Research Article

A Hybrid Approach by Integrating Brain Storm Optimization Algorithm with Grey Neural Network for Stock Index Forecasting

Yanqiu Sun

School of Information Engineering, Liaoning University of Traditional Chinese Medicine, Shenyang, Liaoning 110847, China

Correspondence should be addressed to Yanqiu Sun; syq506@163.com

Stock index forecasting is an important tool for both the investors and the government organizations. However, due to the inherent large volatility, high noise, and nonlinearity of the stock index, stock index forecasting has been a challenging task for a long time. This paper aims to develop a novel hybrid stock index forecasting model named BSO-GNN based on the brain storm optimization (BSO) approach and the grey neural network (GNN) model by taking full advantage of the grey model in dealing with data with small samples and the neural network in handling nonlinear fitting problems. Moreover, the new developed BSO-GNN, which initializes the parameters in grey neural network with the BSO algorithm, has great capability in overcoming the deficiencies of the traditional GNN model with randomly initialized parameters through solving the local optimum and low forecasting accuracy problems. The performance of the proposed BSO-GNN model is evaluated under the normalization and nonnormalization preprocessing situations. Experimental results from the Shanghai Stock Exchange (SSE) Composite Index, the Shenzhen Composite Index, and the HuShen 300 Index opening price forecasting show that the proposed BSO-GNN model is effective and robust in the stock index forecasting and superior to the individual GNN model.

1. Introduction

The stock market not only occupies an important position in the financial investments domain but also plays an important role in the financial market researching. Stock index is a significant indicator in reflecting the stock market. It refers to the stock price index and is regarded as a reference indicator which is compiled by the stock exchange corporations or the financial services sectors to indicate the changes of the overall stock prices. Stock index forecasting is an important tool for both the investors and the government organizations. However, the accuracy of the stock index forecasting has been affected by many factors, and it has characteristics such as large volatility, high noise, and nonlinearity. Thus, it is a quite complex and difficult task to provide an accurate stock index forecasting.

For the recent years, researchers have made great effort to develop new forecasting models so as to improve the stock index forecasting accuracy. The application of the neural network models to stock indices forecasting has attracted great attention in the recent few decades. The artificial neural network (ANN) was used by de Faria et al. [1] as an alternative method along with the adaptive exponential smoothing (ES) approach to forecast the Brazilian stock market. Shen et al. [2] surveyed the effectiveness of the radial basis function neural network optimized by artificial fish swarm approach in the stock indices forecasting. Lu and Wu [3] developed a cerebellar model articulation controller neural network, whose performance is better than the support vector regression (SVM) and the backpropagation neural network (BPNN) when it was adopted to forecast the Nikkei 225 and Taiwan Stock Exchange Capitalization Weighted Stock Index closing indices. BPNN was also integrated with the ES algorithm and the autoregressive integrated moving average method by Wang et al. [4] to predict the closing price of the stock index. Nayak et al. [5] also introduced a hybrid model formed by the chemical reaction optimization and ANN to the stock index forecasting. Aside from the neural network models,

fuzzy time series approaches have also achieved considerable successes in stock indices forecasting. A type-2 fuzzy time series approach was applied by Huarng and Yu [6] to the stock index forecasting in 2005, and Singh and Borah [7] enhanced this approach by adopting the particle swarm optimization approach. Chen et al. [8] presented the performance of a high-order fuzzy time series with the assistance of the multiperiod adaption approach in stock markets forecasting. By regarding the stock index as well as trading volume as the forecasting factors, a dual-factor fuzzy time series approach was employed by Chu et al. [9] to forecast the stock index. Wong et al. [10] also proposed an adaptive time-variant fuzzy time series model to perform the stock index forecasting. Stock indices forecasting involving other models can be found in [11–14].

Although a large number of models have been developed by researchers to forecast the stock indices, there is no model that can perform well in any situations; thus, novel effective stock indices forecasting approaches still need to be explored. This paper aims to propose a novel hybrid stock index forecasting model. As known, the grey model has great capability in dealing with data with small samples, and the neural network has strong abilities in handling nonlinear fitting problems. Thus, with the consideration that combining these two methods can make up the deficiency of the single models, a grey neural network (GNN) is developed in this paper to forecast the daily stock index price. However, it is found that the grey neural network model with randomly initialized parameters has two deficiencies: on the one hand, it is easy to fall into local optimum; on the other hand, the forecasting accuracy is generally low. Therefore, a new brain storm optimization algorithm-based grey neural network (BSO-GNN) approach, which initializes the parameters in grey neural network with the brain storm optimization (BSO) algorithm, is proposed in this paper to solve the local optimum and low forecasting accuracy problems. The effectiveness of the new proposed BSO-GNN model is validated by the 5-day-ahead daily opening price forecasting results of three stock market indices named the Shanghai Stock Exchange (SSE) Composite Index, the Shenzhen Composite Index, and the HuShen 300 Index.

The remainder of this paper is organized as follows. In Section 2, the related methodologies including the single GNN model and the single BSO algorithm as well as the combined BSO-GNN approach are introduced. The performance measurement criteria of different models can be also found in this section. Section 3 presents the data collection and parameters initialization results. The simulation results and discussions are exhibited in Section 4. And the last section concludes the whole paper.

2. Related Methodologies

2.1. Individual Models

2.1.1. Grey Neural Network Model.
The grey neural network (GNN) model [15] integrates both the advantages of the grey model and the neural network. Given a data sequence
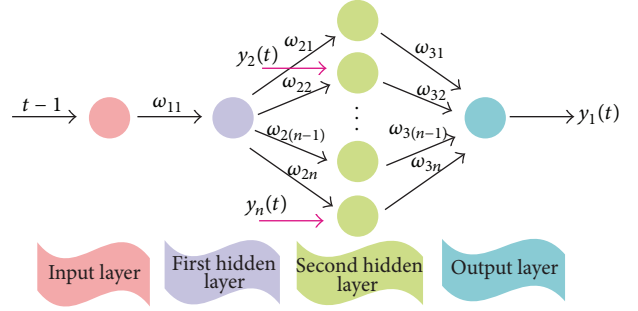

Figure 1: Structure of the GNN model.

$x(1), x(2), \ldots, x(T)$, where T is the number of the data in this sequence, then a new data sequence can be generated based on the 1-accumulated generating operation (1-AGO) as $x^{(1)}(1), x^{(1)}(2), \ldots, x^{(1)}(T)$, where $x^{(1)}(t) = \sum_{k=1}^{t} x(k)$, $t = 1, 2, \ldots, T$.

