Journal Pre-proof

Analysis of earnings forecast of blockchain financial products based on particle swarm optimization

Wenyou Gao, Chang Su

PII: S0377-0427(20)30015-7
DOI: https://doi.org/10.1016/j.cam.2020.112724
Reference: CAM 112724

To appear in: *Journal of Computational and Applied Mathematics*

Received date : 30 June 2019
Revised date : 6 January 2020

Please cite this article as: W. Gao and C. Su, Analysis of earnings forecast of blockchain financial products based on particle swarm optimization, *Journal of Computational and Applied Mathematics* (2020), doi: https://doi.org/10.1016/j.cam.2020.112724.

Analysis of Earnings Forecast of Blockchain Financial Products based on Particle Swarm Optimization

Wenyou Gao
School of Economics and Management, Changchun University of Science and Technology, Changchun 130022, China
[e-mail: gaowenyoujp@126.com]

Chang Su*
School of Foreign Language, Harbin Institute of Technology, Harbin 150000, China
[*Corresponding author: Chang Su, e-mail: suchangchina@163.com]

Abstract: The purpose of this study is to solve the problems of large number of iterations, limitations and poor fitting effect of traditional algorithms in predicting the yield rate of blockchain financial products. In this study, bitcoin yield rate is taken as the research object, and data from June 2, 2016 to December 30, 2018 are collected, totaling 943 pieces. The BP neural network, support vector regression machine algorithm and particle swarm optimization least square vector algorithm are respectively adopted to carry out model simulation and empirical analysis on the collected data, and it is concluded that particle swarm optimization least square vector algorithm has the best fitting effect. Subsequently, the Ethereum (ETH) yield rate is selected as the research object, and the model simulation and empirical analysis are carried out on it, which verifies that the optimized algorithm have better prediction and fitting on the time series. The results show that the particle swarm optimization algorithm among the three algorithms mentioned in this research has the best prediction effect. Therefore, the results of this study have a good fitting effect on the prediction of the yield rate of blockchain financial products, have a good guiding effect on the investors of blockchain financial products, and have a good guiding significance for the study of the yield rate of China's blockchain financial products.

Keywords: particle swarm optimization; blockchain; financial product; earnings

1.Introduction

Since the beginning of the 20th century, the global economy has developed rapidly, and many factors that hinder economic development have been overwhelmed by the fast-growing economy. The financial market has developed rapidly. The financial market is the medium of economic development. It not only regulates the resource allocation of the entire economy and society, but also plays a vital role in economic development [1]. Since 2013, with the development of the Internet, many Internet financial products (such as Yu'E Bao, Baidu Financial Management, etc.) have been born, which has aroused the concern of the whole society. At present, emerging Internet finance with global influence includes crowdfunding, P2P, block chain and digital currency, which will play a crucial role in the future of global financial market [2].

Blockchain technology is an application under a new scenario developed based on computer technology, which has the characteristics of encryption algorithm, consensus mechanism, distributed node for data storage and point-to-point transmission. Consensus mechanism is an important part of blockchain, which uses mathematical algorithm to establish trust and obtain interests between different nodes in the blockchain system [3]. As the first successful application of blockchain technology, bitcoin has become a hot topic in blockchain research [4]. Under the rapid rise of blockchain finance, many domestic financial institutions and academic fields have studied and explored it. Under the limitations of traditional methods, many researchers begin to use machine learning and other methods to process and quantify investment analysis of blockchain financial data. In the process of exploring data, after a large number of empirical analysis, the fruitful results are finally gotten. [5]. First, the model is trained by using time series analysis and learning historical input and output data. Then the new independent variable is input into the model to obtain the prediction of future data [6]. But for some non-linear and non-stationary financial data, the prediction effect of time series is not satisfactory. As an important component of artificial intelligence technology, data mining shows superior performance in model training and learning [7]. After using neural network to predict financial data, many scholars find that it has a good learning ability for non-linear functions, and begin to apply neural network to the prediction and fitting of financial data such as stocks. However, neural network learning algorithm needs a large number of samples, and its generalization ability is not ideal. Therefore, support vector machines based on statistical learning theory, a new technology, have begun to enter the field of view of many researchers. Because support vector machine (SVM) theory is rigorous, which can reduce the problem of over-fitting, many investors used to do quantitative investment analysis. As a kind of machine learning algorithm, support vector machine itself also has its shortcomings. However, its application in the field of finance is still worth further exploration.

Under the characteristics and advantages of the above mentioned blockchain technology, its application in the financial industry has brought great improvement space and development opportunities to the financial industry. As the first successful application of bitcoin blockchain technology, investors are most concerned about its return and risk. Therefore, it is most representative to study the return efficiency prediction of bitcoin. And the prediction of the return rate of blockchain financial products based on the particle swarm optimization algorithm is mainly studied in this research. Firstly, the prediction effect of BT neural network algorithm, support vector regression machine on the rate of return of bitcoin is compared. Then, the prediction effect of BT neural network algorithm, support vector regression machine and particle swarm optimization least-squares support vector machine model on the yield rate of blockchain financial products is verified with the same type of financial products, namely ETH.

2.Literation review

With the rapid rise of blockchain finance, the model innovation of blockchain finance generates a strong impact on the traditional model of financial products, which has successfully attracted the attention of relevant scholars at home and abroad on the research of blockchain financial products. Due to limitations of traditional research methods, with the development of artificial intelligence algorithms, relevant scholars at home and abroad began to use mathematical algorithms to process and analyze quantitative financial data. Kiktenko et al. (2018) adopted the method of reducing dimensions to separate several major variables from many characteristic variables, and then used SVM algorithm for classification and identification. The results showed that this method could significantly improve the measurement accuracy [8]. Hussein et al. (2017) predicted the Taiwan stock market with support vector machine based on GA algorithm [9]. Mousavi et al. (2017) applied the POS algorithm model in solving the optimal securities portfolio model and obtained a good effect [10]. Huang et al. (2017) respectively used BP neural network, linear discriminant analysis, SVM and secondary discriminant analysis to conduct prediction analysis on nikkei 225 index. The results showed that support vector machine had the best prediction effect among the four methods [11]. Boubaker et al. (2017) used GARCH model and EWMA method to obtain better results when studying the volatility of the benchmark interest rate yield issued by the central bank [12]. Zhu et al. (2017) used ARIMA model and GARCH model to predict interest rates of central bank, and compared the forecasting ability of the two models [13]. Jin et al. (2017) adopted a variety of research methods to predict the 7-day government securities repurchase rate of Shanghai stock exchange, and the research results showed that the addition of GARCH model had a better effect on the fitting effect [14]. Li et al. (2017) used VAR and single-equation regression model to study the relationship between yield rate and future development trend. The results showed that the fluctuation of interest rate was directly proportional to its future trend [15]. Zheng et al. (2017) adopted ARIMA model when predicting the overnight lending rate, and the results showed that it had a good prediction ability [16]. Zhao (2017) and Tang (2018) improved SVM algorithm by combining support vector machine and wavelet function, and the results showed that this method could better predict the trend change of stock market [17, 18]. Ren (2018) used two models, that is, ARIMA model and GARCH model, to study the prediction of bond repo rate, and the results showed that the prediction effect of ARIMA model was better than that of GARCH model [19]. Havangi et al. (2017) studied the risk of marketization of interest rates based on market interest rates [20]. Martínez et al. (2017) established a personal credit system by studying the historical borrowing conditions of investors and financiers on the online financial platform of the Internet, and improved the information symmetry between borrowers and lenders, thus promoting the development of online finance [21].

