International Journal of Grid and Distributed Computing
Vol. 9, No. 9 (2016), pp.307-316
http://dx.doi.org/10.14257/ijgdc.2016.9.9.26

Inventory Prediction Research Based on the Improved BP Neural Network Algorithm

Fu-bin Pan*

School of Economics and Management, Xiamen University of Technology, Xiamen, 361024, China
**Corresponding Author E-mail: 475591423@qq.com*

Abstract

Since the idea of the supply chain management is proposed, many enterprises have attached great importance to the supply chain management and pay a lot of manpower and resources to study. It is also the focus to study the inventory in the field of the supply chain. Quantity of the inventory is not only related to the profit of the enterprises, but also related to the survival of the entire supply chain. Predicting the inventory can improve the ability of enterprises to prevent risk, increase the profits and reduce the losses. In order to predict better on inventory, we propose an improved BP neural network algorithm. In the algorithm, we use the improved GSA algorithm to optimize the parameters of BP neural network algorithm and improve the BP neural network algorithm aiming at its deficiency. The experimental results show that this method has good prediction effect.

***Keywords**: BP neural network algorithm; inventory; prediction*

1. Introduction

In the normal operation of the enterprises, the inventory cost occupies a great proportion of the total cost of enterprise [1].If the inventory is too high, the enterprise will take a lot of inventory cost. If the inventory is too low, the enterprise will lead to the great loss because of shortages and will affect the consumer's information and brand image [2-3].Predicting the inventory on the supply chain can better control the inventory cost and increase the amount of money in circulation [4].

There were many scholars studying the inventory prediction. Giacomo Sbrana and Andrea Silvestrini provided the analytical prediction properties of the random coefficient exponential smoothing model in the "multiple source of error" framework. This model presented an empirical application comparing the prediction accuracy of the suggested model vis-à-vis other benchmark models [5]. Brent D. Williams and other people thought that echelon inventory processes translated into a long-run balance or equilibrium between orders and POS, which we referred to as the inventory balance effect, allowing for more accurate order prediction [6]. Aris A. Syntetos and other people studied the prediction of the intermittent inventory demands by evaluating the effects of prediction on stock control performance in more than 7,000 demand series [7].Enrique Martin Alcalde and other people presented a method to predict the yard inventory of container terminals over an extended period and address an integrated yard planning problem for determining the optimal storage space utilization [8]. JiSun Shin and other people considered a production adjustment method for an automobile parts production process by using a dynamic Bayesian network to predict the production inventory [9]. Liu Ting made the inventory management and the control state of the automotive bearings as the research object [10]. The author used the principle of the artificial neural network to predict safety inventory of the parts. Then, he applied the theory of the data envelopment analysis (DEA) in operations research to analyze the inventory control performance in different periods of one year. Finally, he formed an information management system to

ISSN: 2005-4262 IJGDC

predict the safety inventory and the performance evaluation. Li Kongming thought that the enterprise must pay attention to the inventory management in order to improve the economic efficiency and enhance the competitiveness [11]. In the third party logistics center inventory predict factors, the independent random uncertain factors were too much. Therefore, Zhao Jun and Wang Xiao used the data mining prediction to establish the flexible and accurate prediction database. Then, the author applied the inventory demand prediction model based on the exponential smoothing model to the logistics center inventory demand prediction [12]. Combined with the manufacturing enterprise inventory prediction status, Gao Haojie analyzed deeply the inventory demand characteristics of the valve manufacturing industry enterprises. Then, the author applied the BP neural network to the inventory demand prediction of one valve manufacturing enterprises. In addition, the author also deeply analyzed the prediction accuracy and sensitivity of the model [13]. In addition, there were some scholars conducting the research on the inventory prediction [14-16].