The GNN model involving n parameters is expressed as

$$\frac{dx_1^{(1)}}{dt} + b_1 x_1^{(1)} = b_2 x_2^{(1)} + b_3 x_3^{(1)} + \cdots + b_n x_n^{(1)}, \quad (1)$$

where $x_2^{(1)}, x_3^{(1)}, \ldots, x_n^{(1)}$ are the input variables, $x_1^{(1)}$ is the output variable, and $b_1, b_2, \ldots, b_n$ are the parameters that need to be predetermined. To simplify the symbols, let $y_i = x_i^{(1)}$, $i = 1, 2, \ldots, n$. According to (1), the following expression is obtained:

$$y_1(t) = \left[y_1(1) - \lambda \right] e^{-b_1(t-1)} + \lambda, \quad (2)$$

where

$$\lambda = \frac{b_2}{b_1} y_2(t) + \frac{b_3}{b_1} y_3(t) + \cdots + \frac{b_n}{b_1} y_n(t). \quad (3)$$

So formula (2) is equivalent to

$$\begin{aligned}
y_1(t) &= \left[\left(y_1(1) - \lambda \right) \frac{e^{-b_1(t-1)}}{1 + e^{-b_1(t-1)}} + \frac{\lambda}{1 + e^{-b_1(t-1)}} \right] \\
&\quad \times \left(1 + e^{-b_1(t-1)} \right) \\
&= \left[\left(y_1(1) - \lambda \right) \left(1 - \frac{1}{1 + e^{-b_1(t-1)}} \right) + \frac{\lambda}{1 + e^{-b_1(t-1)}} \right] \\
&\quad \times \left(1 + e^{-b_1(t-1)} \right) \\
&= \left[\left(y_1(1) - \lambda \right) - \frac{y_1(1)}{1 + e^{-b_1(t-1)}} + \frac{2\lambda}{1 + e^{-b_1(t-1)}} \right] \\
&\quad \times \left(1 + e^{-b_1(t-1)} \right).
\end{aligned} \quad (4)$$

According to (4), a 4-layer GNN structure presented in Figure 1 can be constructed as follows.

Step 1. Determine the values of parameters $b_1, b_2, \ldots, b_n$.

Step 2. Calculate the weights by using $\omega_{11} = b_1$, $\omega_{21} = -y_1(1)$, $\omega_{22} = 2b_2/b_1$, $\omega_{23} = 2b_3/b_1, \ldots, \omega_{2n} = 2b_n/b_1$, $\omega_{31} = \omega_{32} = \cdots = \omega_{3n} = 1 + e^{-b_1(t-1)}$ and let $\theta = (y_1(1) - \lambda) * (1 + e^{-b_1(t-1)})$.

Step 3. For each input-output pair $((t-1), y_1(t))$, $t = 1, 2, \ldots, T$, determine the output in each layer by using the following activation functions:

(a) from the input layer to the first hidden layer: the logistic sigmoid (logsig) function is used; that is, the output in the first hidden layer is expressed as $z_{11} = 1/(1 + e^{-\omega_{11}(t-1)})$;

(b) from the first hidden layer to the second hidden layer: the linear function is used; that is, the output in the second hidden layer is expressed as $z_{21} = \omega_{21} z_{11}$, $z_{2i} = \omega_{2i} z_{11} y_i(t)$, $i = 2, \ldots, n$;

(c) from the second hidden layer to the output layer: the linear function is used; that is, the output in the output layer is expressed as $z_{31} = \sum_{i=1}^{n} \omega_{3i} z_{2i} + \theta$.

Step 4. Calculate the error between the target output $y_1(t)$ and the calculated output z_{31}, and update the weight according to the following expressions:

(a) the weights from the first hidden layer to the second hidden layer are updated by $\omega_{21} = -y_1(1)$, $\omega_{22} = \omega_{22} - \eta_2 \delta_2 z_{11}$, $\omega_{23} = \omega_{23} - \eta_3 \delta_3 z_{11}, \ldots, \omega_{2n} = \omega_{2n} - \eta_n \delta_n z_{11}$, where $\delta_1 = \delta_2 = \delta_3 = \cdots = \delta_n = (z_{31} - y_1(t))(1 + e^{-\omega_{11}(t-1)})$ and $\eta_2, \eta_3, \ldots, \eta_n$ are called the learning rate;

(b) the weights from the input layer to the first hidden layer are updated by $\omega_{11} = \omega_{11} + b_1 t \delta_{n+1}$, where $\delta_{n+1} = (1 + e^{-\omega_{11}(t-1)})^{-1}(1 - 1/(1 + e^{-\omega_{11}(t-1)})) \sum_{i=1}^{n} \omega_{2i} \delta_i$;

(c) meanwhile, update θ as

$$\theta = \left(1 + e^{-\omega_{11}(t-1)}\right) \\ \times \left(\frac{(\omega_{22} y_2(t) + \omega_{23} y_3(t) + \cdots + \omega_{2n} y_n(t))}{2} - y_1(1)\right). \tag{5}$$

Step 5. Judge whether the termination criterion has been reached: if so, the whole GNN model has been determined; otherwise, return to Step 3 and repeat the process.

Once the sequence $y_1(2), \ldots, y_1(T)$ has been forecasted by the GNN model, the forecasted values of the original data sequence $x(1), x(2), \ldots, x(T)$ can be obtained by the 1-inverse AGO (1-IAGO) defined by

$$x(1) = y_1(1) \\ x(i) = y_1(i) - y_1(i-1), \quad i = 2, 3, \ldots, T. \tag{6}$$

2.1.2. Brain Storm Optimization Algorithm.
Parameter optimization is one of the effective ways to improve the accuracy of forecasting approaches [16–19]. In this paper, the brain storm optimization (BSO) algorithm [20] is used to optimize the unknown parameters in the GNN model. For a population of n individuals $P_i = (P_{i,1}, P_{i,2}, \ldots, P_{i,s})$, $i = 1, 2, \ldots, n$, where s is the dimension of the individuals, the BSO algorithm consists of seven procedures in all, which is described as follows.

Step 1. Initialize the population of individuals by using $P_{i,j} = l + (r - l) * \text{rand}$, where l and r are the left boundary and the right boundary of the parameter, and rand is a random value selected from the unit interval $(0, 1)$.

Step 2. Divide these n individuals into m clusters with the means cluster method.

Step 3. Calculate the n objective function values with regard to the n individuals, and pick up the individual which corresponds to the smallest objective function as the clustering centroid in each cluster.

Step 4. Choose a random value r_1 from the interval $(0, 1)$, and compare this value with a predetermined probability p_1:

(a) if $r_1 < p_1$, randomly select one clustering centroid and replace it with a randomly generated individual;

(b) otherwise, go to Step 5.

