight process of the particle is the search process of the individual [24]. The flight speed of the particle can be dynamically adjusted according to the particle's historical optimal position and the population's historical optimal position. The particle has only two properties: velocity and position. The velocity represents the speed of movement, and the position represents the direction of movement [25]. The optimal solution searched by each individual particle is called the individual extremum, and the optimal individual extremum of the particle swarm is used as the current global optimal solution. By continuously iterating, the velocity and position are updated, and finally, the optimal solution satisfying the termination condition is obtained.

The algorithm flow is as follows.

1. Initialization

First, this study sets the maximum number of iterations, the number of independent variables of the objective function, the maximum velocity of the particles, and the position information for the whole search space. In this paper, the velocity and position are randomly initialized on the velocity interval and the search space, and the particle swarm size is set to M. Each particle is randomly initialized with a flying velocity.

2. Individual extremes and global optimal solutions

The fitness function is defined, and the individual extremes are the optimal solutions found for each particle. Then, a global value is found from these optimal solutions, called the current global optimum, which is updated by comparing it with the historical global optimum.

3. Equation of the updated velocity and position

$$\begin{aligned} V_{id}^{k+1} =& \omega V_{id}^{k} + C_1 random(0,1)(P_{id}^{k} - X_{id}^{k}) \\ &+ C_2 random(0,1)(P_{gd}^{k} - X_{id}^{k}) \end{aligned} \tag{2.7}$$

$$X_{id}^{k+1} = X_{id}^{k} + rV_{id}^{k+1} \tag{2.8}$$

where ω is the inertia factor with a nonnegative value. When ω is larger, the global optimization ability is stronger, and the local optimization is weaker. In contrast, the local optimization is stronger and the global optimization is worse. C_1, C_2 is called the acceleration factor, which is generally a constant. The former is the individual learning factor of the particle, which is the weight coefficient of the particle tracking its own historical optimal value, and the latter is the particle's social learning factor, which is the weight coefficient of the particle tracking group's optimal value. $random(0,1)$ represents the random number in the interval (0, 1), P_{id} represents the dth dimension of the individual extreme value of the ith variable, P_{gd} represents the dth dimension of the global optimal solution of the ith variable, X_{id} represents the position of the dth dimension of the ith variable, and V_{id} represents the velocity of the dth dimension of the ith variable. r is a coefficient added in front of the velocity when updating the position. This coefficient is called the constraint factor and is usually set to 1. The particles here track their own historical optimal value and global (group) optimal value at the same time to change their position and velocity, so this method is also called the global version of the standard PSO algorithm [26].

4. Termination Conditions

(1) The set number of iterations is reached;

(2) The adaptation value reaches a certain value.

The acceleration factors C_1 and C_2 represent the weight of the statistical acceleration term that pushes each particle to the individual optimal solution and the global optimal solution position. A lower value allows the particles to hover outside the target area before being pulled back, and a higher value causes the particles to suddenly rush toward or over the target area. When $C_2 = 0$, it is called the self-aware PSO algorithm (i.e., "only self, no society"); there is social sharing without information at all, resulting in the slow convergence of the algorithm. When, $C_1 = 0$, it is called the selfless PSO algorithm (i.e., "only society, no self"); it will quickly lose group diversity, easily fall into the local optimal solution and cannot jump out. When C_1 and C_2 are not 0, then it is called the complete PSO algorithm; in this case, it is easier to maintain the balance between the convergence speed and the search effect, which is the best choice.

2.3. Wavelet threshold denoising

A prerequisite for accurate water quality prediction is the quality and accuracy of the DO dataset. During data acquisition, noise may be generated if the sensor equipment has a low accuracy and performance degradation. If the unprocessed raw data are directly used for DO prediction, the data prediction accuracy will be affected. Therefore, noise reduction is needed during raw data acquisition to ensure the DO prediction accuracy.

Depending on the noise energy, conventional denoising methods generally focus on high frequencies. Based on the signal spectrum distributed in a finite interval, the Fourier transform can be used to transform the noisy signal to the frequency domain, and then a low-pass filter can be used for filtering. However, the Fourier transform-based denoising method cannot effectively distinguish between the high-frequency part of the useful signal and the high-frequency interference caused by noise, so there is a tradeoff between protecting the localization of the signal and noise suppression. The wavelet transform can preserve the spikes and local prominence of the signal very well. Given the advantageous nature of the wavelet thresholding method, the widely used wavelet thresholding denoising [27] is used here for analysis.

The basic idea of wavelet threshold denoising is the selection of the generated wavelet coefficients after transforming the signal through the Mallat algorithm [28]. Given that the wavelet coefficients of the signal are larger after wavelet decomposition, the wavelet coefficients of the noise are smaller than the wavelet coefficients of the signal. By selecting a suitable threshold value, wavelet coefficients greater than the threshold value are considered signal and should be retained; those less than the threshold value are considered noise and should be set to zero for the purpose of denoising.

The basic steps of the wavelet threshold contraction method are as follows:

(1) Wavelet basis function selection: Generally, the wavelet basis function is selected based on the support length, vanishing moment, symmetry, regularity, similarity, etc., for comprehensive consideration. Given that wavelet basis functions have their own characteristics in the processed signals, no single wavelet basis function can achieve optimal denoising results for all types of signals.

In general, the Daubechies wavelet and symlet families [29] are often used in speech denoising, and a wavelet with N layers is selected to perform wavelet decomposition of the signal [30].

(2) Selecting the number of decomposition layers: On the one hand, the larger the number of decomposition layers is, the more obvious the differences in the noise and signal performance characteristics, and the more beneficial in the separation of the two. On the other hand, the larger the number of decomposition layers is, the larger the distortion of the reconstructed signal, which will affect the final denoising effect to a certain extent. Therefore, extra attention is needed to address the contradiction between the two and choose an appropriate decomposition scale.

(3) Threshold selection: In the wavelet domain, the effective signal corresponds to a large coefficient, while the noise corresponds to a small coefficient. At this point, the coefficients corresponding to the noise in the wavelet domain still follow the Gaussian white noise distribution. The threshold selection rule is based on the model y=f(t)+e, where e is Gaussian white noise (N(0,1)). Therefore, the evaluation of the threshold that can eliminate noise in the wavelet domain can be performed by wavelet coefficients or the original signal. Currently, the most common threshold selection methods are fixed threshold estimation, extreme value threshold estimation, unbiased likelihood estimation, and heuristic estimation (N is the signal length).

(4) Threshold function selection: After determining the threshold of Gaussian white noise in the wavelet coefficients (domain), a threshold function is needed to filter these wavelet coefficients containing noise coefficients and remove the Gaussian noise coefficients. Among them, the commonly used thresholding functions are soft thresholding and hard thresholding methods.

(5) Reconstruction: The signal is reconstructed with the processed coefficients.

