Complex System Analysis on Voter Stochastic System and Jump Time Effective Neural Network of Stock Market

Jun Wang, Huopo Pan
School of Science, Beijing Jiaotong University
Beijing 100044, P.R. CHINA

Yiduan Wang
School of Economics and Management, Beijing Jiaotong University
Beijing 100044, P.R. CHINA

Hongli Niu
School of Science, Beijing Jiaotong University
Beijing 100044, P.R. CHINA
E-mail: niuhonglibjtu@yeah.net

Received 8 January 2014
Accepted 25 May 2015

Abstract
The finite-range voter system, one of stochastic particle systems, is applied to model a financial price process for further description and investigation of fluctuations of Shanghai Composite Index. For different parameter values of the intensity $\lambda$ and the range $R$, we investigate the statistical behaviors of the simulation data for this financial model. Then we develop the random jump time effective neural network model to forecast the fluctuations of Shanghai Composite Index. Moreover, we compare the two models by analyzing the returns and the absolute returns of Shanghai Composite Index, the simulation data and the predictive data through Detrended Fluctuation Analysis and classic Rescaled Range Analysis.

Keywords: Prediction; financial time series; voter interacting system; random jump time effective neural network; detrended fluctuation analysis; rescaled range analysis

1. Introduction
As the stock markets are becoming deregulated worldwide, the modelling of the dynamics of the forwards prices is becoming a key problem in the risk management, physical assets valuation, and derivatives pricing, and it is also important to understand the statistical properties of fluctuations of stock prices in globalized securities markets. By applying the theory of stochastic systems, some research work has been done to investigate the statistical behaviors of fluctuations for stock markets, and the corresponding valuation and hedging of contingent claims for the price process models are also studied. In these financial models, the main assumption is that the stock price fluctuation is influenced by the information in a stock market, and the investors decide their investment opinions by other investors' attitudes, so the investors' investment attitudes to the stock market lead to the stock price fluctuation. In the present paper, we apply a stochastic interacting particle system—the finite-range voter system—to study the fluctuation behaviors of the return processes. Since interacting particle systems consist of a large number of interacting units, we think that economic systems such as financial markets are similar to interacting particle systems in that they are comprised of...
a large number of interacting ‘agents’. Through the computer simulation\textsuperscript{14-17}, we study the statistical behaviors of returns for this financial model.

Artificial neural networks (ANN) is one of technologies that have made great progress in studying stock markets\textsuperscript{18-25}. ANN has good self-learning ability, strong anti-jamming capability, has been widely used in the financial fields such as stock prices, profits, exchange rate, risk analysis and prediction. In a financial market, although the historical data has a great influence on the investors’ positions, we think that the impacts of different historical data on the stock price are not the same. In the present paper, we suppose that the degree of impact of the data depends on its occurrence date (or time), we give the data a high level effect when it is very near to the current state. Furthermore, we also introduce the Brownian motion and Poisson jump in the financial model\textsuperscript{26-27}, in order to make the model have the effect of random movement and random jump while maintaining the original trend. In the real stock market, drastic fluctuations (or jumps) in financial assets play a crucial role in volatility forecasting. And jumps have positive and significant impact on future volatilities. In this work, the artificial neural network model based on random jump time effective function is applied to forecast Shanghai Composite Index (SHCI).

It is widely acknowledged that financial market is a complex dynamic system, which are characterized by various interesting statistical fluctuation properties. The simulation and forecasting of the financial stock indices have received considerable attentions from the financial investors. The main purpose in this paper is to perform the complex statistical analysis of the simulated and forecasted data of real market index, with the hoping to the complex statistical analysis of the simulated and forecasted data of real market index, with the hoping to investors’ positions, we think that the impacts of different historical data on the stock price are not the same. In the present paper, we suppose that the degree of impact of the data depends on its occurrence date (or time), we give the data a high level effect when it is very near to the current state. Furthermore, we also introduce the Brownian motion and Poisson jump in the financial model\textsuperscript{26-27}, in order to make the model have the effect of random movement and random jump while maintaining the original trend. In the real stock market, drastic fluctuations (or jumps) in financial assets play a crucial role in volatility forecasting. And jumps have positive and significant impact on future volatilities. In this work, the artificial neural network model based on random jump time effective function is applied to forecast Shanghai Composite Index (SHCI).

In this section, we introduce two important modelling processes for stock index. One is the financial stock price model by voter interacting system, according to which we can obtain various simulative stock price time series. Another is the random jump time effective neural network model in which we introduce a random jump time effective function to the BP algorithm. From this model, we can achieve the forecasting of stock index.