Step 5. Update the individuals. First randomly choose a value r_2 from $(0, 1)$:

(a) if r_2 is less than a predetermined probability p_2, then select a random value r_3 which follows the uniform distribution on $(0, 1)$:

(a1) if r_3 is less than a given probability p_3, choose the clustering centroid and add random values to it to update the individuals;

(a2) otherwise, choose an individual randomly from this cluster and update it by adding random value to it;

(b) otherwise, select a random value r_4 from $(0, 1)$:

(b1) if r_4 is less than a given probability p_4, first choose two clusters randomly and combine them; then add random values to update the individual;

(b2) otherwise, randomly choose two individuals from two randomly selected clusters and combine them; then add random values to update the individual.

After this, compare the objective function values obtained by the new updated individuals and the corresponding original individuals; then reserve the better one and regard it as the new individual.

Step 6. Repeat Step 5 until the n individuals have been all updated.

Step 7. Judge whether the termination criterion has been reached: if yes, end the whole process; otherwise, go to Step 2 and repeat the process.
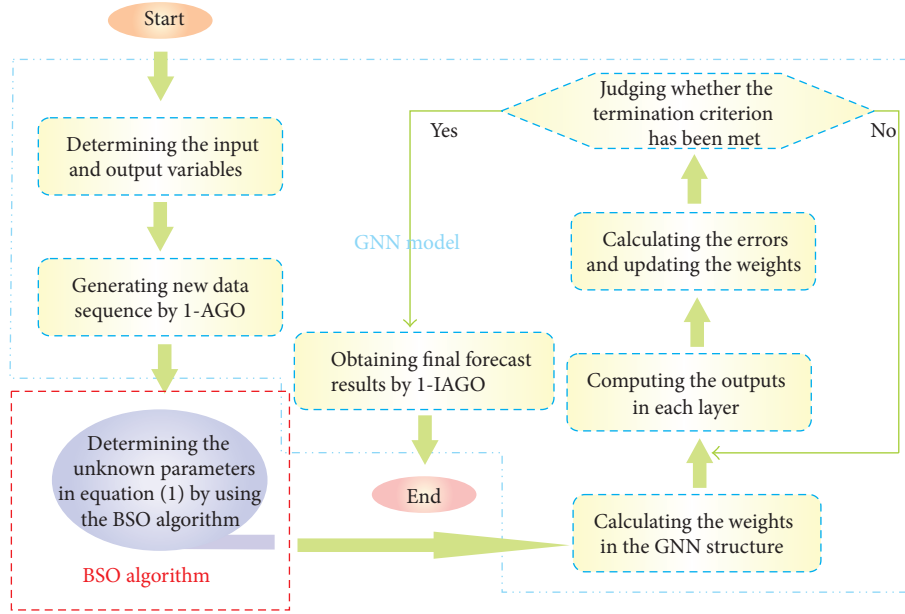
FIGURE 2: Flowchart of the new proposed BSO-GNN approach.

As observed, the individual update in Step 5 is of great importance in conducting the BSO process. The update of an individual in this step is carried out by

$$P_{\text{updated}} = P_{\text{selected}} + \phi * n(\mu, \sigma),\qquad(7)$$

where P_{selected} and P_{updated} are the selected and updated individuals, respectively, $n(\mu, \sigma^2)$ is the normal random function of which the mean is μ and the variance is σ^2, and the coefficient ϕ is determined by

$$\phi$$
$$= \frac{1}{1 + \exp\left[(0.5 * \text{max\_iteration} - \text{current\_iteration})/k\right]}$$
$$* \text{rand},$$
$$(8)$$

where max_iteration and current_iteration denote the maximum iteration number and the current iteration number, respectively, rand is a value randomly chosen from the interval $(0, 1)$, and k is a predetermined value.

2.2. New Combined Model. As observed, in Step 1 of constructing the GNN model, values of $b_1, b_2, \ldots, b_n$ should be first determined. However, in previous studies, values assigned to these parameters are randomly selected or just determined by experience; this will greatly affect the accuracy of the GNN model. Therefore, this paper develops a new BSO-GNN model, which utilizes the BSO algorithm to optimize these parameters and the optimal parameters are substituted for (1) to conduct the GNN model construction. In this manner, error caused by the improper parameter values can be reduced a lot, and the whole forecasting error is expected to be lowered. Flowchart of the new combined BSO-GNN model is shown in Figure 2.

2.3. Performance Evaluation Criteria. To evaluate the performance of different models, three error evaluation criteria named the mean absolute error (MAE) [21], the root mean square error (RMSE) [22], and the mean absolute percentage error (MAPE) [23], which are defined as follows, respectively, have been adopted:

$$\text{MAE} = \frac{1}{T} \sum_{i=1}^{T} |x(i) - \hat{x}(i)|,$$

$$\text{RMSE} = \sqrt{\frac{1}{T} \sum_{i=1}^{T} (x(i) - \hat{x}(i))^2},\qquad(9)$$

$$\text{MAPE} = \frac{1}{T} \sum_{i=1}^{T} \left| \frac{x(i) - \hat{x}(i)}{x(i)} \right| \times 100\%,$$

where $x(i)$ and $\hat{x}(i)$ are the actual value and the forecasted value at time i, respectively.

3. Data Collection and Parameters Initialization

In this paper, five indicators with regard to three stock market indices named the Shanghai Stock Exchange (SSE) Composite Index (http://q.stock.sohu.com/zs/000001/lshq .shtml), the Shenzhen Composite Index (http://q.stock.sohu .com/zs/399106/lshq.shtml), and the HuShen 300 Index (http://q.stock.sohu.com/zs/000300/lshq.shtml) are collected to evaluate the performance of the new proposed BSO-GNN model. Herein the five indicators consist of the opening price (OP), the closing price (CP), the highest price (HP), the lowest price (LP), and the trading volume (TV). All the five indicators of these three stock market indices are collected

Table 1: Statistical characteristics of the three stock indices.

Index	Variable	Number	Minimum	Maximum	Mean	Std.[a]	Skewness	Kurtosis
SSE Composite Index	OP	66	2148.15	2464.92	2338.63	74.2912	−0.466	−0.056
	CP	66	2148.45	2460.69	2342.81	70.0056	−0.448	0.088
	HP	66	2132.63	2444.63	2322.36	74.6363	−0.457	−0.158
	LP	66	2164.32	2478.38	2358.21	69.0148	−0.385	0.114
	TV	66	45859524.00	179786032.00	87257856.18	24217788.4145	1.139	2.822
Shenzhen Composite Index	OP	66	811.14	1014.16	918.76	52.33264	−0.288	−0.970
	CP	66	813.99	1010.46	921.17	51.00838	−0.352	−0.889
	HP	66	797.17	996.81	909.91	52.98899	−0.380	−0.938
	LP	66	817.78	1020.29	929.26	50.19085	−0.272	−0.980
	TV	66	31650934.00	123093328.00	61848462.18	17652160.18852	0.759	1.338
HuShen 300 Index	OP	66	2274.35	2694.48	2528.56	96.66418	−0.606	0.190
	CP	66	2276.39	2681.07	2533.82	90.90981	−0.604	0.332
	HP	66	2254.57	2660.96	2508.58	97.34628	−0.613	0.059
	LP	66	2291.89	2705.75	2553.97	89.09531	−0.572	0.467
	TV	66	30527144.00	108426536.00	53832080.73	15420163.9229	1.199	2.328

[a]Std. refers to the standard deviation.

from January 5, 2012, to April 17, 2012. Since the stock trading is not made on weekends and during the national holidays, the number of the data during the previous collection period presented in the form of $(OP_t, CP_t, HP_t, LP_t, TV_t)$ is only 66, where the integer t varies from 1 to 66 in the case of regarding January 5, 2012, as the first day. And the purpose of this paper is to forecast the 5-day-ahead daily opening price of the stock.