BP neural network was a method of the artificial intelligence. This method is one of the most simple abstract and simulation for the human brain system. BP neural network had been widely used in the prediction. Feng Yu and Xiaozhong Xu proposed an appropriate combinational approach which was based on improved BP neural network for short-term gas load forecasting, and the network was optimized by the real-coded genetic algorithm [17]. The integration model improved by modified additional momentum factor through analyses and comparisons of the above several different combinational algorithms. Jun Wang *et al.*, used the back propagation neural network improved by cuckoo search algorithm (hereafter CS-BP neural network) to forecast lightning occurrence from sounding-derived indices over Nanjing is presented [18]. Some statistical skill score parameters (namely POD, SAR, CSI, et.al.) indicated that the CS-BP model excelled in lightning forecasting and had a better performance compared with the traditional BP neural network and linear multiregression method. Aiming at the problem that cannot be ignored in the ERP supply chain security stocks, Wang Dongxu, Shen Yimin and Wang Zhiqiang established the BP neural network model, trained, studied and predicted the actual problems. Then, they achieved the better predicted results that were superior to the traditional method [19]. In order to improve the prediction accuracy of the BP neural network prediction model, Li Song, Liu Lijun, Zhai Man proposed the optimized BP neural network prediction method based on the Particle Swarm [20]. By using the improved particle swarm algorithm to optimize the weights and thresholds of the optimized BP neural network, the author trained the BP neural network and got the optimal solution. The author applied the method to the short-term traffic flow prediction and the achieved good results. In addition, there were many scholars studying the BP neural network prediction [21-24].

To predict the inventory not only can effectively control the cost, but also can enhance the confidence of consumers and enterprise reputation. In this paper, we propose an improved BP neural network algorithm and apply the algorithm to study the inventory prediction. The structure of this paper is as follows. The first part is introduction. In this part, we introduce the research status of the inventory prediction and the BP neural network. The second part is the BP neural network. The third part is the GSA algorithm. The fourth part is the improved BP neural network algorithm. In this part, we use the improved GSA algorithm to optimize the parameters of BP neural network algorithm and improve the shortages of the BP neural network algorithm. Then, we propose the improved BP neural network algorithm. The fifth part the experiment and the sixth part is the conclusion.

2. BP Neural Network Algorithm

As the most widely used algorithm in the artificial neural network, BP neural network has a complete theoretical system and learning mechanism. It imitates the reaction process of the human brain neurons to external excitation signal, establishes the multilayer perceptron model and uses the learning mechanism of the positive signal propagation and error reverse regulation. By iterative learning, it successfully builds an intelligent network model of nonlinear information processing. BP neural network model is as follows.

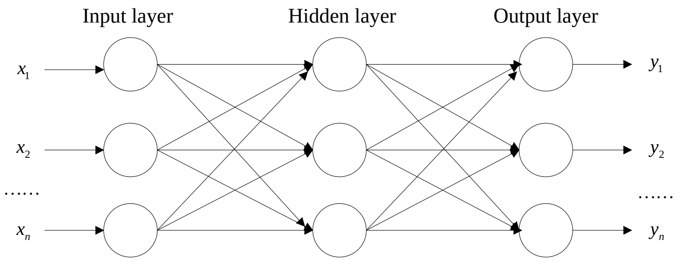

Figure 1. BP Neural Network

In BP neural network, there are n nodes and L layers. Each unit only receives the output information of the previous layer and it is as the input information of the next layer. We give N samples $(x_k, y_k)(k = 1, 2, \cdots, N)$. The output of ant node is O_i and the input is x_k. The output of the network is y_k. The output of the node i is O_{ik}. When we input the k sample, the input of the node j in k layer is,

$$net_{jk}^{l} = \sum w_{ij}^{l} O_{jk}^{l-1} \tag{1}$$

The output is,

$$O_{jk}^{l-1} = f(net_{jk}^{l}) \tag{2}$$

Where, O_{jk}^{l-1} is the $l-1$ layer.
The error of the output node is,

$$E_k = \frac{1}{2}\sum (y_{jk} - \bar{y}_{jk})^2 \tag{3}$$

The total errors are,

$$E = \frac{1}{2N}\sum_{k=1}^{N} E_k \tag{4}$$

The actual output of the unit j is $\bar{y}_{jk}$.

$$W_{jk} = \frac{\partial E_k}{\partial net_{jk}^{l}} \tag{5}$$

$$\frac{\partial E_k}{\partial w_{jk}^{l}} = \frac{\partial E_k}{\partial net_{jk}^{l}} O_{jk}^{l-1} = W_{jk}^{l} O_{jk}^{l-1} \tag{6}$$