3. AI-HydSu model

In the early stage of the algorithm search, the searchability of the particle itself is evaluated, and the later stage of the algorithm search should focus on the optimal global particle. Therefore, in this paper, a nonlinear cosine function is used for the acceleration factor C_1, and a nonlinear sine function is used for the acceleration factor C_2:

$$C_1 = \begin{cases} C_{1\max} \times \cos(k/m) & C_1 > 0 \\ C_{1low} & else \end{cases} \tag{3.1}$$

$$C_2 = \begin{cases} C_{2\max} \times \sin(k/m) & C_2 > 0 \\ C_{2low} & else \end{cases} \tag{3.2}$$

The acceleration factor is updated continuously with an increasing number of iterations to accommodate the particle velocity updates. In the late stage of the algorithm search, the acceleration factor can obtain a larger C_2 and a smaller C_1, thus jumping out of the local extremes.

The propose AI-HydSu algorithm refers to the PSO algorithm based on the nonlinear adaptive learning factor's optimization capability to continuously train the SVR model by searching for the parameters C, σ, and ε. The PSO algorithm based on the nonlinear adaptive learning factor has a better ability to optimize high-dimensional functions, so it has a better ability to optimize the parameters of high-dimensional kernel functions in SVR to achieve the accurate prediction of the DO concentration. Applying DO prediction to optimize the two parameters in SVR not only improves the parameter search capability but also better balances the global and local search capability of the algorithm and reduces the prediction time.

Algorithm 1 Particle Swarm Optimization & Support Vector Regression

```
Use wavelet threshold denoising to denoise the original data
// Initialize the relevant parameters of the PSO algorithm and SVR parameters
Initialize $C$, $\sigma$, $\varepsilon$, locations $x_i$ and velocities $v_i$
// Determine the fitness function of the particle swarm
Fitness function $G(x) = \frac{1}{n}\sum_{i=1}^{N}(f(x_i) - y)^2$
// Find the optimal solution
Repeat
for each particle $i \in M$ do
    if $G(x_{id}) < G(p_{id})$ then
        $p_{id} = x_{id}$
    end if
    if $G(p_{id}) < G(p_{gd})$ then
        $p_{gd} = p_{id}$
    end if
end for
// Update particle velocity and position
for each particle $i \in M$ do
    $V_{iC}^{k+1} = \omega V_{iC}^{k} + C_1 random(0,1)(P_{iC}^{k} - X_{iC}^{k}) + C_2 random(0,1)(P_{gC}^{k} - X_{iC}^{k})$
    $X_{iC}^{k+1} = X_{iC}^{k} + rV_{iC}^{k+1}$
    $V_{i\varepsilon}^{k+1} = \omega V_{i\varepsilon}^{k} + C_1 random(0,1)(P_{i\varepsilon}^{k} - X_{i\varepsilon}^{k}) + C_2 random(0,1)(P_{g\varepsilon}^{k} - X_{i\varepsilon}^{k})$
    $X_{i\varepsilon}^{k+1} = X_{i\varepsilon}^{k} + rV_{i\varepsilon}^{k+1}$
    $V_{i\sigma}^{k+1} = \omega V_{i\sigma}^{k} + C_1 random(0,1)(P_{i\sigma}^{k} - X_{i\sigma}^{k}) + C_2 random(0,1)(P_{g\sigma}^{k} - X_{i\sigma}^{k})$
    $X_{i\sigma}^{k+1} = X_{i\sigma}^{k} + rV_{i\sigma}^{k+1}$
    // Use formulas (3.1) and (3.2) to update $C_1$ and $C_2$
    Update $C_1$ and $C_2$ and by formulas (3.1) and (3.2)
end for;
// Update iterations
T=T+1
// Determine whether the iteration termination requirement has been met
Until T>Tmax
```

The AI-HydSu DO concentration prediction model is shown in Figure 2. First, the noise in the original data are reduced using the wavelet threshold denoising algorithm, and the initial values of the parameters of the PSO algorithm (i.e., the initial velocity and initial position of the particles) are determined; the parameters to be optimized by the SVR model (i.e., the penalty parameters, kernel coefficients, and algorithm accuracy) are identified, and the optimization-seeking intervals for each parameter are determined. Second, the fitness function of the SVR model is set. The dissolved oxygen prediction model proposed in this paper uses the mean squared error (MSE) function as the fitness function. This paper then takes advantage of the search in the solution space of the PSO algorithm to update the velocities and positions of the particles by introducing nonlinear learning factors and then finding the individual optimal solutions and the global optimal solution. Finally, the DO concentration is predicted by using the optimal combination of SVR parameters, and the optimal combination of the SVR parameters is obtained. The specific process is shown in Algorithm 1.