From the development of the above research, the algorithm model is very popular in the field of Internet financial products research, while the research on blockchain financial products is relatively rare. Therefore, this study aims to forecast the yield rate of blockchain financial products based on particle swarm optimization. It is hoped to solve the problems of large number of iterations, limitations and poor fitting effect of traditional algorithms in predicting the yield rate of blockchain financial products.

3.Proposed method

3.1 BP neural network

BP neural network is an artificial intelligence method based on human neuron structure to simulate the memory and learning process of human brain. The BP neural network model is widely used in the fields of facial recognition and signal processing. The BP neural network consists of three sensing layers: the input layer, the middle layer, and the output layer [22].

The middle layer of BP neural network does not touch the outside of neural network, also known as hidden layer. The hidden layer can be not only a single layer, but also multiple layers, but the number of neurons in each layer is independent of each other. Figure 1 is a schematic diagram of a BP neural network model.

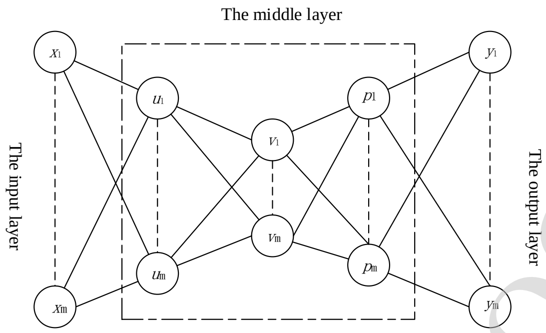

Figure 1 The Topology graph of BP Neural Network

BP neural network is a supervised learning algorithm. The main idea is that learning samples need to pass through the input layer to enter the neural network system. The back propagation algorithm is used to train the sample data repeatedly. The output vector and the expectation vector are respectively mapped to the neurons of each hidden layer according to the weight information of the input data, and the sample data is trained until the prediction error range of the maximum number of iterations is required or allowed [23]. Figure 2 is the learning flow chart of BP neural network.

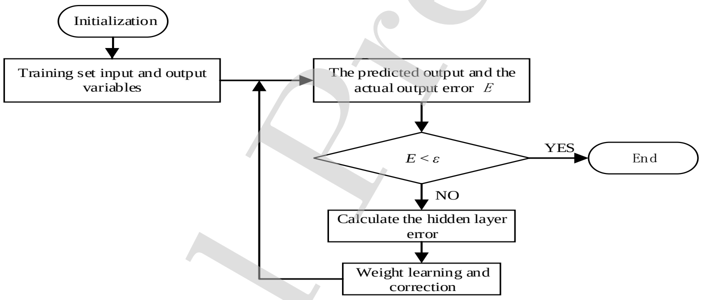

Figure 2 The flow chart of the BP neural network learning

3.2 Support vector regression model

3.2.1 Principle of support vector regression machine model

Support Vector Machine (SVM) is an important index for studying the learning performance of functions defined by statistical theory in the study of uniform convergence and generalization [24]. The Support Vector Regression (SVR) is a kind of generalized linear classifier that conducts binary classification of data by supervised learning, and its decision boundary is the maximum margin hyperplane for solving learning samples. In the process of solving practical problems, SVR is first applied in the field of classification and pattern recognition. The support vector machine introduces an insensitive loss function in the classification process. This introduction method can make it achieve better results in the application of regression problems.

When SVM is applied to classification problems, it works by constructing an optimal plane to distinguish two categories that need to be separated. Therefore, when SVM is used to solve regression problems, firstly, a plane should be constructed to minimize the distance from all training dataset to the plane. Linear regression mapped to high-dimensional space is used to approximate the input and output [25, 26]. The basic idea of SVM is shown in Figure 3.

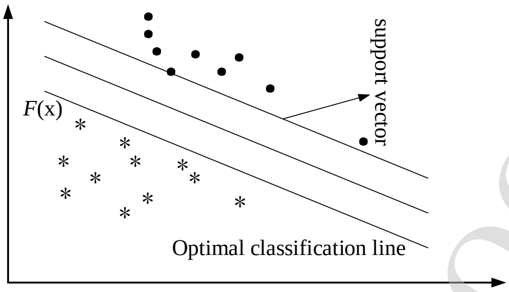

Figure 3 The basic idea of SVR

3.2.2 The principle of support vector regression machine model algorithm

The derivation process of the SVR algorithm is as follows.

The first step is to give a sample pair of training dataset: $T=\{(x_i,y_i),i=1,2,\ldots,l\}$, $x_i \in R^d$, and $y_i \in R$. Equation (1) is a linear regression function established in the high dimensional feature space.

$$f(x)=w\Phi(x)+b \tag{1}$$

Among them, w is the weight vector; $\Phi(x)$ is nonlinear mapping function; b representative bias.

The C-linear insensitive loss function is defined as follows.

$$L(f(x),y,\varepsilon)=\begin{cases}0, |y-f(x)|\le \varepsilon \\ \left||y-f(x)-\varepsilon\right|, |y-f(x)|>\varepsilon\end{cases} \tag{2}$$

Among them, $f(x)$ is the predictive value, y is the corresponding true value, ε represents the insensitive coefficient, which is used to control the accuracy of the fitting.

The second step is to introduce the slack variables, that is, ξ_i and ξ_j^*, and a minimized objective function is constructed according to the SRM principle.

$$\min \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{l}(\xi_i+\xi_i^*) \tag{3}$$

$$s.t.\begin{cases} y_i - w\Phi(x_i)-b\le \varepsilon+\xi_i \\ -y_i + w\Phi(x_i)+b\le \varepsilon+\xi_i^*, i=1,2,\ldots,l \\ \xi_i \ge 0, \xi_i \ge 0 \end{cases} \tag{4}$$

Among them, $C>0$ is penalty factor, that is, the degree of sample punishment when the training error is greater than ε.