In general, it is believed that the opening price of the stock on one special day is relevant to the five indicators of the corresponding stock on the former day. Therefore, the GNN model constructed in this paper is as follows:

$$\frac{dOP_d}{dt} + b_1 OP_d = b_2 OP_{d-1} + b_3 CP_{d-1} + b_4 HP_{d-1}$$
$$+ b_5 LP_{d-1} + b_6 TV_{d-1}, \tag{10}$$
$$(d = 2, \ldots, 66),$$

where the coefficient b_i $(i = 1, 2, \ldots, 6)$ is unknown and needs to be determined by the BSO algorithm, the output variable OP_d in the GNN model denotes the opening price of the stock on the dth day, and the input variables OP_{d-1}, CP_{d-1}, HP_{d-1}, LP_{d-1}, and TV_{d-1} in the GNN model represent the opening price, the closing price, the highest price, the lowest price, and the trading volume of the corresponding stock on the $(d-1)$th day, that is, on the former day, respectively. In this manner, there are 65 input-output pairs in all (because the subscript d varies from 2 to 66). These 65 input-output pairs are divided into two parts: the training set and the testing set, where the number of the two parts is set to 60 and 5, respectively; that is, the first 60 input-output pairs are regarded as the training set, while the last 5 pairs are employed to validate the performance of the new proposed model. In other words, this paper is devoted to the 5-day-ahead daily opening price forecasting. According to the GNN algorithm introduced in

Section 2.1.1, the number of the output in the GNN model is 1, so the 5-day-ahead daily opening price forecasting is finished by carrying out the GNN model five times in all.

In this paper, a $1 \times 1 \times 6 \times 1$ network is constructed. Parameters involving the GNN model in this paper are initialized as follows: $\eta_2 = \eta_3 = \cdots = \eta_6 = 0.0015$. When it comes to the BSO algorithm, the related parameters are initialized with $n = 100$, $s = 5$, $p_1 = 0.2$, $p_2 = 0.8$, $p_3 = 0.4$, $p_4 = 0.5$, $k = 20$, $\mu = 0$, $\sigma^2 = 1$, and max_iteration = 200.

4. Simulations Results and Discussions

4.1. Basis Statistical Characteristic Analysis. In this section, the basic statistical characteristics of the three stock indices are first analyzed and the results are provided in Table 1.

The mean figures listed in Table 1 demonstrate that the variation ranges of the SSE Composite Index and the HuShen 300 Index revealed by the five indicators, that is, the opening price, the closing price, the highest price, the lowest price, and the trading volume, are much similar to each other, however, differ a lot from those of the Shenzhen Composite Index. This phenomenon is clearly shown in Figure 3. And the standard deviation values of the first four indicators can be ordered as the HuShen 300 Index, the SSE Composite Index, and the Shenzhen Composite Index, from large to small, respectively. Among these three stock markets, the standard deviation values of the fifth indictor (i.e., the trading volume) in the SSE Composite Index market are the greatest and in the HuShen 300 Index market are the least. The skewness statistic is adopted to describe the symmetry of the variables, and the greater the absolute value of the skewness, the more asymmetric the variable. So, as observed, for a specific variable among the five indicators, the most relative symmetric is the Shenzhen Composite Index, while the most relative asymmetric is the HuShen 300 Index. In addition,
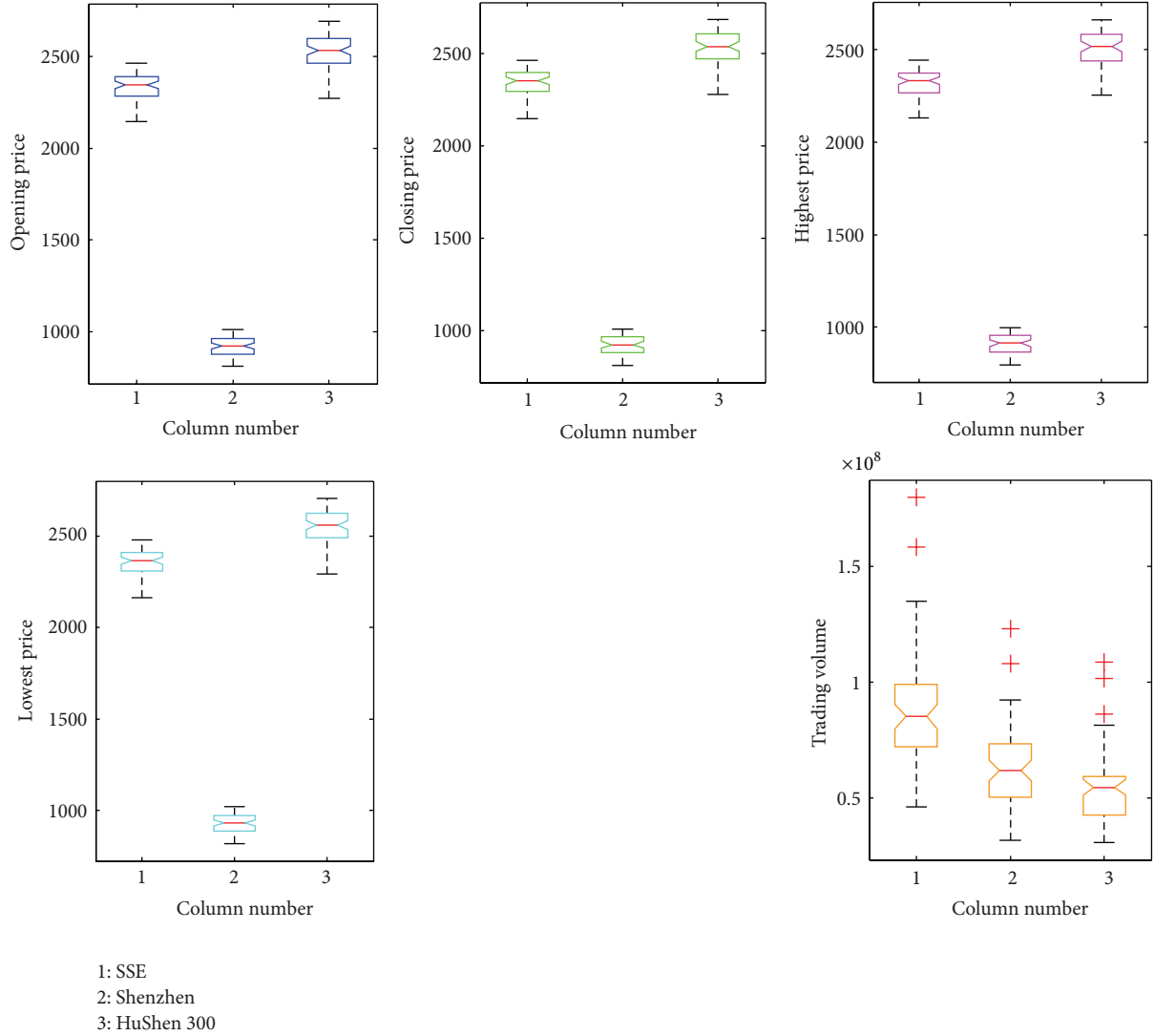
1: SSE
2: Shenzhen
3: HuShen 300