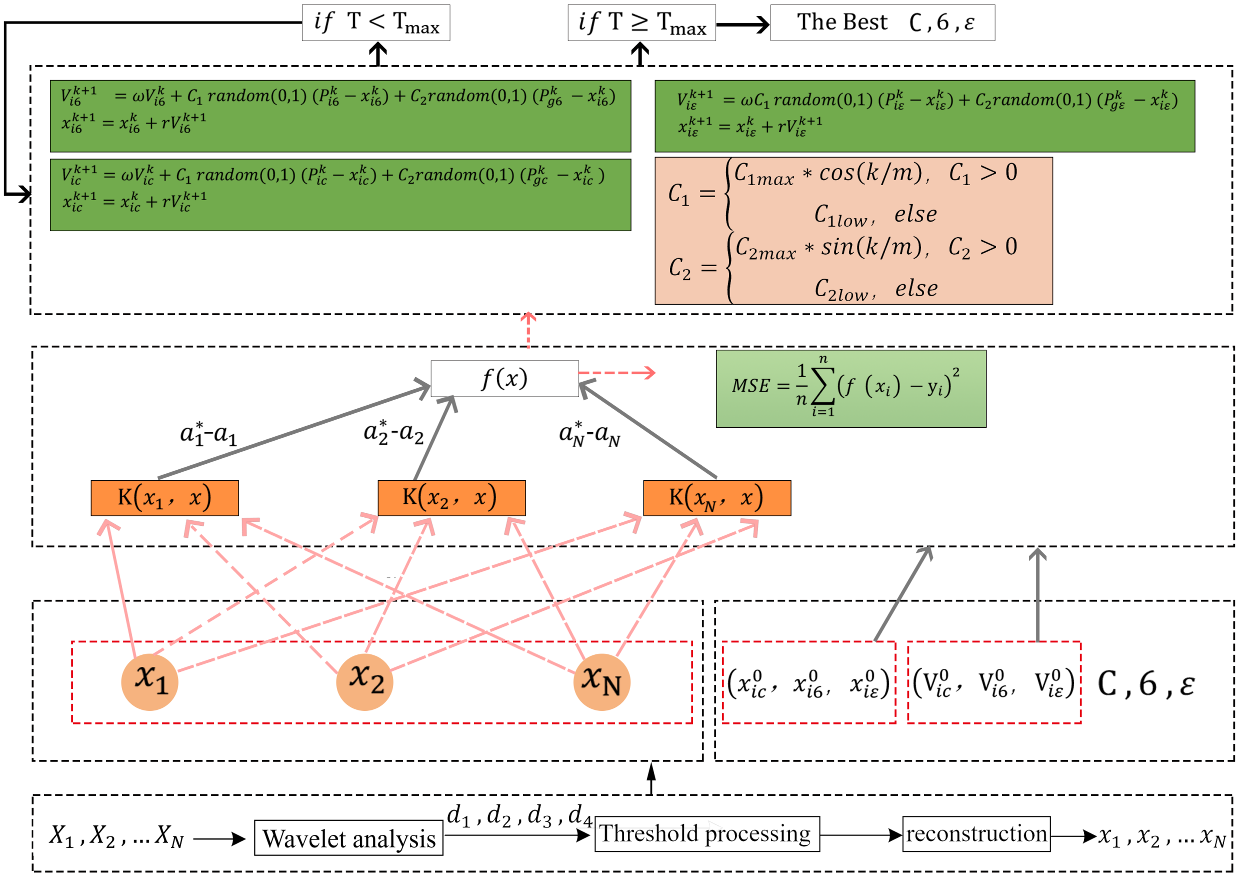
$if\ T < T_{max}$
$if\ T \geq T_{max}$
The Best C , 6 , ε
$V_{i6}^{k+1} = \omega V_{i6}^{k} + C_1\, random(0,1)\, (P_{i6}^{k} - x_{i6}^{k}) + C_2 random(0,1)\, (P_{g6}^{k} - x_{i6}^{k})$
$x_{i6}^{k+1} = x_{i6}^{k} + rV_{i6}^{k+1}$
$V_{i\varepsilon}^{k+1} = \omega C_1\, random(0,1)\, (P_{i\varepsilon}^{k} - x_{i\varepsilon}^{k}) + C_2 random(0,1)\, (P_{g\varepsilon}^{k} - x_{i\varepsilon}^{k})$
$x_{i\varepsilon}^{k+1} = x_{i\varepsilon}^{k} + rV_{i\varepsilon}^{k+1}$
$V_{ic}^{k+1} = \omega V_{ic}^{k} + C_1\, random(0,1)\, (P_{ic}^{k} - x_{ic}^{k}) + C_2 random(0,1)\, (P_{gc}^{k} - x_{ic}^{k})$
$x_{ic}^{k+1} = x_{ic}^{k} + rV_{ic}^{k+1}$
$C_1 = \begin{cases} C_{1max} * cos(k/m), & C_1 > 0 \\ C_{1low}, & else \end{cases}$
$C_2 = \begin{cases} C_{2max} * sin(k/m), & C_2 > 0 \\ C_{2low}, & else \end{cases}$
$f(x)$
$MSE = \frac{1}{n}\sum_{i=1}^{n}(f\ (x_i) - y_i)^2$
a_1^*-a_1
a_2^*-a_2
a_N^*-a_N
$K(x_1,\ x)$
$K(x_2,\ x)$
$K(x_N,\ x)$
x_1
x_2
x_N
$(x_{ic}^0,\ x_{i6}^0,\ x_{i\varepsilon}^0)$
$(V_{ic}^0,\ V_{i6}^0,\ V_{i\varepsilon}^0)$
C , 6 , ε
$X_1, X_2, \ldots X_N$
Wavelet analysis
d_1, d_2, d_3, d_4
Threshold processing
reconstruction
$x_1, x_2, \ldots x_N$

Figure 2. AI-HydSu model.

4. Experimental analysis

4.1. Data source and preprocessing

In this paper, the DO concentration in a shrimp farming base in a marine ranch in Yantai in China's Blue Economic Zone, was taken as the research object. The sampling period was August 1, 2016, June 30, 2020, with 10-minute intervals adopted. The DO data for 54 consecutive days were used as the data sample set for this experiment, with a total sample size of 7776 samples.

The experimental data were obtained from various aquaculture farms in the Blue Economic Zone of China. Figure 3 shows some of the raw DO signal sequences. Figure 3 shows that the changes in the DO peaks and troughs in the farmed water are relatively substantial, showing strong nonlinear, nonsmooth, and periodic characteristics, and the raw DO signal contains noise. If used directly for model training, it will increase the training time and lead to slow convergence or even a failure to converge. To reduce the interference of noise and obtain the real DO signal, it is necessary to reduce the amount of noise the raw signal of the monitored DO concentration.

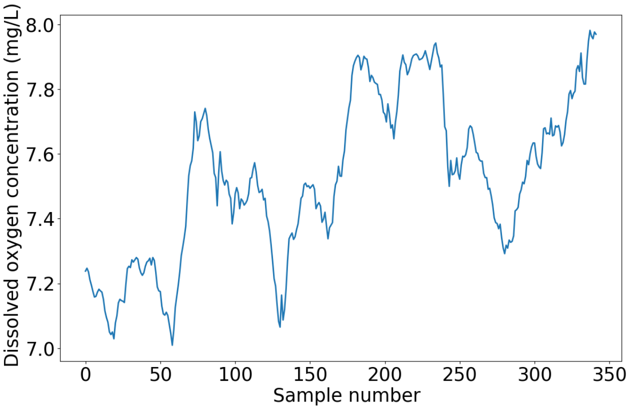

Figure 3. Raw data.