In the third step, the Lagrange formula is introduced to obtain the dual optimization problem as in equation (5):

$$\min\left[\frac{1}{2}\sum_{i,j=1}^{l}\left(a_i-a_i^*\right)K\left(a_j-a_j^*\right)-\sum_{i=1}^{l}\left(a_i+a_i^*\right)\varepsilon+\sum_{i=1}^{l}\left(a_i-a_i^*\right)y_i\right] \tag{5}$$

$$s.t.\begin{cases}\sum_{i=1}^{l}\left(a_i-a_i^*\right)=0\\ a_i,a_i\in\left[0,C\right]\end{cases} \tag{6}$$

Among them, $K\left(x_i,x_j\right)=\Phi\left(x_i\right)\Phi\left(x_j\right)$ is the introduced kernel function.

In the fourth step, a_i and a_i^* can be solved from equation (6), and then w^* and b^* can be obtained. Then the regression function is as follows.

$$\begin{aligned}f(x)&=\sum_{i=1}^{l}\left(a_i-a_i^*\right)\Phi\left(x_i\right)\Phi\left(x_j\right)+b^*\\ &=\sum_{i=1}^{l}\left(a_i-a_i^*\right)K\left(x_i,x_j\right)+b^*\end{aligned} \tag{7}$$

The schematic diagram of the support vector regression machine is shown in Figure 4.

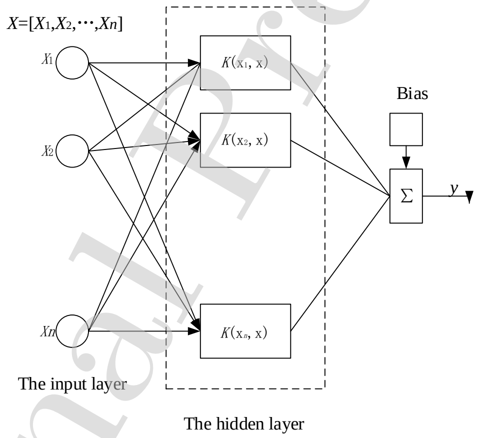

Figure 4 Schematic graph of SVR

3.3 Particle swarm optimization of least squares support vector machine model

3.3.1 Principle of least squares support vector machine model

Least squares support vector machine (LS-SVM) refers to the given sample set $S=\left\{\left(x_i,y_i\right),x_i\in R^n,y_i\in R\right\}_{i=1}^{l}$, if its corresponding linear regression equation is $f\left(x\right)=w^T\Phi(x)+b$, among them, x_i is the ith input, y_i is the ith output, and *l* is the sample size, $w_i\in R^n,b\in R$, $\Phi(\bullet)$ is kernel function that solves nonlinear problems, then in LS-SVM, the above problem can be transformed into the form of equations (8) and (9):

$$\min Q(w,b,e)=\frac{1}{2}\|w\|^2+\frac{\gamma}{2}\sum_{i=1}^{l}e_t^2 \tag{8}$$

$$w^T\Phi(x_i)+b+e_i \tag{9}$$

Among them, e_i is the error value between the ith predicted value and the estimated value, and γ is regularization parameter.

According to the comparison between equations (8) and (9), the squared error term is introduced into LS-SVM for optimizing. At this point, the constraint conditions of LS-SVM become equivalent conditions. The LS-SVM function expressed by Lagrange function is as follows.

$$L(w,b,e,\alpha)=Q(w,b,e)-\sum_{i=1}^{l}a_i\left[w^T\Phi(x_i+b+e_i-y_i\right] \tag{10}$$

The partial derivatives of w,b,e,α are as follows.

$$\begin{cases}\frac{\partial t}{\partial w}=0\Rightarrow w=\sum_{i=1}^{l}\alpha_i\Phi(x_i)\\ \frac{\partial t}{\partial b}=0\Rightarrow\sum_{i=1}^{l}\alpha_i=0\\ \frac{\partial t}{\partial e_i}=0\Rightarrow Ce_i-\alpha_i=0\\ \frac{\partial t}{\partial w}=0\Rightarrow w^T\Phi(x_i)+b+e_i-y_i=0\end{cases} \tag{11}$$

The above optimization conditions can be transformed into the following form.

$$\begin{bmatrix}0 & 1^T\\ 1 & \Omega+\gamma^{-!}1\end{bmatrix}\begin{bmatrix}b\\ a\end{bmatrix}=\begin{bmatrix}0\\ Y\end{bmatrix} \tag{12}$$

Among them, $\vec{1}=[1;\ldots,1];\Omega_{kj}=K(x_k,x_j),k,j=1,\ldots l$ is kernel function matrix, γ is regularization parameter, $\alpha=[\alpha_1;\ldots\alpha_1];Y=[y_i;\ldots y_1]$. If $A=\Omega+\gamma^{-1}I$, then:

$$\begin{cases}b=\frac{1^{-T}A^{-1}y}{1^{-1}A^{-1}\vec{1}}\\ \alpha=A^{-1}(Y-b\vec{I})\end{cases} \tag{13}$$

Finally, the prediction function of LSSVM can be obtained:

$$f(x)=\sum_{i=1}^{l}\alpha_iK(x,x_i)+b \tag{14}$$

In the formula, the nuclear function $K(x,x_i)$ can be replaced according to the MERCER theorem.

3.3.2 Principle of particle swarm optimization

The mathematical method of particle swarm optimization is that n potential solutions (particles) form a population $S=\{x_1,x_2,\ldots,x_n\}$ in an m-dimensional space. Among them, the ith particle represents a D-dimensional vector, $X_i=\left(x_{i1},x_{i2},\ldots,x_{iD}\right)$, $i=1,2,\ldots,n$, that is, the position of the i-th particle in the m-dimensional space.

According to the objective function, the fitness (such as distance) of each particle is calculated. P_i^t represents the optimal solution experienced by particle i iterating to t times, $rand\left(2\right)$ represents the optimal solution experienced by the entire population iterating to t times, $V_i=\left(v_{i1},v_{i2},\ldots,v_{iD}\right)$ indicates the velocity of particle i. The iterative approach of the basic particle swarm algorithm is as follows.

$$V_i^{i+1}=wV_i^t+c_1rand\left(1\right)\left(P_i^t-X_i^t\right)+c_2rand\left(2\right)\left(P_g^t-X_i^t\right) \tag{15}$$

$$X_i^{t+1}=X_i^t+V_i^{t+1} \tag{16}$$

Among them, $i=1,2,\ldots,n$ is particle label, t is number of iterations, w is inertia weight (The algorithm didn't have this parameter when it was first proposed, and the defa ult value is 1), c_1 and c_2 are learning factors (take the positive number), $rand\left(1\right)$ and $rand\left(2\right)$ are the random number between [0, 1].