FIGURE 3: Box graphs of the five indictors in the three stock markets.

TABLE 2: Optimal parameters obtained by the BSO algorithm.

Index name	b_1	b_2	b_3	b_4	b_5	b_6
SSE Composite Index	4.6375	0.1510	3.3261	0.3365	0.5815	0.1790
Shenzhen Composite Index	4.2021	0.6432	2.2083	0.9525	−0.0108	0.3482
HuShen 300 Index	5.4745	0.2125	3.0149	1.1007	0.7272	0.4550

a statistic which is used to describe the steep degree of the variables named the kurtosis is also introduced. For this statistic, values large than 0 reveal that the distribution of the variable values is steeper than the standard Gaussian distribution; on the contrary, values smaller than 0 mean that the distribution of the data is less steep than the standard Gaussian distribution; and the optimal situation occurs when the value of this statistic is equal to 0, which shows that values have the same distribution with the standard Gaussian distribution [24].

4.2. Results Obtained by BSO-GNN and GNN Models. In this section, the performance of the new proposed BSO-GNN model is validated by comparing the fitting results (corresponding to the training set) and the forecasting results (corresponding to the testing set) obtained by the BSO-GNN model and the individual GNN model.

The new BSO-GNN model proposed is carried out by performing the BSO algorithm first to determine the unknown parameters presented in (10); the optimal values with regard to the six parameters are listed in Table 2. As seen
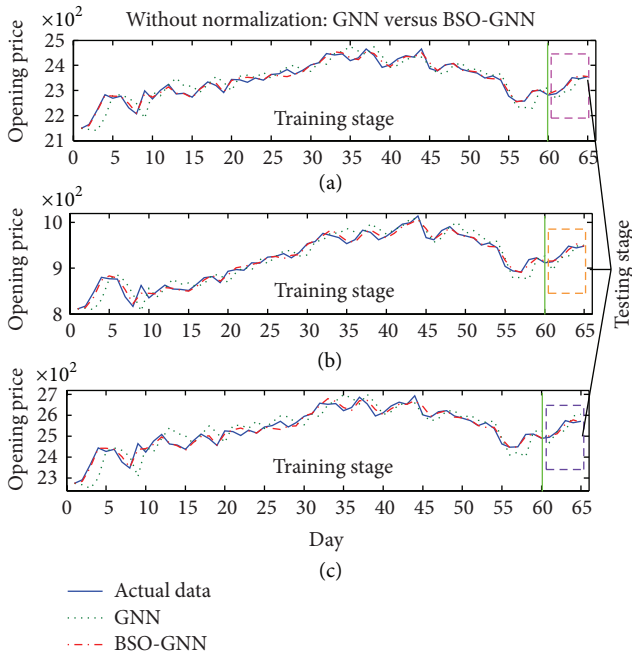
FIGURE 4: Opening price fitting and forecasting results of the three stock market indices gained by the individual GNN model and the BSO-GNN model under the nonnormalization situation: (a) SSE Composite Index, (b) Shenzhen Composite Index, and (c) HuShen 300 Index.
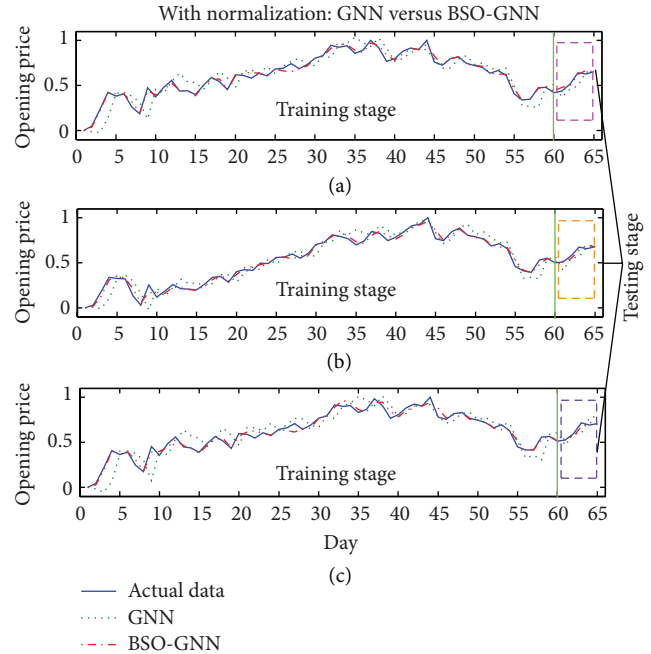

FIGURE 5: Opening price fitting and forecasting results of the three stock market indices gained by the individual GNN model and the BSO-GNN model under the normalization situation: (a) SSE Composite Index, (b) Shenzhen Composite Index, and (c) HuShen 300 Index.

from Table 2, apart from the negative value (-0.0108) of the parameter b_5 for the Shenzhen Composite Index, all the other parameter values are positive.

Different from the BSO-GNN model, the individual GNN model is constructed based on the following randomly selected parameters: $b_i = 0.3 + \text{rand}/4$, $i = 1, 2, \ldots, 6$, where rand is a value randomly chosen from the interval $(0, 1)$.

