

Research on score prediction in sports competitions——based on PSO-BP neural network and ARIMAX hybrid prediction model

Longhui Shi*

Department of Information Science and Technology Yenching Institute of Technology, Langfang, China, 065201

* Corresponding Author Email: shi26790126@163.com

Abstract. Predicting match results has always been an important problem in the field of sports. Athletes and coaches can benefit from accurately predicting the path of a match, allowing them to strategize ahead of time and take control during the match. The PSO-BP neural network combines the advantages of particle swarm optimization and backpropagation neural networks to improve the accuracy of the model in processing complex nonlinear data interactions. ARIMAX is ideal for evaluating data that exhibits linear trends and seasonality. The model combines an autoregressive synthetic moving average component with external variables to account for linear trends, seasonality, and external influences in the data. Through the data in https://github.com/JeffSackmann/tennis_wta, this article establishes a combination of two models to predict game scores. Variables include advantage, turnover, lead, distance traveled, and serve. Using K-fold cross-validation, the accuracy rates of the PSO-BP neural network and ARIMAX prediction model were measured to be 82.89% and 80.12% respectively. Then bringing this model into the basketball game, we get its prediction accuracy: 77.12% and 76.21%. A random forest analysis method was used to evaluate the significance of each variable in prediction. The significance of each indicator is as follows: 0.33, 0.26, 0.17, 0.15 and 0.09.

Keywords: Tennis Score, PSO-BP Neural Network, ARIMAX, Random Forest

1. Introduction

Forecasting tennis match outcomes has consistently captivated the interest of several researchers and fans. Prediction accuracy depends on both the reliability of the prediction model and the researcher's profound understanding of the game's regulations and athletes' performance. Due to technological advancements, prediction models have grown increasingly precise. These technologies can analyze extensive game data to glean significant insights and improve the accuracy of predictions. Tennis match result forecasts will become more precise, offering a scientific reference for athletes' training, game strategy, and fans' viewing experience.

Ana Šarčević [1] et al proposed a combinatorial approach in predicting the outcome of tennis matches. This article proposes a more efficient alternative, using a combination method that relies on the binomial distribution, which achieves the same accuracy as the recursive method. Wei Gu [2] et al proposed Predicting the Outcome of a Tennis Tournament: Based on Both Data and Judgments. This article is based on a combination of data and judgment, establishing an ANP Decision-making Network to predict tennis matches.

Scholars have conducted a large amount of research on the prediction of tennis match results and applied the research results to specific practices, which has certain significance for the gaming industry and the audience's viewing experience. At present, scholars have done little research on specific scores during tennis matches. Based on the research on existing match data, this paper establishes a prediction method that combines PSO-BP neural network and ARIMAX models to predict scores during the match. Then, random forest is used to evaluate the importance of each feature in the prediction results.

2. The model-building process

2.1. Establish PSO-BP neural network prediction model

The training method of the BP Neural Network involves forward propagation of the signal and back propagation of the error [3]. The signal travels from the input layer of the neural network, through the hidden layer, to the output layer, where the output signal and error signal are produced. If the error signal satisfies the criteria, the calculation stops; otherwise, the signal is sent on to back propagation. The weights and thresholds are modified sequentially in each layer, and the network parameters are updated using the gradient descent method. The training process of the BP Neural Network is as follows.

(1) Define input and output:

First, clearly define the inputs and outputs of the problem. The input is the data you want the neural network to learn from, and the output is the target you want the neural network to predict.

(2) Build the network structure:

Determine the structure of the neural network, including the number of nodes in the input layer, the number of nodes in the output layer, and the number and number of nodes in the hidden layers in between.

(3) Initialize weights and thresholds:

During network construction, weights and thresholds are usually initialized to random values. These random initial values will be continuously adjusted and optimized during the subsequent training process, allowing the neural network to adapt to the data and produce accurate output.

(4) Forward propagation calculation.

Calculate the hidden layer output H_j and the output layer output O_k by forward propagating the determined network.

$$H_j = f\left(\sum_{i=1}^n W_{ij} + a_k\right), j = 1, 2, \dots, L \quad (1)$$

$$O_k = \sum_{j=1}^n W_{jk} H_j + b_k, k = 1, 2, \dots, M \quad (2)$$

Where W_{ij} is the weight from the input layer to the hidden layer, a_j is the hidden layer threshold, L is the number of hidden layer nodes, W_{jk} is the weight from the hidden layer to the output layer, b_k is the output layer threshold.

(5) Error calculation.