For the moving speed v_{id}^k of particle i in the D dimension at time K, it should coincide with equation (17):

$$\begin{cases} v_{id}^k=v_{d,\max}, & \text{if } v_{id}^k=v_{d,\max} \\ v_{id}^k=-v_{d,\max}, & \text{if } v_{id}^k<v_{d,\max} \\ i=1,\ldots,m \\ d=1,\ldots n \end{cases} \tag{17}$$

$v_{\max}=\left(v_{1,\max},\ldots,v_{d,\max},v_{n,\max}\right)$ is the given maximum speed limit, ω is the inertia weight factor, the ω value of the standard PSO algorithm is determined by equation (5):

$$\omega^k=(\omega_{\max}-\omega_{\min})\frac{k}{k_w},k=1,\ldots,k_{\max} \tag{18}$$

$\omega_{\max}$ indicates the maximum value of ω, usually $\omega_{\max}=0.9$. $\omega_{\min}$ indicates the minimum value of ω, usually $\omega_{\min}=0.4$. $k_{\max}$ is the specified maximum number of iterations, and k is the current number of iterations.

The flow of the least square's method based on particle swarm optimization (PSO-LS-SVR) is shown in figure 5.

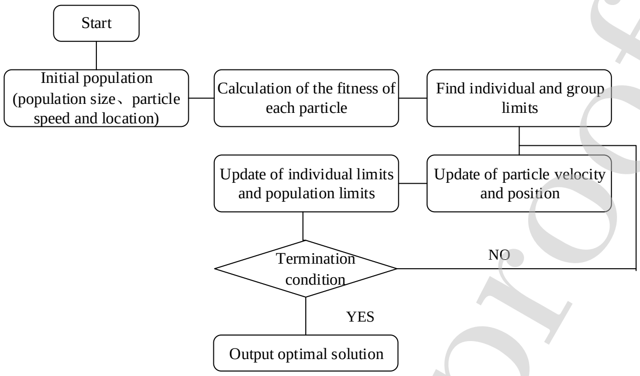

Figure 5 The flow graph of PSO

4.Experiments

4.1 Empirical analysis

4.1.1 Selection of target data

The research objects of this experiment adopt the rate of return of bitcoin as the target, and the data collection phase includes 943 days of data from June 2, 2016 to December 30, 2018. The rate of return of bitcoin is verified by model verification and experimental simulation. The target data is divided into training dataset and testing dataset according to the ratio of nearly 9: 1. The data of training dataset refers to the bitcoin yield data from June 2, 2016 to September 6, 2017, with a total of 830 pieces. The testing dataset refers to data from September 7, 2018 to December 30, 2018, with a total of 113 pieces. The goal in the data processing stage is to obtain the best learning machine model by training a large number of training dataset. After training, testing dataset is used to predict on the learning machine model for model verification and modification. This process is repeated to achieve the learning goal.

Since the return rate of financial products is expressed in the form of percentage, the return rate of bitcoin is no exception. In this study, for the convenience of calculation, the value before the percent sign is directly used for analysis. Figure 6 shows some data of yield rate of bitcoin. Due to the large amount of research sample data, only the data of the previous month and the last month in the collected data samples are shown.

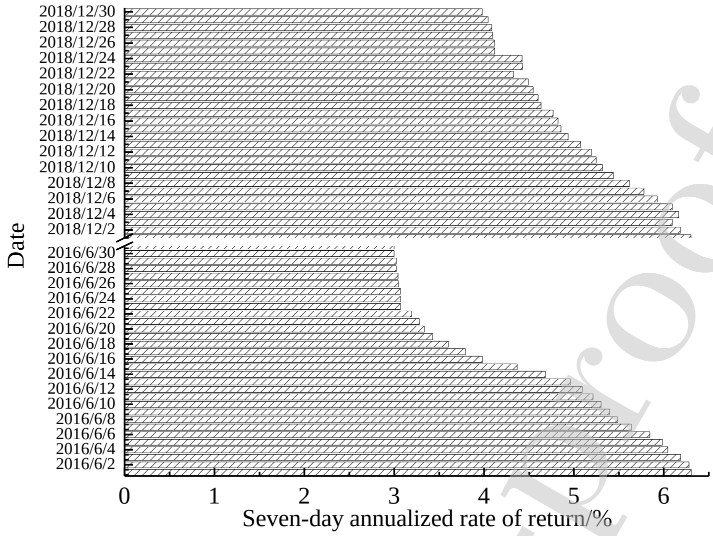

Figure 6 The Data selection of the Seven-day annualized rate of return of Bitcoin

Statistical analysis was performed on 943 pieces of yield rate of bitcoin, and the results shown in Table 1 were obtained by SPSS.

Table 1 Statistics of the annualized rate of return of Bitcoin

N	M	SD of Mean	SD	variance	β	β_k
943	4.29	0.2991	0.96002	0.912	0.238	0.141

It can be concluded from table 1 that the mean value of yield rate of bitcoin is 4.29 and the variance is 0.912, and the yield rate is non-normally distributed, while the skewness value of the distribution is 0.238 and the kurtosis value is -0.141. From the perspective of financial theory, Bitcoin's rate of return has an extreme value, which means that there is a risk in the investment of Bitcoin. Through the analysis of statistical data, the traditional time series analysis is not in line with the actual situation, because the distribution of bitcoin yield does not conform to the general rules. Therefore, an algorithm is needed to simulate and predict the yield data of bitcoin.

4.1.2 Empirical analysis of BP neural network

In this research, Matlab 2016 software is used to conduct BP neural network analysis on partial data of bitcoin yield rate. BP neural network has the function of feedback information in the hidden layer. Choosing the appropriate number of hidden layers can not only effectively adjust the transmission and feedback of prediction information, but also ensure the training speed of the model. Therefore, it is necessary to select the suitable number of nodes in the hidden layer, referring to equation (19):

$$l < \sqrt{m+n} + a \tag{19}$$

In the formula, the number of nodes in the input layer is n, the number of nodes in the hidden layer is l, the number of nodes in the output layer is m, and a is a constant of 1-10.

In order to ensure the generality of the BP neural network, it is necessary to consider selecting a double hidden layer neural network to analyze the data. Meanwhile, the number of nodes in the input layer should be set to 5, the number of nodes in the output layer should be set to

1, the number of nodes in the first hidden layer should be set to 6, the number of nodes in the second hidden layer should be set to 4, and the number of iterations should be set to 100. Firstly, 830 pieces of data are trained, and then 113 steps of single-step prediction are carried out. The output results are shown in figure 7.