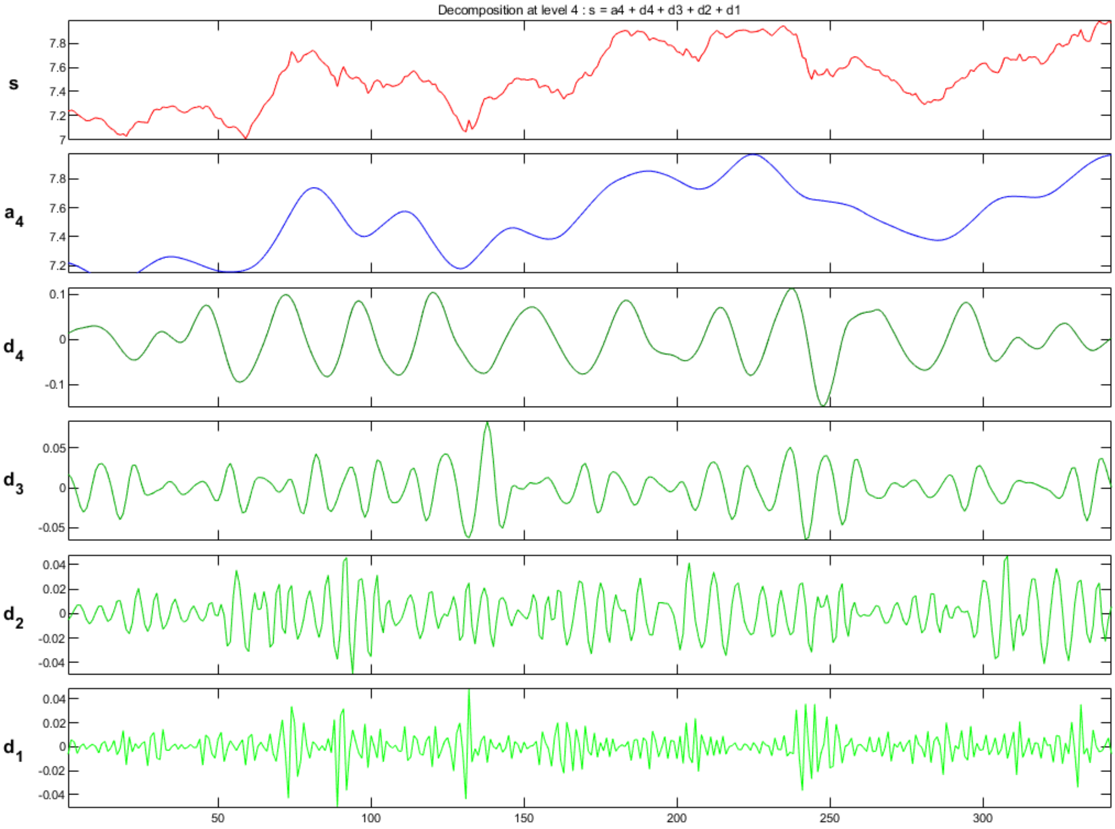

Figure 4. Wavelet decomposition.

In this study, the noise in the raw signal of the monitored DO was reduced by using wavelet threshold denoising. The noise-containing signal was decomposed by orthogonal wavelets at each scale, as shown in Figure 4. Figure 4 shows the wavelet transform of the original data using the db8 wavelet with 5 decomposition layers.

In this study, the decomposition values at large scales (low resolution) are retained, and for the decomposition values at small scales, a default threshold is used. Wavelet coefficients with amplitudes below this threshold are set to 0, and those above this threshold are retained. Finally, in this study, the processed wavelet coefficients are reconstructed using a wavelet transform to recover the useful signal. The noise reduction process is carried out on the raw signal of the monitored DO, and its noise is shown in Figure 5, while the smoothed data after removing the noise are shown in Figure 6. The noise reduction model based on wavelet analysis can effectively reduce the interference of noise in the farm water quality parameters of the DO monitoring data, which can not only retain the useful components of the original DO signal but also reflect the change trend in DO concentration.

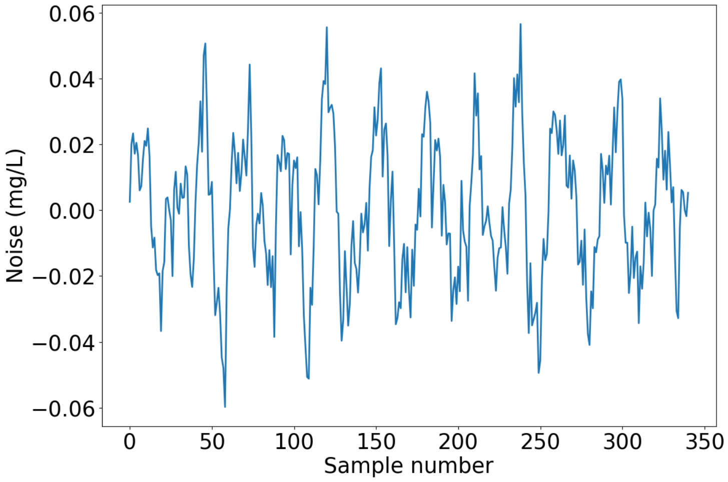

Figure 5. Noise data.

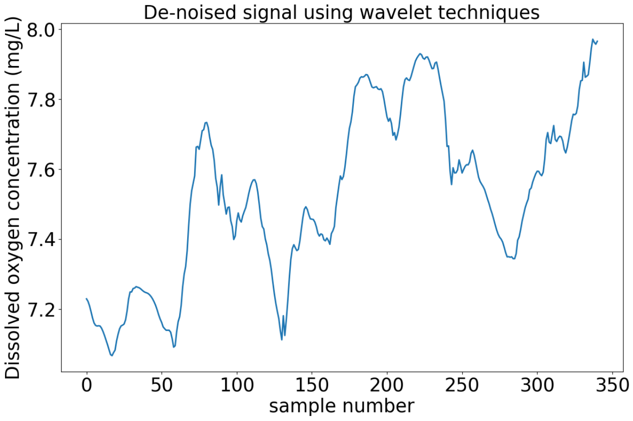

Figure 6. Smoothed data.

4.2. Algorithm implementation and testing

In this study, the AI-HydSu-based DO prediction simulation for shrimp aquaculture was written in Python 3.7. The dataset was divided into training and testing sets for DO concentration prediction. The experiment use 95% of the dataset as the training set and 5% as the test set.The DO data of the first half-hour were used as the visual layer input quantity to predict the change in DO in the next moment. To verify the reliability of the algorithms, the dataset was disrupted without rules. The maximum population size for the PSO algorithm and AI-HydSu algorithm was set to N=10; the maximum number of iterations was set to Tmax=20; the initial values of acceleration factors were set to $C_1 = 4$ and $C_2 = 4$, and the initial weight was set to $w = 0.73$.

The fitness curves of the two algorithms are shown in Figures 7 and 8. It can be seen that the PSO algorithm [31] without the nonlinear learning factor falls into a local optimum at evolutionary generation 3 and keeps falling into the local optimum afterwards, and finally, the MSE function value stabilizes at 0.001065 at 14 iterations. In contrast, the AI-HydSu algorithm has a large slope at the beginning and stabilizes at 4 iterations, with the MSE function value stabilizing at 0.00095. The main reason why PSO easily falls into a local optimum is that the learning factor is a fixed value, which leads to the algorithm not focusing on the optimal global particles in the later stage and thus very easily falling into local optimum solutions. The AI-HydSu algorithm focuses on the searchability of the particles themselves at the early stage of the algorithm search and focuses on the optimal global particles at the later stage; thus, it avoids falling into the local optimum. Through the experimental analysis, it can be seen that the AI-HydSu algorithm is faster and less likely to fall into the local optimum than the PSO algorithm, and its overall performance is better than that of the PSO algorithm.

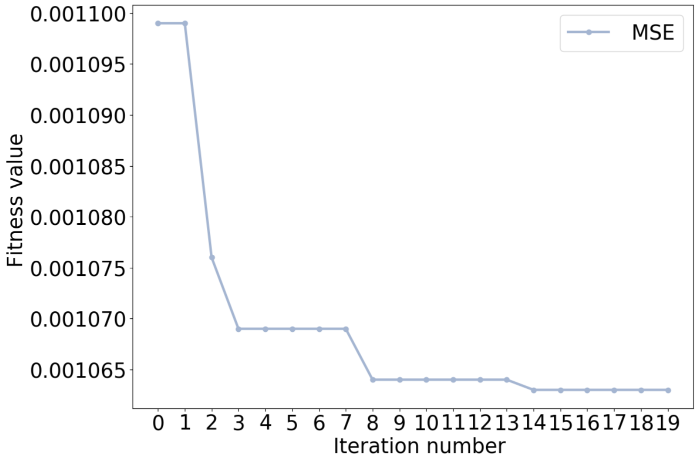

Figure 7. PSO-SVR fitness.