If node j is the output unit,

$$W_{jk} = \frac{\partial E_k}{\partial net_{jk}^{l}} = \frac{\partial E_k}{\partial \bar{y}_{jk}} \frac{\partial \bar{y}_{jk}}{\partial net_{jk}^{l}} = -(y_k - \bar{y}_k) f'(net_{jk}^{l}) \quad , \quad O_{jk}^{l} = \bar{y}_{jk} \tag{7}$$

Else

$$W_{jk} = \frac{\partial E_k}{\partial net_{jk}^{l}} = \frac{\partial E_k}{\partial O_{jk}^{l}} \frac{\partial O_{jk}^{l}}{\partial net_{jk}^{l}} = \frac{\partial E_k}{\partial O_{jk}^{l}} f'(net_{jk}^{l}) \tag{8}$$

Due to

$$\sum \frac{\partial E_k}{\partial net_{jk}^{l+1}} w_{mj}^{l+1} = \sum W_{mk}^{l+1} w_{mj}^{l+1} \tag{9}$$

Then

$$W_{jk} = \sum W_{mk}^{l+1} w_{mj}^{l+1} f'(net_{jk}^{l}) \tag{10}$$

We can get,

$$W_{jk} = \sum W_{mk}^{l+1} w_{mj}^{l+1} f'(net_{jk}^{l}) \tag{11}$$

$$\frac{\partial E_k}{\partial w_{ij}} = W_{jk}^{l} O_{jk}^{l-1} \tag{12}$$

The steps of the BP neural network algorithm are as follows.

Firstly, it initializes the network and the training parameters. It gives the connection weights values and the neuron threshold values of the implicit layer and each node in output layer a random arbitrary decimal. In general, the range is [-1,1].

Secondly, it provides the training model. That is, it needs to provide the input vector X and the expected output T.

Thirdly, it repeats the following processes until convergence.

1. For k, from $k=1$ to $k=n$, we calculate O_{jk}^{l-1}, net_{jk}^{l} and net_{jk}^{l} for $l = L-1, \cdots L-2, \cdots, 2$. According to formula 11 and 12, we calculate O_{jk}.

2. Weight correction

$$\frac{\partial E_k}{\partial w_{ij}} = \sum_{k=1}^{N} \frac{\partial E}{\partial w_{ik}} \tag{13}$$

$$w_{ij} = w_{ij} - \_\frac{\partial E}{\partial w_{ij}}, \_ > 0 \tag{14}$$

Where, $-$ is the learning rate.

3. GSA

In the gravitational search algorithm, the individuals with different quality are attracted to each other. The individual performance is decided by its quality. The individuals are attracted by the gravity and they move to the individuals that the quality is heavy. The size of the gravitation is proportional to the quality of the two particles and is inversely proportional to the distance of the two particles.

$$F = G\frac{M_1 M_2}{R^2} \tag{15}$$

Where, F is the force between two particles. G is the gravitational constant. M_1 and M_2 are the qualities of the particle 1 and particle 2. R is the distance between the particle 1 and particle 2.

The relationship between the gravitational constant and the time is as follows.

$$G(t) = G(t_0) \cdot (\frac{t_0}{t})^{\beta}, \beta < 1 \tag{16}$$

$G(t)$ is the value for the gravitational constant at t time. $G(t_0)$ is the value for the gravitational constant at t_0 time.

We assume that there are N particles. We define that the location of the i particle is $X_i = (x_1, x_2, \cdots, x_i)$ and $i = 1, 2, \cdots, N$. N is the population. x_i is the location of the i particle. According to the formula (17) and the formula (18), we calculate the inertial quality of the particle. From the formula (19), we get the normalized quality.

$$M_{ai} = M_{pi} = M_{ii} = M_i, i = 1, 2, \cdots, N \tag{17}$$

$$m_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)} \tag{18}$$

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^{N} m_j(t)} \tag{19}$$

Where, $fit_t(t)$ is the fitness for the particle i at t time.