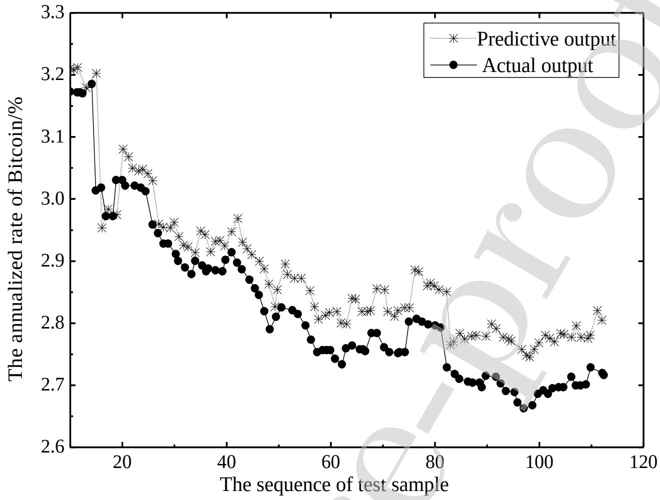

Figure 7 The prediction line of BT neural network of Bitcoin

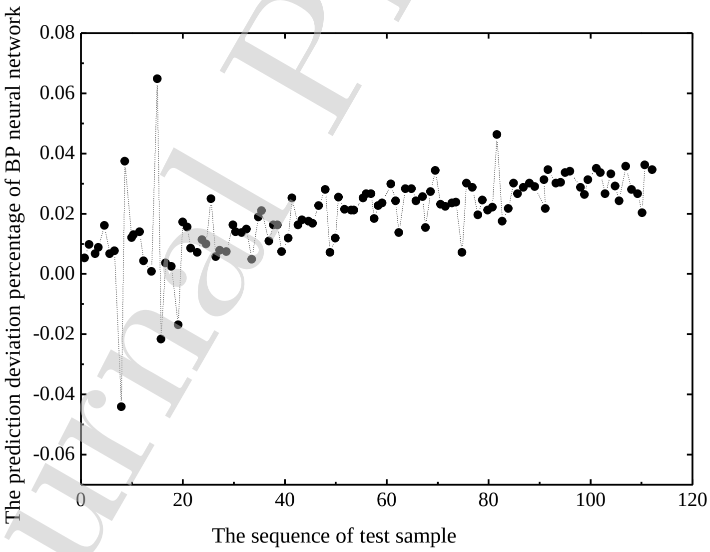

Figure 8 The prediction error percentage line of BT neural network of Bitcoin

As can be concluded from figure 7, although BP neural network can reflect the overall volatility of data in a timely manner on the whole, the overall fitting effect is not so ideal. At the beginning stage, the fitting effect between the output value and the predicted value is satisfactory. As time moves forward, the longer the time is delayed, the more the output value deviates from the predicted value. The magnitude of the data itself also has an impact on the magnitude of the prediction error, so it uses the value between the predicted value and the actual output value to measure the error in this research. As can be concluded from figure 8, as time goes by, the percentage error of BP neural network prediction keeps increasing, and the final percentage error remains around 0.03.

Furthermore, in the running of BP neural network, the maximum number of iterations set in this article is 100. However, the stability is reached by only 32 iterations of the model training and it indicates that the learning effect of the model is good. It can be concluded from the calculation results of MATLAB that the MAPE and MSE between the predicted output and the actual output are 3.2529 and 0.0098 respectively. Finally, it shows that the prediction effect of BP neural network on Bitcoin rate of return is very good, which can reflect the tendency and volatility of the rate of return data.

4.1.3 Empirical analysis of support vector regression machine

Similarly, the Matlab 2016 is used to analyze the bitcoin yield data according to the SVR. The predicted results are shown in figure 9.

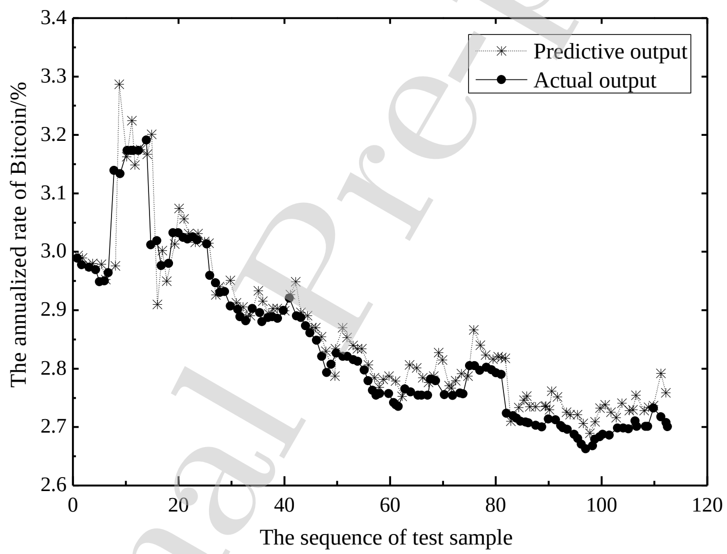

Figure 9 The prediction line of SVR of Bitcoin

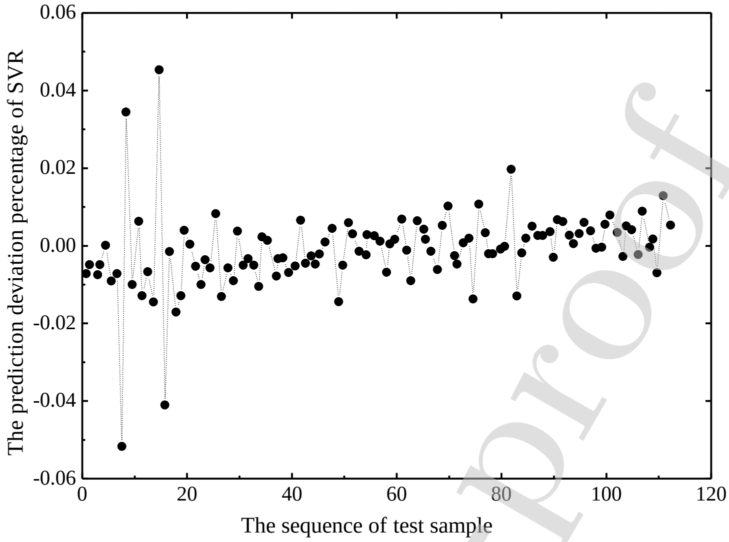

Figure 10 The prediction error percentage line of SVR of Bitcoin

By comparing the fitting effect of images in figure 7 and figure 9, it can be concluded that the fitting effect of SVR on sample data is better than that of BP neural network. As can be concluded from figure 9, from September 27, 2018, the output of bitcoin yield rate first shows an upward trend, and keeps rising until the peak of the rising stage around October 5. After the peak point, the trend turns into a decline of volatility, and the predicted trend is almost consistent with the actual trend of bitcoin yield rate. It can also be concluded from figure 10 that the error percentage of data on September 20, 25, November 17 and 27, 2018 is relatively large, and the remaining error percentage ranges from 0 to 0.02.