Figure 9 shows the relative error between the true value and the predicted value. From Figure 9, it can be seen that the maximum value of the AI-HydSu algorithm prediction relative error curve does not exceed 2%, and the prediction results are relatively stable, which further indicates that the AI-HydSu model achieves a good prediction accuracy.

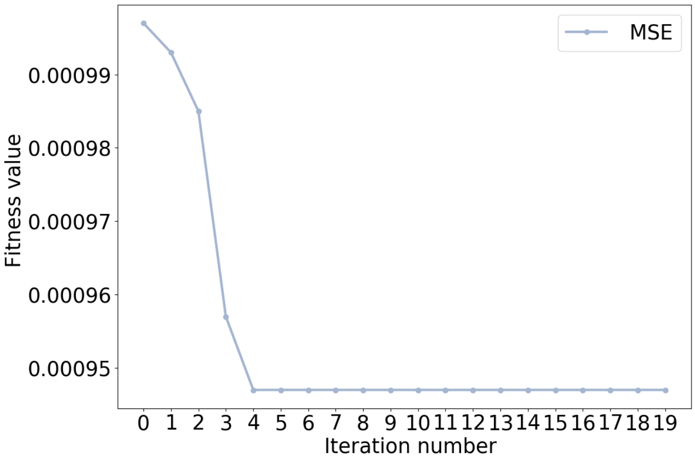

Figure 8. AI-HydSu fitness.

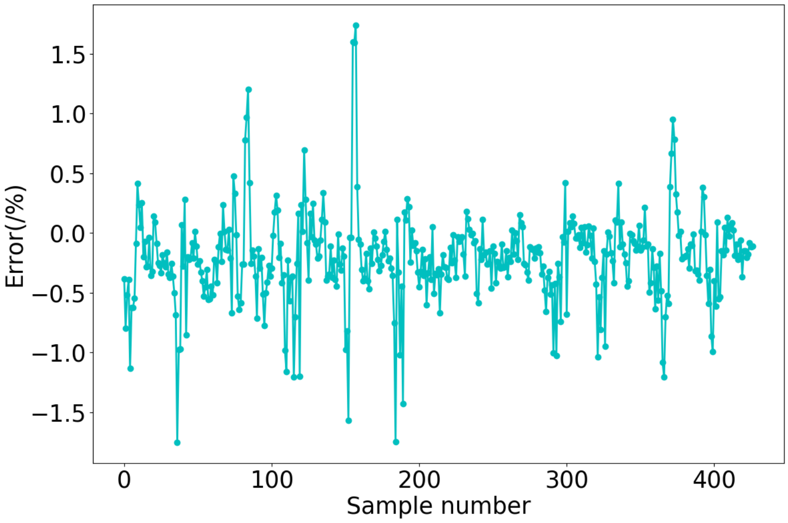

Figure 9. Relative error.

To verify the prediction performance of the proposed AI-HydSu-based DO prediction model, the backpropagation neural network (BPNN) [32, 33], PSO-SVR [34, 35], autoregressive integrated moving average (ARIMA), and long short-term memory (LSTM) algorithms were chosen to predict the DO time series of shrimp aquaculture waters (Figure 10), and the prediction results were evaluated by the mean absolute error (MAE) [36], root mean square error (RMSE) [37], and coefficient of determination R^2 [38].

$$MAE=\frac{1}{n}\sum_{i=1}^{n}\left|\hat{y}_i-y_i\right| \tag{4.1}$$

$$RMSE=\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\hat{y}_i-y_i\right)^2} \tag{4.2}$$

$$R^2 = 1 - \frac{\sum (y_i - \overset{\wedge}{y_i})^2}{\sum (y_i - \bar{y_i})^2} \tag{4.3}$$

Table 1. Analysis of the precision of the forecasting results in terms of the three evaluation indicators.

	Evaluation Index		
Model	Mean absolute error	Root mean square error	R^2
LSTM	0.0431	0.0369	0.836
ARIMA	0.0516	0.0427	0.796
BPNN	0.0431	0.0319	0.825
PSO-SVR	0.0267	0.0186	0.856
AI-HydSu	0.0221	0.0155	0.932

As shown in Table 1, the MAE and RMSE values of the AI-HydSu prediction model proposed in this paper were reduced by 48.7% and 51.4%, respectively, compared to the BPNN results under the same conditions. Compared with the LSTM prediction model, the MAE and RMSE of the proposed model were reduced by 48.7% and 58%, respectively. Compared with the PSO-SVR prediction model, the MAE and RMSE of the proposed model are reduced by 17.2% and 16.7%, respectively. Compared with the ARIMA prediction model, the MAE and RMSE of the proposed model were reduced by 57.2% and 63.7%, respectively. R^2 is an important statistic reflecting the goodness of fit of the model and is the ratio of the regression sum of squares to the total sum of squares. R^2 takes values between 0 and 1 and is unitless, and its numerical magnitude reflects the relative degree of the regression contribution, i.e., the percentage of the total variation in the dependent variable Y that is explained by the regression relationship. R^2 is the most commonly used index to evaluate the degree of merit of regression models; the larger R^2 is (i.e., the closer it is to 1), the better the fitted regression equation is. In Table 1, the R^2 of the AI-HydSu model proposed in this paper is closest to 1. The experiment shows that the AI-HydSu model has high accuracy in the prediction of DO concentration.

Figure 10 shows the comparison between the predicted and true value curves of the proposed AI-HydSu prediction model and other similar prediction models under the same conditions[39, 40], and it can be seen from this figure that the proposed model fits the true value curve more closely and does not have large deviation points and abrupt change points. The AI-HydSu model is applicable not only to DO data but also to other water quality parameters.

Figure 11 shows the temperature prediction of the AI-HydSu model applied to the water quality parameters, and it can be seen that the predicted values do not deviate much from the true values, and some of the curves almost overlap[41, 42], reflecting the superiority of the AI-HydSu model.

To verify the accuracy, stability, and generalizability of the AI-HydSu model, experiments were conducted on nine different datasets [43], with experimental data from nine ranches in China's Blue Economic Zone. Since the amount of data varies from ranch to ranch, the experiments all use 95% of the dataset as the training set and 5% as the test set and compare the real values with the predicted values. The experimental results are shown in Figure 12. From this figure, it can be seen that the AI-HydSu model achieves a better prediction effect for different training samples.

The AI-HydSu model was further validated against the RBFNN, LSTM, BPNN, and PSO-SVR model on nine different datasets, and the experimental results are shown in Figure 13. The stability and accuracy of the AI-HydSu model are experimentally proven to be better than other similar algorithms, and the average absolute errors in the different datasets are minimized.