For the minimum problem, $best(t)$ and $worst(t)$ are defined as follows.

$$best(t) = \min_{j=\{1,\cdots N\}} fit_j(t) \tag{20}$$

$$worst(t) = \max_{j=\{1,\cdots N\}} fit_j(t) \tag{21}$$

For the maximum problem, $best(t)$ and $worst(t)$ are defined as follows.

$$best(t) = \max_{j=\{1,\cdots N\}} fit_j(t) \tag{22}$$

$$worst(t) = \min_{j=\{1,\cdots N\}} fit_j(t) \tag{23}$$

At t time, the gravitation between the particle i and particle j is as follows.

$$F_{ij}(t) = G(t)\frac{M_i(t) \times M_j(t)}{R_{ij}(t) + \varepsilon}(x_j(t) - x_i(t)) \tag{24}$$

ε is a very small constant in order to preventing that the denominator is zero.

$R_{ij}(t)$ is the Euclidean distance between the particle i and particle j.

$$R_{ij}(t) = \left\| x_i(t), x_j(t) \right\|_2 \tag{25}$$

$G(t)$ is the gravity coefficient and we initialize it as G_0.

$$G(t) = G_0 \times e^{-\alpha \frac{t}{T}} \tag{26}$$

G_0 and α are the constants. T is the maximum iterations.

In the gravitational search algorithm, in order to increase the randomization the algorithms, we think that the force on the particle i is equal to the sum that forced on other particles in d dimension. The force is defined as follows.

$$F_i(t) = \sum_{j=1, j\neq i}^{N} rand_j F_{ij}(t) \tag{27}$$

$rand_j$ is a random number belongs to $[0,1]$.

According to the second law of Newton, in d dimension, the acceleration of the particle i is defined as follow at t time.

$$a_i(t) = \frac{F_i(t)}{M_i(t)} \tag{28}$$

Where, $M_i(t)$ is the inertial mass of the particle i at t time.

According to the following formula, we update the speed and position of the particle.

$$v_i(t+1) = rand_i \times v_i(t) + a_i(t) \tag{29}$$

$$x_i(t+1) = x_i(t) + x_i(t+1) \tag{30}$$

According to the above description, the processes of the gravitational search algorithm are as follows.

1. Search the identification.

2. Initialize randomly the search group.
3. Calculate the fitness value of each particle.
4. Update the gravity coefficient $G(t), best(t), worst(t), M_i(t), i = 1, 2, \cdots, N$.
5. Calculate the force of the different directions and calculate the force.
6. Calculate the speed and the acceleration.
7. Update the position of the particle.
8. Circle the step 3 to the step 7 until the conditions meet.
9. End.

4. The Improved BP Neural Network Algorithm

In the BP neural network, the network initial weights and thresholds have a great influence on the learning effect. We use the improved GSA algorithm to optimize the initial weights and thresholds for the BP neural network. In addition, we improve the shortcomings of the BP network and put forward the improved BP neural network algorithm.

Firstly, we propose the improved GSA algorithm and optimize the initial weights and thresholds for the BP neural network.

According to the movement of the particles, the GSA algorithm completes the search of the space. In the search process, because the particles are not sharing information group, the development ability of particle is reduced. At the same time, because the inertia weight adopts the way by the random number, the global and local search capabilities of the particles have not been effectively balanced. Therefore, in order to improve the search speed and precision of GSA, we introduce the black hole to improve the particle development ability by improving the inertia weight and balancing the global search and local search. In the GSA that introducing the black hole, we take the particle with the optimal adaptation degree value as the black hole. The particle is forced by the black holes and other particles. Therefore, when the particle moves, it moves to the black holes. Then, the particle completes the search for the whole area. The radius of the black hole is defined as follows.

$$R = M_{bh}(t) \cdot \log(t) \tag{31}$$

where, $M_{bh}(t)$ is the quality of the black hole at t time.

The quality of the particle with the optimal particle is defined as follows.

$$M_{bh}(t) = \frac{\max\limits_{i \in \{1,2,\cdots,N\}} (fit_i(t) - worst(t))}{\sum\limits_{j=1}^{N} fit_i(t) - worst(t)} \tag{32}$$

The formula of the particles that moves to the black hole in the search area of the black hole is as follows.

$$x_i(t+1) = x_t(t) + rand \cdot (x_{bh} - x_i(t)) \tag{33}$$

$x_t(t)$ is the position of the particle i at t time. x_{bh} is the position of the black hole in the whole search space.