According to the results of SVR algorithm's running, the MSE and MAPE of the bitcoin yield data's training dataset part are 0.0006 and 0.3513, respectively, while the MSE and MAPE between the predicted output and the actual output of the testing dataset part are 0.0022 and 1.2239. It shows that the learning ability and the corresponding generalization ability of SVR model is better than that of BP neural network.

4.1.4 Empirical analysis of optimized least squares support vector machine model based on particle swarm optimization

Similarly, Matlab 2016 is used for PSO-LS-SVR analysis of yield rate data of bitcoin. The predicted results are shown in figure 11.

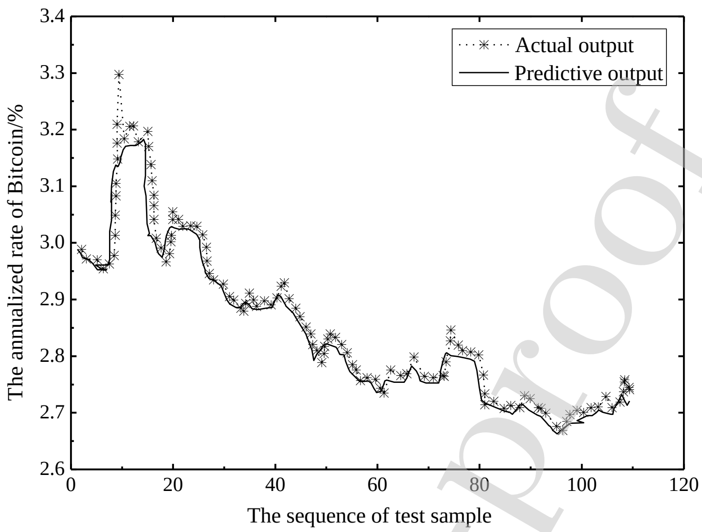

Figure 11 The prediction line of POS-LSSVR of Bitcoin

As can be concluded from figure 11, starting from September 27, 2018, the yield rate of bitcoin first shows an upward trend and reached the peak of the upward period around October 5, and then shows a downward trend of volatility. In the process of decline, there is a small fluctuation, and the predicted trend is completely consistent with the actual trend of the bitcoin yield rate.

It can be concluded from figure 7 to figure 11 that before September 27, the prediction fitting effects of the three algorithms (namely BP neural network, support vector regression machine and particle swarm optimization least-squares support vector machine model) are significantly different, with relatively large deviations, and all of them amplify the volatility of time series data. After October 5, the trend and volatility of the original data are well fitted between the three algorithms. In general, PSO-LS-SVR has the best fitting effect.

4.2 Model verification

4.2.1 Selection of target data during model verification

Bitcoin is a blockchain financial product. In order to verify the superiority of the PSO-LS-SVR model, it is necessary to select the yield rate data of representative blockchain financial products to verify it. Therefore, in this research, the ETH yield data is selected as the research object to ensure that the empirical analysis and algorithm are universal for the research of blockchain financial product data. The data selection time is from June 3, 2016 to February 1, 2018, a total of 617 data.

In order to avoid the influence of the data ratio adopted on the verification results, the data ratio of training dataset and testing dataset is adjusted to 8:2, resulting in 500 training dataset and 117 testing datasets.

In this research, Mean Squared Error (MSE) and Mean Absolute Percentage Error (MAPE) are used to compare the accuracy. The calculation equations of MSE and MAPE are respectively as follows:

$$MSE(y,\widehat{y}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \widehat{y})^2 \tag{20}$$

$$MAPE(y,\widehat{y}) = \frac{\sum_{i=1}^{n}\left|\frac{y_i - \widehat{y}_i}{y_i}\right|}{n} \times 100 \tag{21}$$

Among them, y_i represents the true value; $\widehat{y}_i$ represents the predicted value; N is the number.

4.2.2 Empirical analysis of model validation

BP neural network, SVR and PSO-LS-SVR are used to train and test the yield of ETH. The predictions of the yields by the BP neural network, SVR, and PSO-LS-SVR are shown in figure 12, figure 13, and figure 14, respectively.