The prediction error is the difference between the expected value Y_k and the predicted value.

$$e_k = Y_k - O_k, k = 1, 2, \dots, M \quad (3)$$

(6) Back propagation:

Back propagation is the core step in training neural networks. By comparing the predicted output with the actual output, the gradient of the loss function with respect to weights and thresholds is calculated.

(7) Repeat training:

The steps of back propagation are repeated multiple times to train the neural network by continuously updating the weights and thresholds. During the training process, batches or samples of training data are used and iteration is continued until the set stopping condition is reached.

(8) Verification and testing:

During the training process, the data set is usually divided into a training set, a validation set, and a test set. The training set is used to train the neural network, the validation set is used to adjust hyperparameters and prevent over fitting, and the test set is used to evaluate the performance of the model.

(9) Model application:

After the training is completed, the trained neural network model is used for prediction tasks. Input new input data into the model and obtain the output results through forward propagation.

The performance of the generic BP neural network is significantly influenced by the starting weights and thresholds[4], which are either artificially set or randomly generated. It is simple to get stuck in a local optimal solution, leading to poor convergence speed. The particle swarm algorithm is utilized to optimize the weights and thresholds in order to achieve network optimization by obtaining the ideal initial weights and thresholds.

The PSO method was initially introduced in 1995 [5]. It is a form of evolutionary algorithm. The main concept is utilizing the information exchange among group members to consistently update their position and speed data, enabling the group to navigate the problem-solving environment together. The progression from chaos to organization occurs, leading to the eventual attainment of the best solution to the problem. Formulas (4) and (5) represent the update formulas for particle speed and position, respectively. The particle's position and speed will be updated iteratively until the stopping condition is met.

$$v_{i(t+1)} = \omega v_{i(t)} + c_1(pbest_i - x_{i(t)}) + c_2 r_2(gbest - x_{i(t)}) \quad (4)$$

$$x_{i(t+1)} = x_{i(t)} + v_{i(t+1)} \quad (5)$$

where, $v_{i(t)}$ is the velocity vector of the i particle at time t . ω is the inertia weight, which is used to control the impact of the speed at the previous moment on the speed at this moment. c_1 and c_2 are acceleration factors, used to control the impact of individual experience and group experience on speed. r_1 and r_2 are random numbers in the range $[0,1]$. $pbest_i$ is the individual optimal position of the i particle. $gbest$ is the global optimal position of the group. $x_{i(t)}$ is the position vector of the i particle at time t .

2.2. Establish ARIMAX prediction model

ARIMA is the current classic time series forecasting approach based on a review of numerous literatures[6]. This method utilizes the ideas of analogy (auto regression) and inertia (moving average) to provide a complete and reliable analysis.

The meaning of each letter of ARIMA:

AR (autoregressive) part: represents the correlation between the current observation and the observation at some point in the past. The AR model uses an autoregressive term to predict the current value by calculating the linear relationship between the observed value at a past time point and the current observed value.

I (difference) part: represents the difference operation, used to stabilize the original time series. The first difference of the time series can be obtained by subtracting the value at the previous time point. This process is repeated until a stationary time series is obtained.

MA (moving average) part: represents the correlation between the current observation and the error term at some point in time in the past. The MA model uses a moving average term to predict the current value by calculating the linear relationship between the random error at past time points and the current observed value.

ARIMA analysis and forecasting method — theoretical model:

$$x_t = \sum_{i=1}^p \phi_i x_{t-i} + a_t - \sum_{j=1}^q \varphi_j a_{t-j} \quad (6)$$

$$y_t \sim I(d) \quad (7)$$

$$x_t = \Delta^d y_t \quad (8)$$

Where, p is the autoregressive coefficient, q is the moving average coefficient, d is the difference order, i is the autoregressive parameter, and j is the moving average parameter.

ARIMAX[7], an extension of the classic ARIMA model, has emerged as the current leading time series forecasting approach based on a review of numerous literatures. This model builds upon the principles of analogy (auto regression) and inertia (moving average) while introducing external variables (regressors) to provide a more comprehensive and reliable analysis. These external variables, which are typically highly correlated with the predicted variable, allow ARIMAX to capture a wider range of patterns and trends in the data, leading to more accurate and nuanced predictions.