In the black hole search process, the particle that going to the black hole boundary conditions will be sucked in the black hole. When a particle is absorbed by the black hole, the space will randomly generate a new particle at the same time. The total number of the particles in the black hole space remains unchanged. If the fitness degree value of the particle is better than the fitness degree value of the black hole, it indicates that the position of the particle is better. Then, the particle and the black hole need to change their positions. We take the new generated black hole as the center by the algorithm. Repeating the process, other particles will continue to change the position, move to the black hole and be attracted by the black hole.

In order to maintain the balance of the global and local search ability, we add the inertia weight.

$$v_i(t+1) = w_i(t)v_i(t) + a_i(t) \tag{34}$$

Where, $w_i(t)$ is the inertia weight of the i particle at t time.

$$w_i(t) = \begin{cases} w_{\min} - \dfrac{(w_{\max} - w_{\min}) \cdot (fit_i(t) - worst(t))}{fit_{avg}(t) - worst(t)}, \ fit_i(t) \le fit_{\max}(t) \\ w_{\max}, \ fit_i(t) > fig_{\max} t(t) \end{cases} \tag{35}$$

$w_{\min}$ and $w_{\max}$ are the minimum and the maximum values of the inertia weight. $fit_{avg}(t)$ is the average fitness degree at the current time. In the formula, the inertia weight particle changes with the change of the objective value of the particle. Then, it achieves the global search and the local search.

In the topological structure of BP neural network, the input node and the output node are determined by the problem itself. The key is the number of the hidden layer and hidden layer nodes. In order to find a suitable number of the hidden nodes, the best way is in the network learning process. According to environmental requirements, it is self-organization learning, adjusts their structure and gets a proper size of the neural network model. During the process of the training, we can combine and delete reasonably the number of the hidden nodes.

We set O_{pi} is the output of the hidden node i when it learns the p sample. O_{pj} is the output of the hidden node j when it learns the p sample. N is the total number of learning samples.

$$\overline{O}_i = \frac{1}{N}\sum_{p=1}^{N} O_{pi} \tag{36}$$

$$\overline{O}_j = \frac{1}{N}\sum_{p=1}^{N} O_{pj} \tag{37}$$

Set $m_p = O_{pi} - \overline{O}_i = O_{pi} - \frac{1}{N}\sum_{p=1}^{N} O_{pi}$

$$n_p = O_{pj} - \overline{O}_j = O_{pj} - \frac{1}{N}\sum_{p=1}^{N} O_{pj} \tag{38}$$

The correlation coefficient between O_{pj} and O_{pi} is,

$$R_{ij} = \frac{\sum_{p=1}^{N} m_p n_p}{\sqrt{\sum_{p=1}^{N} m_p^2} \cdot \sqrt{\sum_{p=1}^{N} n_p^2}} \tag{39}$$

Obviously, $|R_{ij}| \le 1$.

The greater the related degree of hidden nodes value is, the smaller the dispersion of regression is. When the node function is repeated, $|R_{ij}| \to 1$ can be compressed.

If S_i is the divergence of the sample,

$$S_i = \frac{1}{N}\sum_{p=1}^{N} O_{pi}^2 - \overline{O_i^2} \tag{40}$$

If S_i is too small, it shows that the hidden node i has little influence on the network learning.

We set that C_1 and C_2 are the lower limit value. Where, $C_1 \in [0.8, 0.9]$ and $C_2 \in [0.001, 0.01]$.

The rule 1 is as follows.

If $|R_{ij}| \geq C_1$, the hidden nodes i and j in the same layer can be compressed.

The rule 2 is as follows.

If $S_2 < C_2$, the node i can be deleted.

The basic method of learning rate is as follows. In the early process, we give the learning rate with a large value. With learning, the learning rate decreases gradually.