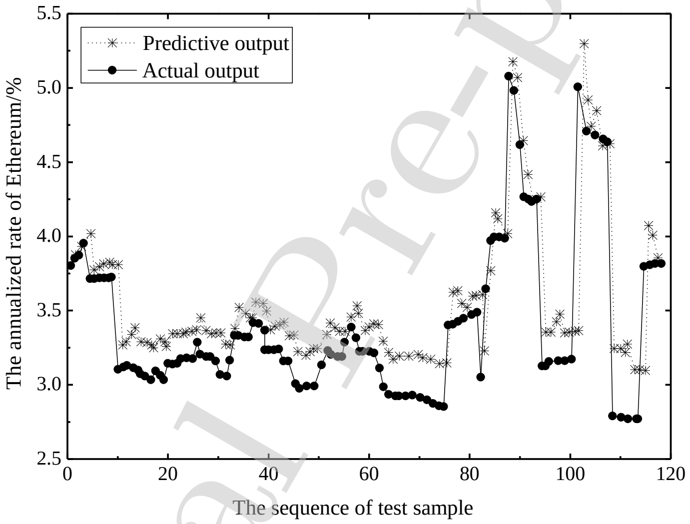

Figure 12 The prediction line of BT neural network of Ethereum

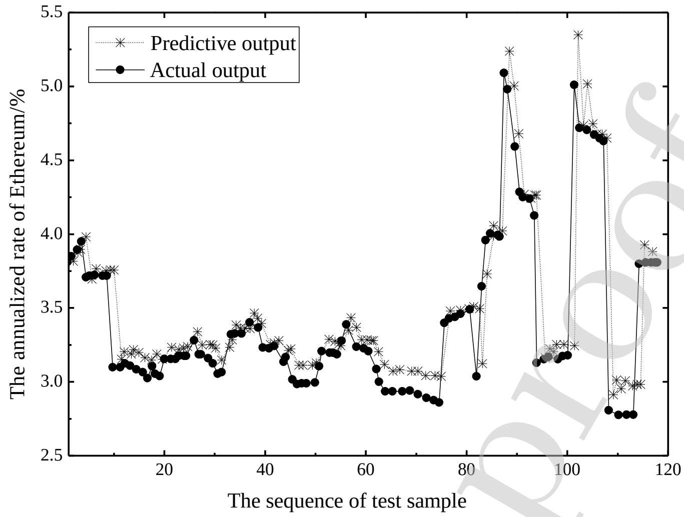

Figure 13 The prediction line of SVR of Ethereum

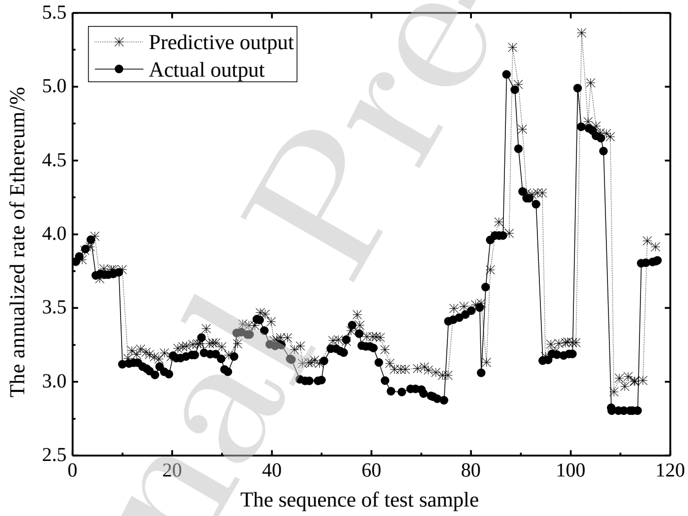

Figure 14 The prediction line of POS-LS-SVR of Ethereum

By comparing figure 12 to figure 14, it can be concluded that the best predictive and fitting effect on the ETH yield rate data is the PSO-LS-SVR algorithm, followed by the SVR and the BP neural network. The results of model verification are consistent with the above experimental results, and it is concluded that PSO-LS-SVR has the best effect in prediction of yield rate.

Table 2 is a comparison table of the accuracy of the three algorithms in predicting Ethereum. It can be concluded from the table that MSE value of PSO-LSSVR is the smallest in the training dataset, then BP neural network, and finally SVR. However, the value of POS-LESSVR algorithm is the minimum for the MAPE evaluation criteria. Although the learning ability of the three algorithm models for Ethereum training samples is not ideal compared with that of the bitcoin training samples, but the overall learning ability of PSO-LSSVR is good. As for partial data of the testing dataset, PSO-LSSVE has the best effect from the perspective of MSE evaluation criteria, followed by SVR, and finally BP neural network. From the final evaluation results of the training dataset and the testing dataset, the model fitting effect and prediction effect of SVR are not always better than that of BP neural network. However, the PSO-LSSVR algorithm can predict the financial time series well.

Table 2 Comparison table of the accuracy of the three algorithms in predicting Ethereum

Algorithm model	Training dataset		Testing dataset	
	MSE	MAPE	MSE	MAPE
BP neural network	0.0712	3.2356	0.1021	4.7372
SVM	0.1091	4.7237	0.1132	4.5721
PSO-LSSVR	0.0701	3.2126	0.0973	4.3531

5.Discussion

5.1 Result analysis

In this research, the BP neural network, support vector regression machine and particle swarm optimization algorithm were used to optimize the least squares support vector model for empirical analysis and model verification. The verification results show that the particle swarm optimization algorithm least squares support vector regression machine has the best fitting effect on financial products in the three algorithms. According to the combination of figure 7 and figure 8, the prediction of BP neural network is well fitted in the early stage, while the predicted value in the later stage is higher than the actual value, so it can be concluded that BP neural network can reflect the volatility and trend of data, and the prediction effect on yield rate of bitcoin is relatively good. Figure 9 and figure 10 show that SVR algorithm not only has better learning ability than BP neural network, but also has better model fitting effect and prediction effect than BP neural network. According to the comparison in figure 7, figure 9 and figure 11, the PSO-LS-SVR algorithm has the best performance and the best fitting degree, followed by the SVR algorithm and then the BP neural network. Subsequently, model simulation experiments were carried out to verify the results, and the comparison and analysis of the verification results showed that, among the three algorithms, PSO-LS-SV had the best fitting of prediction results of ETH.

5.2 Discussion of results

In this study, the rate of return of bitcoin was taken as the research object. Through the prediction analysis and error analysis of BP neural network, support vector regression machine and PSO-LS-SVR model, it was concluded that PSO-LS-SVR model has the best fitting prediction effect on the time series data of rate of return of bitcoin, followed by SVR algorithm and BP neural network. And PSO-LS-SVR model played a good guiding role in the prediction of yield rate of blockchain financial products. As the first successful financial product in the application of blockchain technology, the study on the income of bitcoin is of great guiding significance to the development of blockchain financial products. The results of this research show that the improved particle swarm optimization algorithm has achieved a better structure in predicting the return rate of bitcoin. The reason is that the RBF core is used in the prediction process. On the one hand, because RBF kernel can map samples from low dimension to high latitude, it is easy to solve the nonlinear situation between class labels and features. On the other hand, the proportion of RBF kernel as polynomial kernel has fewer parameters, which can make the model simple and feasible.

6.Conclusions

In this research, the rate of return of bitcoin was taken as the research object, and three algorithms, namely BP neural network, SVR and PSO-LS-SVR, were adopted to carry out experimental simulation and model verification on the research object. The verification results showed that the experimental effect of SVM algorithm was better than that of BP neural network. The PSO-LS-SVR algorithm adopted in this research was an optimization of the algorithm of support vector regression machine. The above experimental results were verified with the rate of return data of ETH. And the verification results showed that among the three algorithms, the PSO-LS-SVR algorithm adopted in this research had the best fitting effect on the yield rate of blockchain finance.

Just like traditional finance, although blockchain financial products develop rapidly by virtue of technological advantages, they are also faced with risks. The yield of any wealth management product will change with the change of economic market. Blockchain financial products are born out of Internet companies, and any Internet finance company has the potential to depreciate assets. If the company's assets shrink, its corresponding blockchain financial products will also encounter risks. Therefore, in this research, the rate of return of bitcoin was taken as the research object. Firstly, three algorithm models were used to predict the rate of return of bitcoin, and the optimal prediction algorithm model was obtained. It was then verified with the forecast of the rate of return of ETH. The results showed that PSO-LS-SVR had a good predictive effect on blockchain financial products. It is hoped that the content of this research can provide correct guidance for the majority of investors in the investment of block chain financial products.

References

[1] Xiong F, Gong P, Jin P, Fan JF (2019) Supply chain scheduling optimization based on genetic particle swarm optimization algorithm. Cluster Comput 22(S6):14767-14775. https://doi.org/10.1007/s10586-018-2400-z.