ARIMAX analysis and forecasting method — theoretical model:

Given that the response sequence $\{y_t\}$ and the input variable sequence $\{x_{1t}\}, \{x_{2t}\}, \dots, \{x_{kt}\}$ are both stationary, create a regression model using these sequences:

$$y_t = \mu + \sum_{i=1}^k \frac{\Theta_i(B)}{\Phi_i(B)} B^{li} x_{it} + \varepsilon_t \quad (9)$$

Because $\{y_t\}$ and the input variable sequence $\{x_{1t}\}, \{x_{2t}\}, \dots, \{x_{kt}\}$ are both stationary sequences, the linear combination of the stationary sequence is still stationary, and the residual sequence $\{\varepsilon_t\}$ obtained by model fitting is also stationary, so the ARMA model can be used to continue to extract the residual sequence $\{\varepsilon_t\}$ of useful information. The resulting final model is:

$$\begin{cases} y_t = \mu + \sum_{i=1}^k \frac{\Theta_i(B)}{\Phi_i(B)} B^{li} x_{it} + \varepsilon_t \\ \varepsilon_t = \frac{\Theta(B)}{\Phi(B)} a_t \end{cases} \quad (10)$$

The above dynamic autoregressive moving regression model. That is, is the ARIMAX model. Where $\Phi(B)$ represents the autoregressive coefficient polynomial of the residual sequence; $\Theta(B)$ is the moving average coefficient polynomial of the residual sequence; a_t is the zero-mean white noise sequence.

3. Results

3.1. data preparation

The data in this article comes from a piece of data in JeffSackmann/tennis_wta: WTA Tennis Rankings, Results, and Stats (github.com), which includes the winning and losing of each game, the mistakes of both sides, the number of leading games, etc. Therefore, Through data analysis, this article has screened out a series of indicators closely related to the prediction goals. In order to further refine these indicators and eliminate redundancy, principal component analysis (PCA)[8] was used for dimensionality reduction, and finally high-quality indicators were extracted, as shown in Table 1.

Table 1. Factors affecting game scores

Influencing factors	
Dominance	Mistake
Leading games	Running distance
Server	

3.2. Application of PSO-BP neural network and ARIMAX prediction model

This research integrates the ARIMAX model with the PSO-BP neural network prediction model using the processed data. The ARIMAX model incorporates linear trend, seasonal factors, and regression term capture capabilities, while the PSO-BP neural network offers powerful nonlinear

mapping and adaptability. This prediction method can thoroughly analyze data and provide reliable decision support for practical applications.

The prediction results are shown in Figure 1 below:

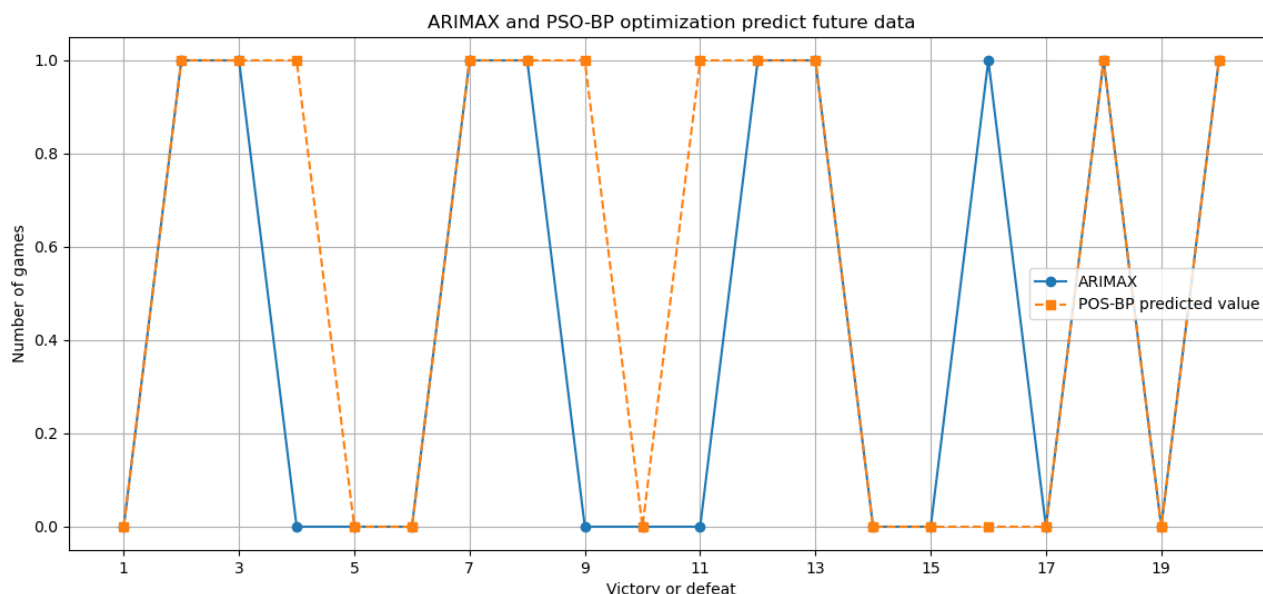


Figure 1. ARIMAX and PSO-BP optimization predict future data

This article will assess the predictive capability of the PSO-BP neural network and ARIMAX model on match outcomes by utilizing real and representative tennis match data and employing the K-fold Cross-validation technique[9].