Due to the reason that different indicators may differ in units and variation intervals, the normalization preprocessing operation is often introduced into the neural network models. Taking the opening price for an example, the following expression is used to normalize this variable: $\text{OP}_{d\_\text{processed}} = (\text{OP}_d - \text{OP}_{\min})/(\text{OP}_{\max} - \text{OP}_{\min})$, where OP_d and $\text{OP}_{d\_\text{processed}}$ are the raw and normalized opening price values on the dth day, respectively, and $\text{OP}_{\min}$ and $\text{OP}_{\max}$ are the minimum and maximum opening price values of all the days, respectively. When the normalization operation is performed in the neural network, the values obtained by the final output of the network should be reproduced by the inverse normalization operation, that is, by carrying out the operation expressed by $\text{OP}_d = \text{OP}_{d\_\text{processed}}(\text{OP}_{\max} - \text{OP}_{\min}) + \text{OP}_{\min}$.

Figures 4 and 5 present the opening price fitting and forecasting results of the three stock market indices gained by the BSO-GNN and the individual GNN models wherein the data are not preprocessed and preprocessed by the normalization operation, respectively. As seen, on some days, curves obtained by the GNN model have an opposite trend with the actual curve; the same situation occurs in curves gained

by BSO-GNN. However, as a whole, in both the without-normalization and with-normalization cases, as compared to the individual GNN model, the fitting and forecasting results obtained by BSO-GNN models for all the three stock market indices are much closer to the actual data on most of the days.

4.3. Error Comparison. It is difficult to judge which model is better by only observing fitting and forecasting results presented in Figures 4 and 5. Thus, in this section, error comparison results obtained by the three error evaluation criteria indictors mentioned in Section 2.3 are provided to gain a much clearer recognition about the good capability of the new proposed BSO-GNN model.

One of the most common comparison methods is to compare the mean values obtained by different models and by the actual data. Figure 6 shows the mean comparison results of the GNN and BSO-GNN models in the training stage and the testing stage. However, results have shown that there is no significant difference between the mean obtained by the GNN model and the mean calculated by the actual data or between the mean obtained by the BSO-GNN model and the mean calculated by the actual data. Thus, a further comparison should be conducted by using other indicators, and error evaluation by using criteria described in Section 2.3 is just a good approach to solve this problem.

As observed, when the raw data is preprocessed by the normalization operation, $\text{OP}_{d\_\text{processed}}$ will be equal to 0 in the case of $\text{OP}_d = \text{OP}_{\max}$. However, in this situation, the MAPE error with regard to the data sequence constructed by
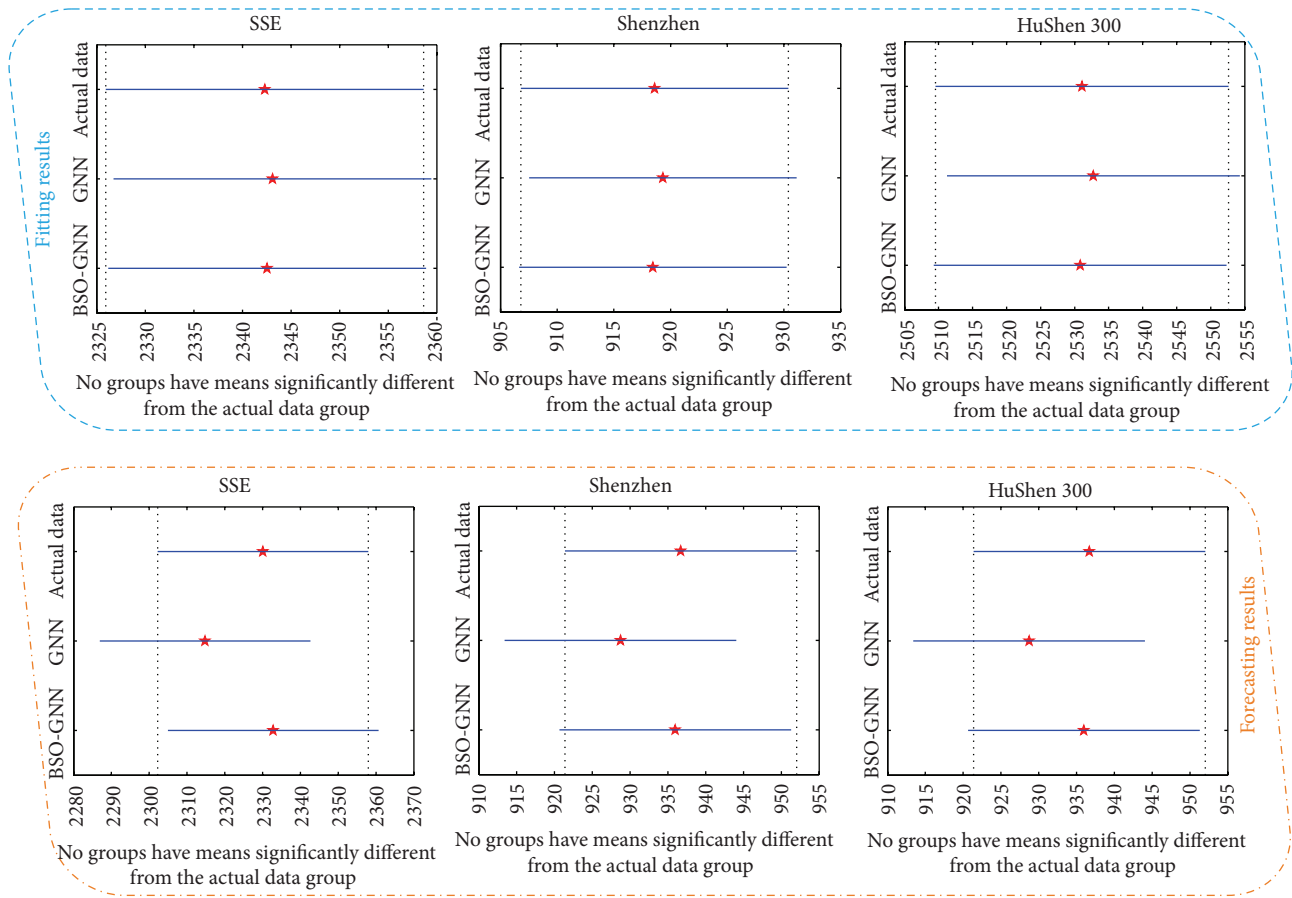
Figure 6: Mean comparison results of the GNN and BSO-GNN models.

Table 3: Fitting error results of different models under different data preprocessing operations.