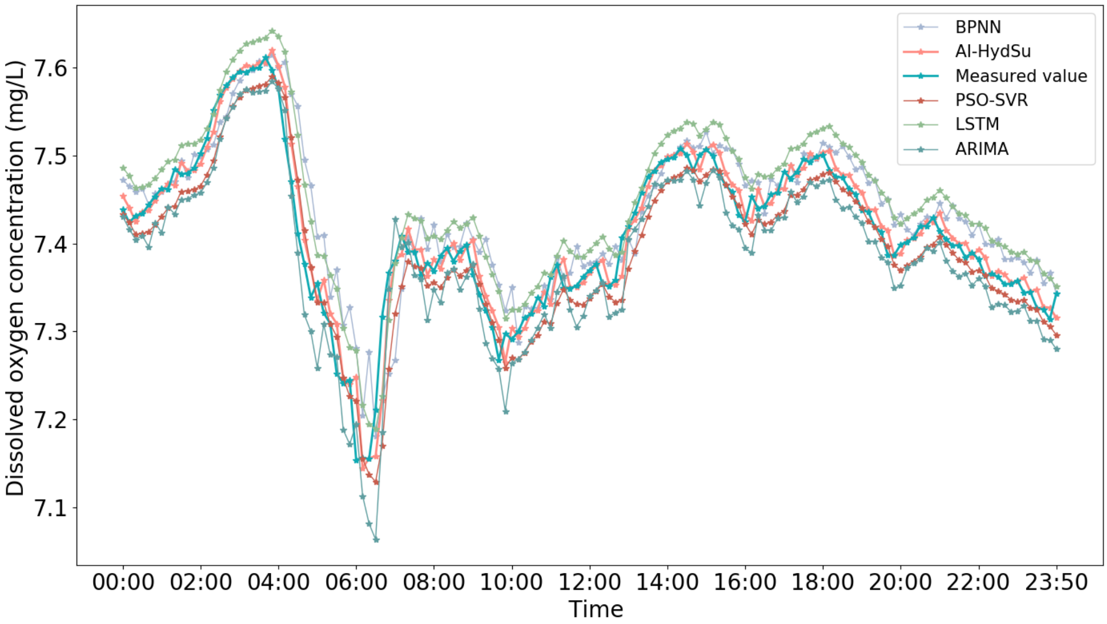

Figure 10. Comparison of different algorithm models.

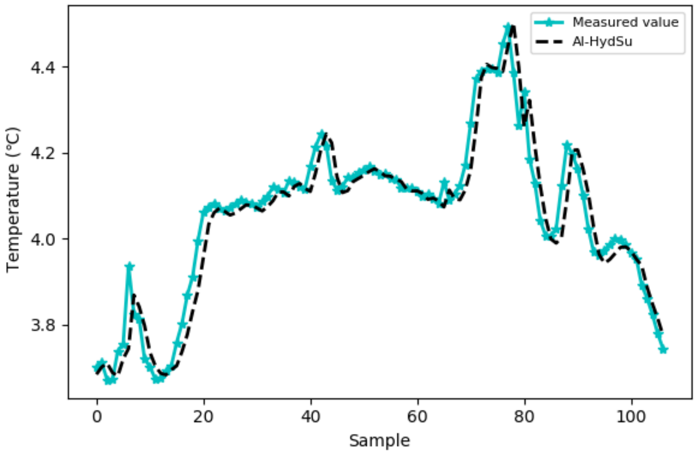

Figure 11. Temperature prediction.

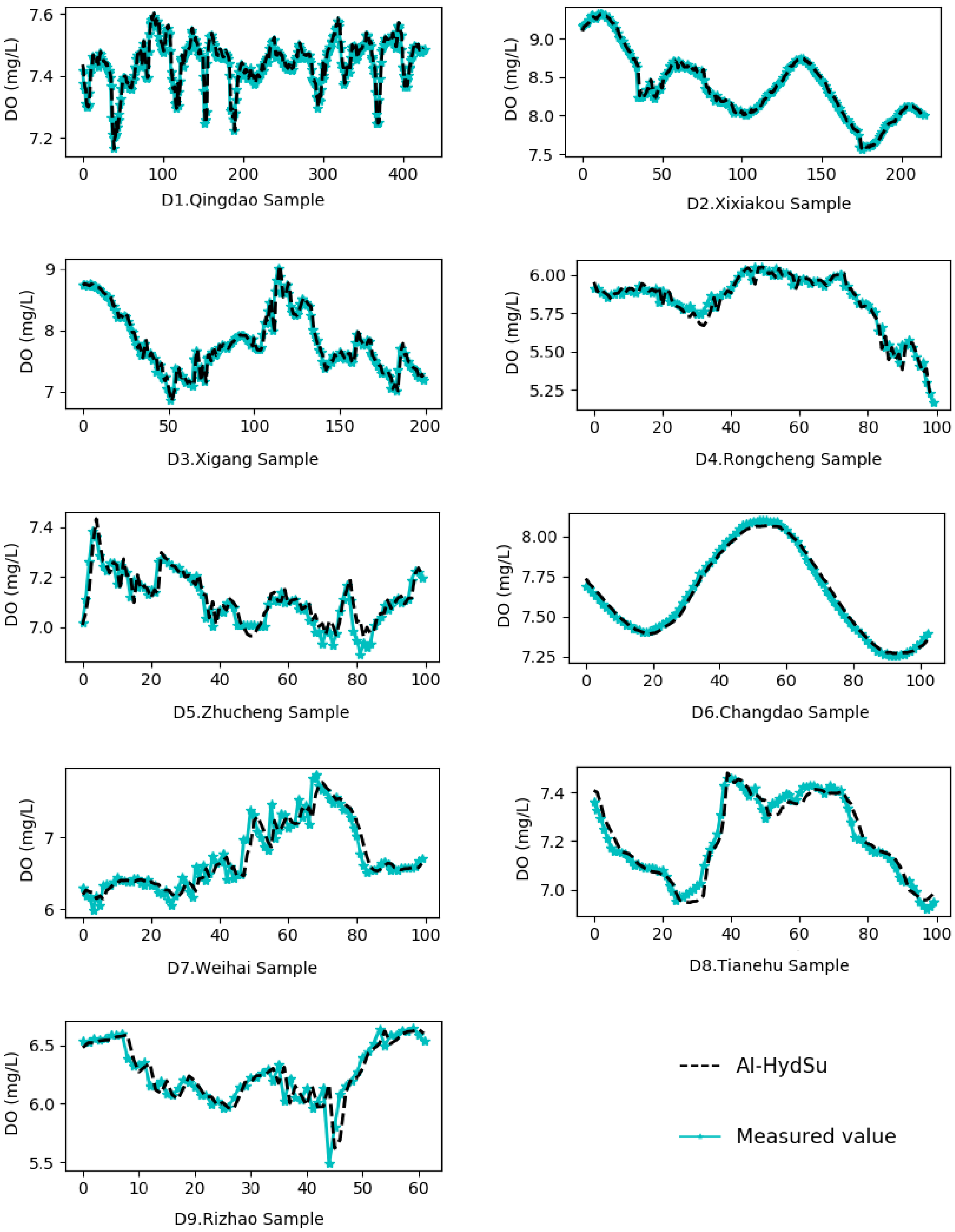

Figure 12. Comparison of the real and predicted values of different datasets.