K-fold Cross-validation technique Partition the historical match data into k segments, with each segment holding an equal number of matches. We then conduct k training and testing iterations. During each cycle, we choose k-1 subsets of the data to serve as the training set for model training, while the remaining subset is utilized as the validation set to assess the model's performance on new data. Repeat the aforementioned processes k times, choosing a distinct portion as the test set on each iteration. Compute the average accuracy from all k validation runs to determine the model's accuracy. The results are shown in Figure 2:

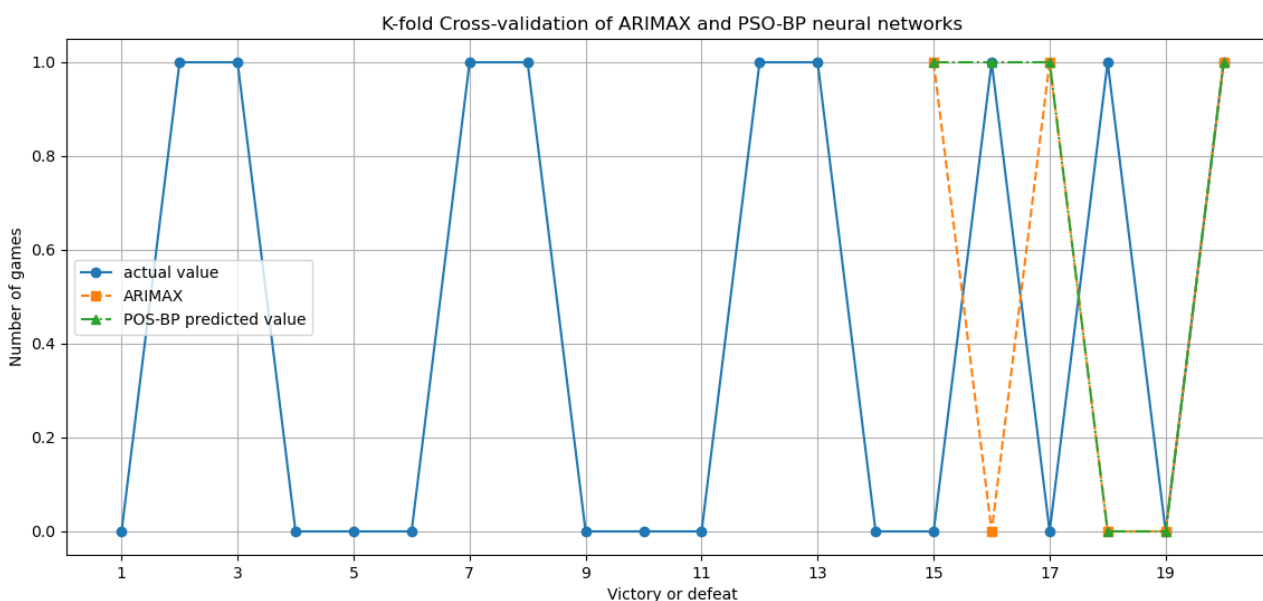


Figure 2. K-fold Cross-validation results

The accuracy of the obtained PSO-BP Neural Network and ARIMAX model are: 82.89% and 80.12% respectively.

3.3. Application in basketball games

In order to deeply evaluate the prediction performance and scalability of the established PSO-BP neural network and ARIMAX prediction model in sports competitions, this article plans to apply these two prediction models to the analysis of basketball events.

The basketball game data comes from neilmj/BasketballData (github.com). The data has been strictly collected, cleaned and standardized to ensure the accuracy and consistency of the data. In order to verify the accuracy of the PSO-BP neural network and ARIMAX prediction model in basketball game prediction, we introduced these data into the prediction model and conducted strict verification through K-fold Cross-validation. In order to display the prediction effect more intuitively, we plotted the test results into a chart (as shown in Figure 3).