We assume that the learning number is e during the learning process. The maximum iteration number is $e_{\max}$. The n learning rate is $\_(n)$. After each learning, the learning rate is increased.

$$\Delta \_ = \frac{\_(1) - \_(e_{\max})}{e_{\max}} \tag{41}$$

The change of the learning rate is as follows.

$$\_(e) = -(1) - e\frac{\_(1) - \_(e_{\max})}{e_{\max}} \tag{42}$$

The flow chart of the improved BP neural network algorithm is as follows.

5. Experiment

In order to verify the reliability and validity of this method, we use the improved BP neural network method to predict the product inventory for one company. We collect 150 sets of data. The first 130 sets of data are the training data and the last 20 set of data sample data are the sample data. At the same time, in order to explaining that the results of the method are better than other algorithms, we compare the results of the experiment. The comparison of the predicted results and the actual results are as follows.

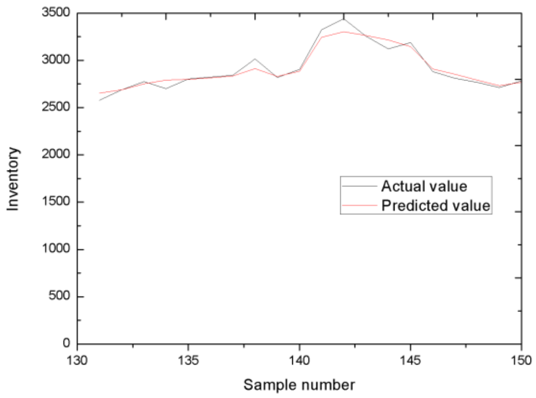

Figure 2. The Comparing of the Actual Value and the Predicted Value

From the above figure, we can see that the trends of two curves are similar. It shows that the predicting results of the improved BP neural network method are accurate and gives the good results.

We compare the results that achieved by the improved BP neural network algorithm with other methods, the results are as follows.

Table 1. The Relative Error of Different Algorithms

Algorithms	Mean relative error
BP neural network	7.11%
Grey algorithm	6.21%
Genetic algorithm	5.45%
GA-BP algorithm	3.56%
Current algorithm	1.31%

We compare the improved BP neural network algorithm with other methods. Then, we can see that the mean relative error that this paper proposes is the smallest. It shows that the method is more accurate to predict the supply chain inventory. The error is small and the predicted results are more ideal. In fact, the method has good applicability and feasibility under the same condition. The prediction results are more accurate.

6. Conclusion

The research of supply chain inventory management in the supply chain research field occupies an important position. It can reduce inventory costs and the risk of supply chain. It also enhances consumer confidence according to predicting the inventory. In order to make a better prediction on inventory, we propose the improved BP neural network algorithm. In this algorithm, we not only use the improved GSA algorithm to optimize the parameters for the BP neural network algorithm, but also improve the shortcomings of the BP neural network algorithm. In this paper, we do the following work. Firstly, we briefly introduce the background of the research. Secondly, we introduce the BP neural network and GSA algorithm. Thirdly, we improve the improved BP neural network algorithm. Finally, we use this algorithm to predict the product inventory and compare with other methods. The results show that this method has more accurate prediction results and has good applicability.