Data preprocessing	Error type	Model	Training stage		
			SSE Composite Index	Shenzhen Composite Index	HuShen 300 Index
Without normalization	MAE	GNN	26.0267	14.2249	36.0929
		BSO-GNN	6.4482	5.3152	10.8070
	RMSE	GNN	34.4066	18.5165	49.4477
		BSO-GNN	9.0112	6.8256	14.8538
	MAPE	GNN	1.1191	1.5722	1.4410
		BSO-GNN	0.2740	0.5813	0.4247
With normalization	MAE	GNN	0.0822	0.0701	0.0859
		BSO-GNN	0.0204	0.0262	0.0257
	RMSE	GNN	0.1086	0.0912	0.1177
		BSO-GNN	0.0285	0.0336	0.0354
	MAPE	GNN	19.2278[b]	28.3084[b]	20.6714[b]
		BSO-GNN	4.0525[b]	9.4707[b]	5.2068[b]

[b]Corrected value.

$OP_{d\_processed}$ will make no sense since the denominator in the definition of MAPE is equal to 0. So we correct the MAPE error by eliminating the data satisfying $OP_d = OP_{max}$ and just calculating MAPE error on the data sequence constructed by other data in this case. Table 3 exhibits the fitting error results of GNN and BSO-GNN models under different data preprocessing operations, that is, with normalization and without normalization.

A crucial phenomenon revealed in Table 3 is that the error values obtained by using the same error evaluation criterion but different preprocessing operations, that is, with normalization situation and without normalization, differ a
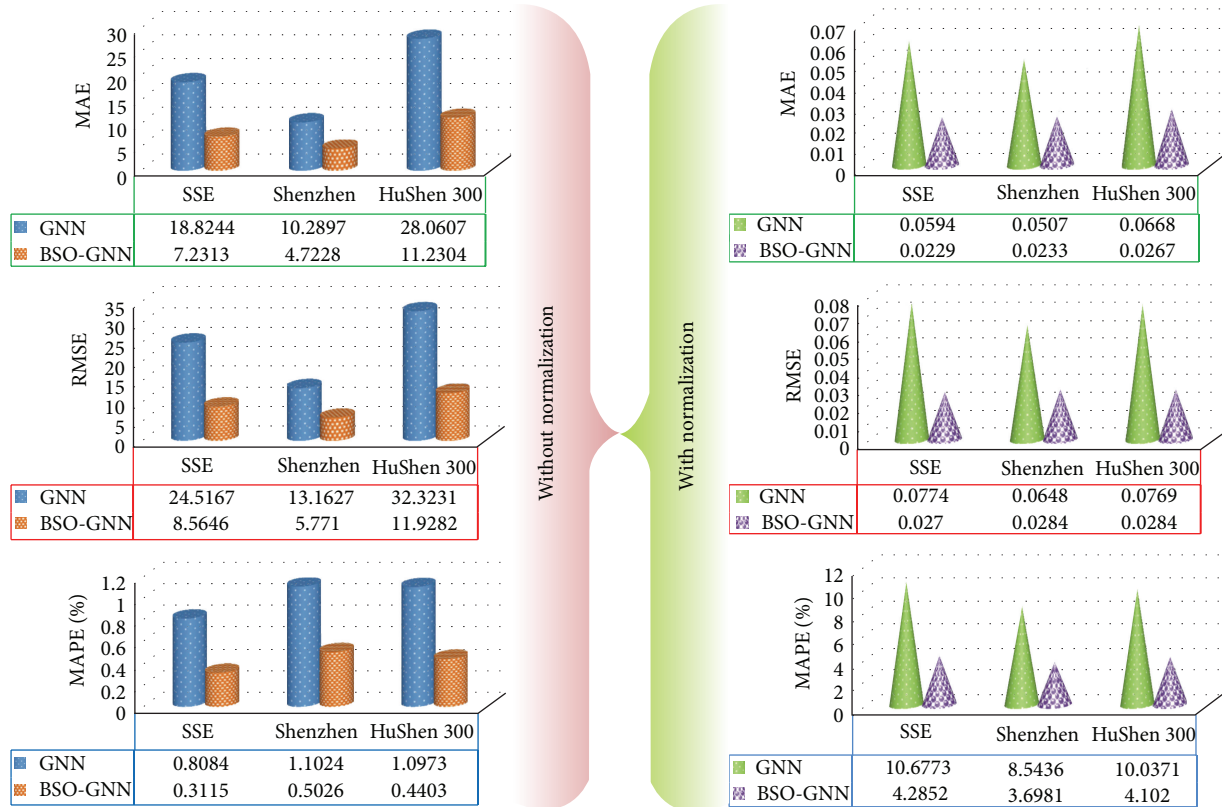
Figure 7: Forecasting errors of the GNN and BSO-GNN models under different data preprocessing operations.

lot: error values obtained by the MAE and RMSE criteria vary in the interval $(5, 50)$ in the without-normalization situation, while the variation interval changes to $(0.02, 0.12)$ in the with-normalization situation. When it comes to the MAPE criterion, the corresponding variation intervals in these two cases are $(0.2, 1.6)$ and $(4, 29)$, respectively. In other words, the normalization operation has a significant impact on the variation range of the error values.

Furthermore, Table 3 demonstrates that for all the three stock market indices, the MAE, RMSE, and MAPE values obtained by the BSO-GNN model are all smaller than those obtained by the individual GNN model no matter whether the raw data are preprocessed by the normalization operation or not. In the case where the raw data are not preprocessed by the normalization operation, the greatest improvements of the BSO-GNN as compared to the GNN models calculated according to the MAE, RMSE, and MAPE criteria are all obtained by the HuShen 300 Index with values of 25.1897 (from 36.0929 to 25.1859), 34.5939 (from 49.4477 to 14.8538), and 1.0163 (from 1.4410 to 0.4247), respectively. While results in the case where the raw data are preprocessed by the normalization operation are different, the most significant improvements obtained according to the MAE, RMSE, and MAPE criteria occur in the SSE Composite Index, the HuShen 300 Index, and the Shenzhen Composite Index, respectively.

Analogously, a clear view with regard to the forecasting error results is provided in Figure 7. As seen, with no exception, the error results gained according to the BSO-GNN model are less than those obtained by the GNN model. And consistent with the results obtained in the training stage, under the without-normalization situation, the greatest improvements demonstrated by the MAE, RMSE, and MAPE criteria all occur in the HuShen 300 Index market, with values of 16.8303 (from 28.0607 to 11.2304), 20.3949 (from 32.3231 to 11.9282), and 0.6570 (from 1.0973 to 0.4403), respectively. As turn to the with-normalization case, there is some difference as compared to the error results obtained in the training stage: the most significant improvement demonstrated by the MAE criterion appears in the HuShen 300 Index market, with an improvement of 0.0401 (from 0.0668 to 0.0267), while those demonstrated by the RMSE and MAPE criteria occur in the SSE Composite Index market, with values of 0.0504 (from 0.0774 to 0.0270) and 6.3921 (from 10.6773 to 4.2852), respectively.