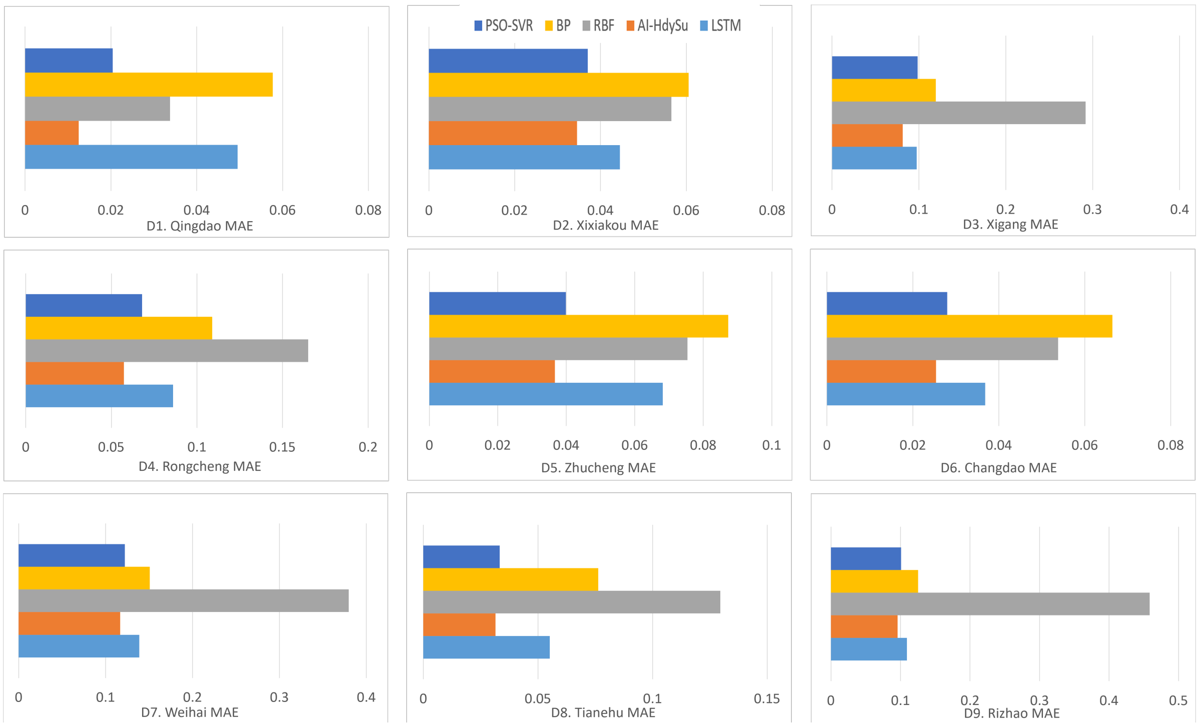

Figure 13. Validation of different datasets.

5. Conclusions

In this paper, a DO prediction model based on a PSO algorithm incorporating an SVM with nonlinear adaptive learning factors is proposed for accurate prediction of DO concentration. The results show that the AI-HydSu model has a better prediction accuracy on a small sample, which can effectively solve the problems of easily falling into a local optimum and slow convergence speed of the traditional PSO-SVR model. In this study, the DO concentration of each marine pasture in the Blue Economic Zone of China was studied, and the results of the AI-HydSu model were experimentally compared with those of the PSO-SVR model and other similar prediction methods. The experimental results show that the AI-HydSu prediction method can fit the training and testing data well and has the characteristics of smaller error fluctuations and more stability. The RMSE and MAE of the AI-HydSu model are smaller than those of the other algorithms. Overall, the proposed AI-HydSu model achieves a good prediction performance, has as strong generalizability and can provide accurate aquaculture fishing information services for the production of intensive fisheries.

There are many factors affecting DO, such as the water temperature, salinity, and pH. Future research will explore the relationship between DO and other water quality parameters in depth to build a more accurate DO prediction model.

Acknowledgments

This work was supported in part by the CERNET Innovation Project (NGII20180319) and in part by Yantai Science and Technology Innovation Development Project (2021YT06000715) and Key R&D Program of Shandong Province (Soft Science Project) (2020RKB01555).

References

1. F. Khademi, M. Akbari, S. M. Jamal, M. Nikoo, Multiple linear regression, artificial neural network, and fuzzy logic prediction of 28 days compressive strength of concrete, *Front. Struct. Civ. Eng.*, **11** (2017), 90–99.

2. R. Pino-Mejías, A. Pérez-Fargallo, C. Rubio-Bellido, Jesús A. Pulido-Arcas, Comparison of linear regression and artificial neural networks models to predict heating and cooling energy demand, energy consumption and CO_2 emissions, *Energy*, **118** (2017), 24–36.

3. P. Singh, P. Gupta, K. Jyoti, TASM: technocrat ARIMA and SVR model for workload prediction of web applications in cloud, *Cluster Comput.*, **22** (2019), 619–633.

4. Y. Wang, C. H. Wang, C. Z. Shi, B. H. Xiao, Short-term cloud coverage prediction using the ARIMA time series model, *Remote Sens. Lett.*, **9** (2018), 274–283.

5. L. Yang, H. Chen, Fault diagnosis of gearbox based on RBF-PF and particle swarm optimization wavelet neural network, *Neural Comput. Appl.*, **31** (2019), 4463–4478.

6. G. Renata, S. L. Zhu, S. Bellie, Forecasting river water temperature time series using a wavelet–neural network hybrid modelling approach, *J. Hydrol.*, **578** (2019), 124115.

7. A. M. Zador, A critique of pure learning and what artificial neural networks can learn from animal brains, *Nat. Commun.*, **10** (2019), 3770.

8. A. Shebani, S. Iwnicki, Prediction of wheel and rail wear under different contact conditions using artificial neural networks, *Wear*, **406–407** (2018), 173–184.

9. E. Olyaie, H. Z. Abyaneh, A. D. Mehr, A comparative analysis among computational intelligence techniques for dissolved oxygen prediction in Delaware River, *Geosci. Front.*, **8** (2017), 517–527.

10. O. Kisi , M. Alizamir, A. D. Gorgij, Dissolved oxygen prediction using a new ensemble method, *Environ. Sci. Pollut. Res. Int.*, **27** (2020), 9589–9603.

11. B. Raheli, M. T. Aalami, A. El-Shafie, M. A. Ghorbani, R. C. Deo, Uncertainty assessment of the multilayer perceptron (MLP) neural network model with implementation of the novel hybrid MLP-FFA method for prediction of biochemical oxygen demand and dissolved oxygen: a case study of Langat River, *Environ. Earth. Sci.* , **76** (2017), 503.