References

[1] Hongyan Dai, Jianbin Li, Nina Yan, Weihua Zhou. Bullwhip effect and supply chain costs with low- and high-quality information on inventory shrinkage [J]. European Journal of Operational Research, 2016, 250: 457-469.
[2] Carl Philip T. Hedenstierna, Stephen M. Disney. Inventory performance under staggered deliveries and autocorrelated demand [J]. European Journal of Operational Research, 2016, 249: 1082-1091.
[3] Samira Saedi, O. Erhun Kundakcioglu, Andrea C. Henry. Mitigating the impact of drug shortages for a healthcare facility: An inventory management approach [J]. European Journal of Operational Research, 2016, 251: 107-123.
[4] Jung Woo Baek, Seung Ki Moon. A production–inventory system with a Markovian service queue and lost sales [J]. Journal of the Korean Statistical Society, 2016, 45: 14-24.
[5] Giacomo Sbrana, Andrea Silvestrini. Random switching exponential smoothing and inventory forecasting [J]. International Journal of Production Economics, 2014,156: 283-294.
[6] Brent D. Williams, Matthew A. Waller, Sanjay Ahire, Gary D. Ferrier. Predicting retailer orders with POS and order data: The inventory balance effect [J]. European Journal of Operational Research,2014, 232: 593–600.
[7] Aris A. Syntetos, M. Zied Babai, Everette S. Gardner Jr..Forecasting intermittent inventory demands: simple parametric methods vs. bootstrapping [J]. Journal of Business Research, 2015, 68: 1746-1752.
[8] Enrique Martin Alcalde, Kap Hwan Kim, Sergi Saurí Marchán. Optimal space for storage yard considering yard inventory forecasts and terminal performance [J]. Transportation Research Part E: Logistics and Transportation Review, 2015, 82: 101-128.
[9] JiSun Shin, Sungshin Kim, Jang-Myung Lee. Production and inventory control of auto parts based on predicted probabilistic distribution of inventory [J]. Digital Communications and Networks,2015, 1: 292–301.
[10] Liu Ting. Safe stock forecast and appraise based on the nerve method and the data envelopment analysis method [D]. Chongqing Jiaotong University, Management Science and Engineering,2008.
[11] Li Kongming. Inventory management and inventory forecast in ship-repair company [D]. Shanghai Jiao Tong University, project management,2013.

[12] Zhao Jun, Wang Xiao. Study on inventory demand forecasting model of TPL Centers based on data mining [J]. , Logistics Technology,2014,33:148-151.
[13] Gao Haojie. The Application research on the forecast of inventory demand based-on back-propagation neural network in the value manufacturing enterprise [D]. Beijing Jiao Tong University, industrial engineering,2015.
[14] Agus Mansur, Triyoso Kuncoro. Product Inventory Predictions at Small Medium Enterprise Using Market Basket Analysis Approach - Neural Networks [J]. Procedia Economics and Finance, 2012,4: 312-320.
[15] Jeong, Bongju. A Computerized Causal Forecasting System using Genetic Algorithm in Supply Chain Management [J]. The Journal of Systems and Software, 2002, Vol.60:223-237.
[16] P.Radhakrishthinan,V.M. Prasad , M.R. Gopalan,Optimizing Inventory Using Genetic Algorithm for Efficient Supply Chain Management [J]. Journal of Computer Science, 2009,5 (3): 233-241,
[17] Feng Yu, Xiaozhong Xu. A short-term load forecasting model of natural gas based on optimized genetic algorithm and improved BP neural network [J]. Applied Energy, 2014, 134:102-113.
[18] Jun Wang, Zheng Sheng, Bihua Zhou, Shudao Zhou. Lightning potential forecast over Nanjing with denoised sounding-derived indices based on SSA and CS-BP neural network [J]. Atmospheric Research, 2014, 137: 245-256.
[19] Wang Dongxu, Shen Yimin, Wang Zhiqiang.Prediction of s ERP safe stocks based on neural network model [J]. Computer application, 2001,3: 51-55.
[20] Li Song, Liu Lijun, Zhai Man. Prediction for short-term traffic flow based on modified PSO optimized BP neural network [J]. System engineering-Theory&Practice, 2012,9: 2045-2049.
[21] Yuebiao Li, Zhiheng Li, Maojing Jin, Shengchao Yin. Multiple-step ahead Traffic Forecasting based on GMM Embedded BP Network [J]. Procedia - Social and Behavioral Sciences, 2013, 96: 1014-1024.
[22] Chao Ren, Ning An, Jianzhou Wang, Lian Li, Bin Hu, Duo Shang. Optimal parameters selection for BP neural network based on particle swarm optimization: A case study of wind speed forecasting [J]. Knowledge-Based Systems, 2014, 56: 226-239.
[23] Yongli Wang, Dongxiao Niu, Li Ji. Short-term power load forecasting based on IVL-BP neural network technology [J]. Systems Engineering Procedia, 2012, 4: 168-174.
[24] Liu Ke, Guo Wenyan, Shen Xiaoliu, Tan Zhongfu. Research on the Forecast Model of Electricity Power Industry Loan Based on GA-BP Neural Network [J]. Energy Procedia, 2012, 14: 1918-1924.