5. Conclusions

In this paper, a hybrid stock index forecasting model BSO-GNN is proposed based on the BSO algorithm and the GNN approach, which utilizes the advantages of the BSO method, the grey model, and the neural network. The normalization operation has been frequently introduced into the neural

network. And this paper presents a view on whether the normalization operation can affect the forecasting results by considering two situations in evaluating the performance of the BSO-GNN model furthermore, that is, without-normalization and with-normalization. In addition, different from the previous GNN models adopted by researchers, the unknown parameters in GNN models, which should be predetermined so as to gain the weights values in the networks, are optimized by the BSO algorithm to guarantee that the error is as low as possible. To evaluate the performance of the new developed model, three stock market indices including the SSE Composite Index, the Shenzhen Composite Index, and the HuShen 300 Index are used. The fitting and forecasting results obtained in the training stage and testing stage reveal that the improved BSO-GNN model is more effective than the individual GNN model in the stock index forecasting. Moreover, the forecasting results and error comparison results demonstrate that the BSO-GNN is effective and robust in the stock index forecasting.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

References

[1] E. L. de Faria, M. P. Albuquerque, J. L. Gonzalez, J. T. P. Cavalcante, and M. P. Albuquerque, "Predicting the Brazilian stock market through neural networks and adaptive exponential smoothing methods," *Expert Systems with Applications*, vol. 36, no. 10, pp. 12506–12509, 2009.

[2] W. Shen, X. Guo, C. Wu, and D. Wu, "Forecasting stock indices using radial basis function neural networks optimized by artificial fish swarm algorithm," *Knowledge-Based Systems*, vol. 24, no. 3, pp. 378–385, 2011.

[3] C.-J. Lu and J.-Y. Wu, "An efficient CMAC neural network for stock index forecasting," *Expert Systems with Applications*, vol. 38, no. 12, pp. 15194–15201, 2011.

[4] J.-J. Wang, J.-Z. Wang, Z.-G. Zhang, and S.-P. Guo, "Stock index forecasting based on a hybrid model," *Omega*, vol. 40, no. 6, pp. 758–766, 2012.

[5] S. C. Nayak, B. B. Misra, and H. S. Behera, "Hybridzing chemical reaction optimization and artificial neural network for stock future index forecasting," in *Proceedings of the 1st International Conference on Emerging Trends and Applications in Computer Science (ICETACS '13)*, pp. 130–134, 2013.

[6] K. Huarng and H.-K. Yu, "A type 2 fuzzy time series model for stock index forecasting," *Physica A: Statistical Mechanics and Its Applications*, vol. 353, no. 1–4, pp. 445–462, 2005.

[7] P. Singh and B. Borah, "Forecasting stock index price based on M-factors fuzzy time series and particle swarm optimization," *International Journal of Approximate Reasoning*, vol. 55, no. 3, pp. 812–833, 2014.

[8] T.-L. Chen, C.-H. Cheng, and H.-J. Teoh, "High-order fuzzy time-series based on multi-period adaptation model for forecasting stock markets," *Physica A: Statistical Mechanics and Its Applications*, vol. 387, no. 4, pp. 876–888, 2008.

[9] H.-H. Chu, T.-L. Chen, C.-H. Cheng, and C.-C. Huang, "Fuzzy dual-factor time-series for stock index forecasting," *Expert Systems with Applications*, vol. 36, no. 1, pp. 165–171, 2009.

[10] W.-K. Wong, E. Bai, and A. W.-C. Chu, "Adaptive time-variant models for fuzzy-time-series forecasting," *IEEE Transactions on Systems, Man, and Cybernetics B: Cybernetics*, vol. 40, no. 6, pp. 1531–1542, 2010.

[11] S.-C. Huang and T.-K. Wu, "Integrating GA-based time-scale feature extractions with SVMs for stock index forecasting," *Expert Systems with Applications*, vol. 35, no. 4, pp. 2080–2088, 2008.

[12] L. Yu, H. Chen, S. Wang, and K. K. Lai, "Evolving least squares support vector machines for stock market trend mining," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 1, pp. 87–102, 2009.

[13] S.-C. Huang, "Forecasting stock indices with wavelet domain kernel partial least square regressions," *Applied Soft Computing Journal*, vol. 11, no. 8, pp. 5433–5443, 2011.

[14] L.-J. Kao, C.-C. Chiu, C.-J. Lu, and C.-H. Chang, "A hybrid approach by integrating wavelet-based feature extraction with MARS and SVR for stock index forecasting," *Decision Support Systems*, vol. 54, no. 3, pp. 1228–1244, 2013.

[15] MATLAB Chinese Forum, "30 cases analysis related to the neural networks in MATLAB," Beihang University Press, 2010, (Chinese).

[16] A. M. Elaiw, X. Xia, and A. M. Shehata, "Combined heat and power dynamic economic dispatch with emission limitations using hybrid DE-SQP method," *Abstract and Applied Analysis*, vol. 2013, Article ID 120849, 10 pages, 2013.

[17] X. Yin, Y. Sun, and C. Wang, "Positioning errors predicting method of strapdown inertial navigation systems based on PSO-SVM," *Abstract and Applied Analysis*, vol. 2013, Article ID 737146, 7 pages, 2013.

[18] W. Zhang, Z. Su, H. Zhang, Y. Zhao, and Z. Zhao, "Hybrid wind speed forecasting model study based on SSA and intelligent optimized algorithm," *Abstract and Applied Analysis*, vol. 2014, Article ID 693205, 14 pages, 2014.

[19] L. Liu, Q. Wang, M. Liu, and L. Li, "An intelligence optimized rolling grey forecasting model fitting to small economic dataset," *Abstract and Applied Analysis*, vol. 2014, Article ID 641514, 10 pages, 2014.

[20] Y. Shi, "Brain Storm Optimization Algorithm," in *Proceedings of the 2nd International conference on Swarm Intelligence*, pp. 303–309, 2011.

[21] M. Cruz-Ramírez, C. Hervás-Martínez, J. Sánchez-Monedero, and P. A. Gutiérrez, "Metrics to guide a multi-objective evolutionary algorithm for ordinal classification," *Neurocomputing*, vol. 135, pp. 21–31, 2014.

[22] R. de Cássia dos Santos Navarro da Silva, V. Paula Rodrigues Minim, A. Navarro da Silva, and L. Antônio Minim, "Number of judges necessary for descriptive sensory tests," *Food Quality and Preference*, vol. 31, pp. 22–27, 2014.

[23] W. Zhang, J. Wu, J. Wang, W. Zhao, and L. Shen, "Performance analysis of four modified approaches for wind speed forecasting," *Applied Energy*, vol. 99, pp. 324–333, 2012.

[24] J. Wu, J. Wang, and D. Chi, "Wind energy potential assessment for the site of Inner Mongolia in China," *Renewable and Sustainable Energy Reviews*, vol. 21, pp. 215–228, 2013.