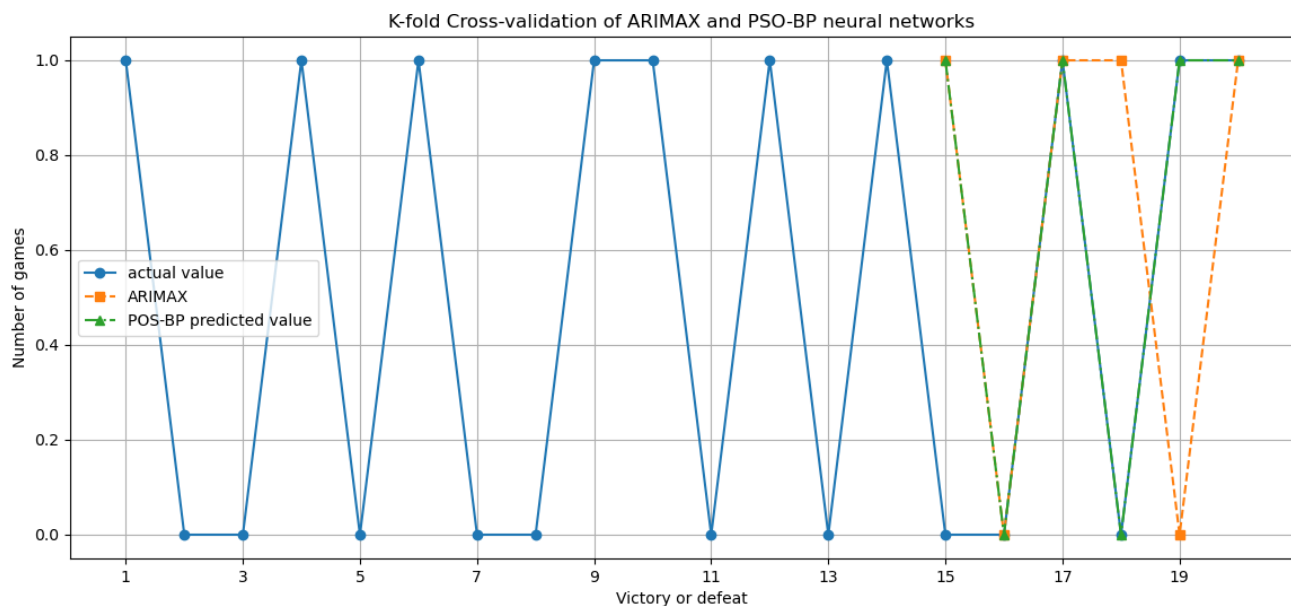


Figure 3. K-fold cross-validation applied to basketball games

The accuracy of the obtained PSO-BP Neural Network and ARIMAX model are: 77.12% and 76.21% respectively.

From this we can know that the prediction model shows strong generalization ability. Their ability to adapt to different data sets and handle complex relationships within the data makes them highly versatile.

3.4. Random forest analysis importance

This article use a random forest classifier to analyze the significance of each feature in the dataset Random forest, a robust machine learning method, can assess the impact of each feature on prediction outcomes [10]. This allows the random forest classifier to determine the significance of the data.

(1) Data import and preparation:

Determine features (independent variables) and target variables (dependent variables)

(2) Data splitting:

The training set is utilized for model training, whereas the test set is employed for model performance evaluation. Their ratio is 4:1.

(3) Random forest classifier initialization:

First initialize a random forest classifier, confirm the number of decision trees constructed, and determine the repeatability of the results.

(4) Model training:

Train a random forest classifier using the training set.

(5) Feature importance analysis:

Get the importance of each feature to the target variable, sort and print it in order to understand which features play a more important role in prediction.

(6) Model evaluation:

Use the trained model to make predictions on the test set, and calculate the accuracy of the model on the test set, that is, the proportion of samples correctly predicted by the model.

The analysis results are shown in Figure 4:

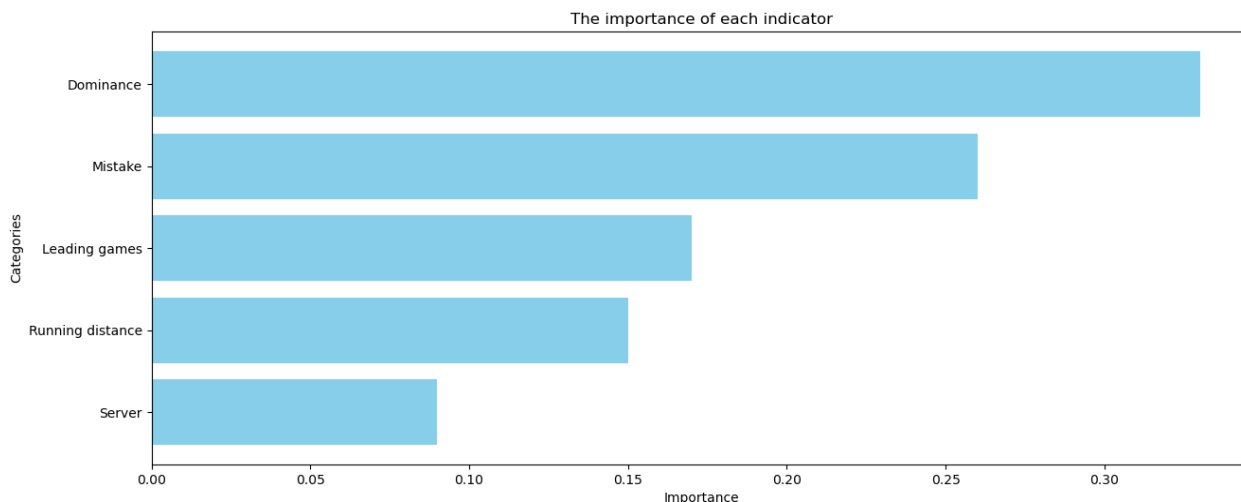


Figure 4. The importance of each indicator

The figure illustrates that Dominance has a prediction importance of 0.33. The significance of Mistake in prediction is 0.26. The Leading games have a predictive significance of 0.17. The significance of Running distance in prediction is 0.15. The Server significance in prediction is 0.09.

4. Conclusions

This paper establishes a prediction model combining PSO-BP neural network and ARIMAX for the prediction of tennis scores. The PSO-BP neural network excels in nonlinear mapping and flexibility, while the ARIMAX model is ideal for analyzing data with linear trends and seasonality. Using K-fold cross-validation, the accuracy rates are 82.89% and 80.12% respectively. And applied it to basketball games, the accuracy rates obtained were 77.12% and 76.21%. Then random forest was used to analyze the importance of predictive indicators. The significance of each indicator is as follows: 0.33, 0.26, 0.17, 0.15 and 0.09.

Many factors come into play when predicting the outcome of a tennis match. Insufficient data and training samples in this article will lead to errors in prediction results. There is potential for further enhancement. As sample data and model training become more extensive, the accuracy of the model will improve, leading to more reasonable and scientifically sound conclusions.

References

- [1] Šarčević A, Vranić M, Pintar D. A combinatorial approach in predicting the outcome of tennis matches[J]. *International Journal of Applied Mathematics and Computer Science*, 2021, 31(3): 525-538.
- [2] Gu W, Saaty T L. Predicting the outcome of a tennis tournament: Based on both data and judgments[J]. *Journal of Systems Science and Systems Engineering*, 2019, 28: 317-343.
- [3] Cui, Kai, and Xiang Jing. "Research on prediction model of geotechnical parameters based on BP neural network." *Neural Computing and Applications* 31 (2019): 8205-8215.
- [4] Rumelhart, David E., Geoffrey E. Hinton, and Ronald J. Williams. "Learning representations by back-propagating errors." *nature* 323.6088 (1986): 533-536.
- [5] Kennedy, James, and Russell Eberhart. "Particle swarm optimization (PSO)." *Proc. IEEE international conference on neural networks, Perth, Australia*. Vol. 4. No. 1. 1995.
- [6] Alabdulrazzaq, Haneen, et al. "On the accuracy of ARIMA based prediction of COVID-19 spread." *Results in Physics* 27 (2021): 104509.

- [7] Yang Shipeng, Jiang Wei, Chen Zhenzhen. Research on the growth forecast of industrial added value in Zhejiang - Empirical test based on ARMAX and neural network hybrid model [J]. *Statistical Science and Practice*, 2022(12):14-19.
- [8] Li Bingrui, Li Rongchang. Fault diagnosis method for papermaking wastewater treatment process based on PCA-PSO-SVM [J]. *Automation Application*, 2023(21):244-247.
- [9] Wang, Shuwei, et al. "Application of Deep Neural Network to Study the Sleep Stage Scoring on the Polysomnography." *Biophysics* 7 (2019): 11-25.
- [10] Sun, Shuping, et al. "Application of random forest algorithm in heart sound classification." *Computer Science and Application* 10.4 (2020): 591-600.