[2] Mallick S, Kar R, Mandal D, Ghoshal PS (2017) Optimal sizing of CMOS analog circuits using gravitational search algorithm with particle swarm optimization. Int J Mach Learn Cybern 8(1):309-331. https://doi.org/10.1007/s13042-014-0324-3.

[3] Fielding B, Zhang L (2017) Evolving Image Classification Architectures with Enhanced Particle Swarm Optimisation. IEEE Access 6:68560-68575. https://doi.org/10.1109/access.2018.2880416.

[4] Hyvärinen H, Risius M, Friis G (2017) A Blockchain-Based Approach Towards Overcoming Financial Fraud in Public Sector Services. Bus Inform Syst Eng 59(6):441-456. https://doi.org/10.1007/s12599-017-0502-4.

[5] Islam N, Faheem Y, Din IU, Talha M, Guizani M, Khalil M (2019) A blockchain-based fog computing framework for activity recognition as an application to e-Healthcare services. Fut Gener Comput Syst 100: 569-578. https://doi.org/10.1016/j.future.2019.05.059.

[6] Manshaei MH, Jadliwala M, Maiti A, Fooladgar M (2018) A Game-Theoretic Analysis of Shard-Based Permissionless Blockchains. IEEE Access 6:78100-78112. https://doi.org/10.1109/access.2018.2884764.

[7] Hussein AF, Arunkumar N, Ramirez-Gonzalez G, Abdulhay E, Tavares MJRS, Albuquerque de VHC (2018) A medical records managing and securing blockchain based system supported by a Genetic Algorithm and Discrete Wavelet Transform. Cogn Syst Res 52:1–11.

https://doi.org/10.1016/j.cogsys.2018.05.004.

[8] Kiktenko EO, Pozhar NO, Anufriev MN, Trushechkin AS, Yunusov RR, Kurochkin YV, Lvovsky AI, Fedorov AK (2018) Quantum-secured blockchain. Quantum Sci Technol 3(3): 035004. https://doi.org/10.1088/2058-9565/aabc6b.

[9] Hong CH, Varghese B (2019) Resource Management in Fog/Edge Computing. ACM Comput Surv 52(5): 1-37. https://doi.org/10.1145/3326066.

[10] Mousavi SM, Bahreininejad A, Musa SN, Yusof F (2017) A modified particle swarm optimization for solving the integrated location and inventory control problems in a two-echelon supply chain network. J Intell Manuf 28(1):191-206. https://doi.org/10.1007/s10845-014-0970-z.

[11] Gharavian D, Bejani M, Sheikhan M (2017) Audio-visual emotion recognition using FCBF feature selection method and particle swarm optimization for fuzzy ARTMAP neural networks. Multimed Tools Appl 76(2):2331-2352. https://doi.org/10.1007/s11042-015-3180-6.

[12] Boubaker S. (2017) Identification of nonlinear Hammerstein system using mixed integer-real coded particle swarm optimization: application to the electric daily peak-load forecasting. Nonlinear Dynamics 90(2):797-814. https://doi.org/10.1007/s11071-017-3693-9.

[13] Zhu ZC, Dhokia V, Newman ST (2017) A novel decision-making logic for hybrid manufacture of prismatic components based on existing parts. J Intell Manuf 28(1):131-148. https://doi.org/10.1007/s10845-014-0966-8.

[14] Jin X, Zhou DM, Yao SW, Nie RC, Jiang Q, He KJ, Wang Q (2017) Multi-focus image fusion method using S-PCNN optimized by particle swarm optimization. Soft Comput 22(19):6395-6407. https://doi.org/10.1007/s00500-017-2694-4.

[15] San PP, Ling SH, Nuryani, Nguyen H (2014) Evolvable rough-block-based neural network and its biomedical application to hypoglycemia detection system. IEEE Trans Cybern 44(8):1338-1349. https://doi.org/10.1109/tcyb.2013.2283296.

[16] Yan HY, Zhu YG (2017) Bang–bang control model with optimistic value criterion for uncertain switched systems. J Intell Manuf 28(3):527-534. https://doi.org/10.1007/s10845-014-0996-2.

[17] Yin YJ, Xu D, Wang XG, Bai MR (2015) Online State-Based Structured SVM Combined With Incremental PCA for Robust Visual Tracking. IEEE Trans Cybern 45(9):1988-2000. https://doi.org/10.1109/tcyb.2014.2363078.

[18] Dao TK, Pan TS, Nguyen TT, Pan JS (2018) Parallel bat algorithm for optimizing makespan in job shop scheduling problems. J Intell Manuf 29(2):451-462. https://doi.org/10.1007/s10845-015-1121-x.

[19] Lin YH, Hu YC (2018) Residential consumer-centric demand-side management based on energy disaggregation-piloting constrained swarm intelligence: towards edge computing. Sen18(5):1365. https://doi.org/10.3390/s18051365.

[20] Havangi R (2018) Target Tracking based on Improved Unscented Particle Filter with Markov Chain Monte Carlo. IETE J Res 64(6):873-885. https://doi.org/10.1080/03772063.2017.1369909.

[21] Martínez CA, Barrón FR, Molina LH, Ramírez SMA, Villa VLA, Cortés AP, Cheng KT (2017) An Automatic Functional Coverage for Digital Systems Through a Binary Particle Swarm Optimization Algorithm with a Reinitialization Mechanism. J Anal Test

33(4):431-447. https://doi.org/10.1007/s10836-017-5665-x.
[22] Wang FY, Wang X, Li JJ, Ye PJ, Li Q (2019) Social Computing: From Crowdsourcing to Crowd Intelligence by Cyber Movement Organizations. IEEE Trans Comput Soc Syst, 2019, 6(4): 619-626. https://doi.org/10.1109/tcss.2019.2930420.
[23] Larsson E, Shcherbakov V, Heryudono A (2017) A Least Squares Radial Basis Function Partition of Unity Method for Solving PDEs. SIAM J Sci Comput 39(6):A2538-A2563. https://doi.org/10.1137/17m1118087.
[24] Chatterjee S, Sarkar S, Hore S, Dey N, Ashour AS, Balas VE (2017) Particle swarm optimization trained neural network for structural failure prediction of multistoried RC buildings. Neural Comput Appl 28(8):2005-2016. https://doi.org/10.1007/s00521-016-2190-2.
[25] Delice Y, Aydoğan EK, Özcan U, İlkay MS (2017) A modified particle swarm optimization algorithm to mixed-model two-sided assembly line balancing. J Intell Manuf 28(1):23-36. https://doi.org/10.1007/s10845-014-0959-7.
[26] Havangi R (2018) Target Tracking based on Improved Unscented Particle Filter with Markov Chain Monte Carlo. IETE J Res 64(6):873-885. https://doi.org/10.1080/03772063.2017.1369909.