12. A. A. Masrur Ahmed, Prediction of dissolved oxygen in Surma River by biochemical oxygen demand and chemical oxygen demand using the artificial neural networks (ANNs), *J. King Saud Univ. Eng. Sci.*, **29** (2017), 151–158.

13. B. Keshtegar, S. Heddam, Modeling daily dissolved oxygen concentration using modified response surface method and artificial neural network: a comparative study, *Neural Comput. Appli.*, **30** (2018), 2995–3006.

14. A. Csábrági, S. Molnár, P. Tanos, J. Kovács, Application of artificial neural networks to the forecasting of dissolved oxygen content in the Hungarian section of the river Danube, *Ecol. Eng.*, **100** (2017), 63–72.

15. S. Heddam, O. Kisi, Extreme learning machines: a new approach for modeling dissolved oxygen (DO) concentration with and without water quality variables as predictors, *Environ. Sci. Pollut. Res.*, **24** (2017), 16702–16724.

16. I. Ahmad, M. Basheri, M. J. Iqbal, A. Rahim, Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection, *IEEE Access*, **6** (2018), 33789–33795.

17. X. Luo, D. Li, S. Zhang, Traffic Flow Prediction during the Holidays Based on DFT and SVR, *J. Sensors*, **2019** (2019), 6461450.

18. M. S. Ahmad, S. M. Adnan, S. Zaidi, P. Bhargava, A novel support vector regression (SVR) model for the prediction of splice strength of the unconfined beam specimens, *Constr. Build. Mater.*, **248** (2020), 118475.

19. Y. Zhang, H. Sun, Y. Guo, Wind power prediction based on PSO-SVR and grey combination model, *IEEE Access*, **7** (2019), 136254–136267.

20. E. Dodangeh, M. Panahi, F. Rezaie, S. Lee, D. T. Bui, C. W. Lee, et al., Novel hybrid intelligence models for flood-susceptibility prediction: Meta optimization of the GMDH and SVR models with the genetic algorithm and harmony search, *J. Hydrol.*, **590** (2020), 125423.

21. Y. Xiang, L. Gou, L. He, S. Xia, W. Wang, A SVR–ANN combined model based on ensemble EMD for rainfall prediction, *Appl. Soft Comput.*, **73** (2018), 874–883.

22. M. Panahi, N. Sadhasivam, H. R. Pourghasemi, F. Rezaie, S. Lee, Spatial prediction of groundwater potential mapping based on convolutional neural network (CNN) and support vector regression (SVR), *J. Hydrol.*, **588** (2020), 125033.

23. J. Kennedy, Particle swarm optimization, in *Proceedings of ICNN'95 - International Conference on Neural Networks* (R. Eberhart), (1995), 1942–1948.

24. X. Liang, T. Qi, Z. Jin, W. Qian, Hybrid support vector machine optimization model for inversion of tunnel transient electromagnetic method, *Math. Biosci. Eng.*, **17** (2020), 3998.

25. M. Rosendo, A hybrid Particle Swarm Optimization algorithm for combinatorial optimization problems, in *IEEE Congress on Evolutionary Computation* (A. Pozo), (2010), 1–8.

26. C. F. Wang, K. Liu, A novel particle swarm optimization algorithm for global optimization, *Comput. Intell. Neurosci.*, **2016** (2016), 9482073.

27. D. L. Donoho, I. M. Johnstone, Ideal spatial adaptation by wavelet shrinkage, *Biometrika*, **81** (1994), 425–455.

28. C. L. Wang, C. L. Zhang, P. T. Zhang, Denoising algorithm based on wavelet adaptive threshold, *Phys. Procedia*, **24** (2012), 678–685.

29. S. Tomassini, A. Strazza, A. Sbrollini, I. Marcantoni, M. Morettini, S. Fioretti, et al., Wavelet filtering of fetal phonocardiography: A comparative analysis, *Math. Biosci. Eng.*, **16** (2019), 6034.

30. F. M. Bayer, A. J. Kozakevicius, R. J. Cintra, An iterative wavelet threshold for signal denoising *Signal Process.*, **162** (2019), 10–20.

31. H. Liu, L. Chang, C. Li, C. Yang, Particle swarm optimization-based support vector regression for tourist arrivals forecasting, *Comput. Intell. Neurosci.*, **2018** (2018), 6076475.

32. A. T. C. Goh, Back-propagation neural networks for modeling complex systems, *Artif. Intell. Eng.*, **9** (1995), 143–151.

33. T. L. Lee, Back-propagation neural network for the prediction of the short-term storm surge in Taichung harbor, Taiwan, *Eng. Appl. Artif. Intel.*, **21** (2008), 63–72.

34. W. Hu, PSO-SVR: A Hybrid Short-term Traffic Flow Forecasting Method, in *2015 IEEE 21st International Conference on Parallel and Distributed Systems (ICPADS)* (L. Yan), (2015), 553–561.

35. X. P. Hu, X. D. Dong, B. H. Yu, Method of optimal design with SVR-PSO for ultrasonic cutter assembly, *Procedia CIRP*, **50** (2016), 779–783.

36. K. Roosa, R. Luo, G. Chowell, Comparative assessment of parameter estimation methods in the presence of overdispersion: a simulation study, *Math. Biosci. Eng.*, **16** (2019), 4229.

37. F. M. Butt, L. Hussain, A. Mahmood, K. J. Lone, Artificial Intelligence based accurately load forecasting system to forecast short and medium-term load demands, *Math. Biosci. Eng.*, **18** (2021), 400.

38. H. Li, J. Tong, A novel clustering algorithm for time-series data based on precise correlation coefficient matching in the IoT, *Math. Biosci. Eng.*, **16** (2019), 6654.

39. S. Zhu, M. Ptak, Z. M. Yaseen, J. Dai, B. Sivakumar, Forecasting surface water temperature in lakes: A comparison of approaches, *J. Hydrol.*, **585** (2020), 124809.

40. J. Quilty, J. Adamowski, A stochastic wavelet-based data-driven framework for forecasting uncertain multiscale hydrological and water resources processes, *Environ. Model Softw.*, **130** (2020), 104718.

41. T. J. Glose, C. Lowry, M. B. Hausner, Examining the utility of continuously quantified Darcy fluxes through the use of periodic temperature time series, *J. Hydrol.*, (2020), 125675.

42. F. Kang, J. Li, J. Dai, Prediction of long-term temperature effect in structural health monitoring of concrete dams using support vector machines with Jaya optimizer and salp swarm algorithms, *Adv. Eng. Softw.*, **131** (2020), 60–67.

43. F. Kang, A. M. ASCE, J. Li, Displacement model for concrete dam safety monitoring via Gaussian process regression considering extreme air temperature, *J. Struct. Eng.*, **146** (2020), 05019001.