2. Experimental Modelling

First we give the brief definitions and properties of voter interacting system. Voter interacting system is one of stochastic particle systems, we think of the sites of the $d$-dimensional integer lattice as being occupied by persons who either in favor of or opposed to some issue. Let $\eta^{s}_u$ denote the state at time $s$ with the initial state $\eta^{0}_u = A$ for the voter system\textsuperscript{9,12,13}. For more generally, we can consider the initial distribution as $\nu_0$, the product measure with density $\theta$, that is, each site is independently occupied with probability $\theta$ and let $\eta^{s}_u$ denote the voter system with initial distribution $\nu_0$. Let $\eta^{s}_u = \{0\}$, and $\eta^{s}_u(u)$ be the state of $u \in \mathbb{Z}^d$ at time $s$ with the initial point $\{0\}$. More formally, the stochastic dynamics of the voter system $\eta$ is a Markov process on $\Omega = \{0,1\}^\mathbb{Z}$ whose generator has the form

\begin{equation}
Af(\eta) = \sum_u c(u,\eta)[f(\eta^u) - f(\eta)]
\end{equation}

where the functions $f$ on $\Omega$ depend on finitely many coordinates, and $\eta^u(v) = \xi(v)$ if $u \neq v$, $\eta^u(v) = 1 - \eta(v)$.
if $u=v$, for $u,v \in \mathbb{Z}^d$. $c(u,\eta)$ is the transition rate function for the process which is given by follows:

$$
c(u,\eta) = \begin{cases} 
\lambda \sum_{v \in \eta(u) \cap R} p(u,v)\eta(v) & \text{if } \eta(u) = 0 \\
\sum_{v \in \eta(u) \cap R} p(u,v)(1-\eta(v)) & \text{if } \eta(u) = 1 
\end{cases}
$$

where $R$ is the range of the model, sites in $\{v:|u-v|\leq R\}$ are the neighbors of $u$, and $\lambda$ is the intensity parameter of the model. When $\lambda = 1$, the model is called the voter system, and when $\lambda > 1$, the model is called the biased voter system. In the above definition, let $p(u,v) \geq 0$ for $u,v \in \mathbb{Z}^d$, and $\sum_{u \in \mathbb{Z}^d} p(u,v) = 1$ for all $u \in \mathbb{Z}^d$, and we assume that $p(u,v)$ is such that the Markov chain with those transition probabilities is irreducible.

For the biased voter model ($\lambda > 1$), there is a “critical value” for the process, that is, on $\Omega = \{0,1\}^\mathbb{Z}^d$ and with the corresponding probability $P$, the critical value $\lambda_c$ is defined as

$$
\lambda_c = \inf \left\{ \lambda : P\left[|\eta_s| > 0, \text{for all } s \geq 0 \right] > 0 \right\}
$$

where $|\eta_s|$ is the cardinality of $\eta_s$. On $d$-dimensional lattice, if $\lambda < \lambda_c$, the process dies out (becomes vacant) exponentially fast, if $\lambda > \lambda_c$, the process survives with the positive probability.

Now we introduce the graphical representation of the model, since the graphical representation is necessary for computer simulations. For simplicity, we give the construction of graphical representation for 1-dimensional voter system with $\lambda = 1$ and $R = 3$ for more general cases, see Refs. 9, 12 and 13. Thinking of 1-dimensional integer points as being laid out on a horizontal axis, with the time lines being placed vertically, above that axis. Define independent Poisson processes with rate for each time lines, at each event time $(u,s)$, we choose one of its six neighbors with probability 1/6, and draw an arrow from that neighbor to $(u,s)$, and write a $\delta$ at $(u,s)$, see Fig. 1. To construct the process from this “graphical representation”, we imagine fluid entering the bottom at the points in $\eta_s$ and flowing up the structure. The $\delta$’s are the dams and the arrows are pipes which allow the fluid to flow in the indicated direction.

In the following, based on the voter system on $d$-dimensional integer lattice, we model a return process for a stock market. We assume that market information leads to the fluctuation of a stock price, and there are three kinds of information: buying information, selling information and neutral information. The fluctuation of a stock price relies on the investor’s investment attitudes, which accordingly classify buying stock, selling stock and holding stock. Consider a model of auctions for the same stock defined above, we can derive the stock price process from the auctions. Suppose that each trader can trade the stock several times at each day $t \in \{1,2,\ldots,n\}$, but at most one unit number of the stock at each time. Let $\beta$ be the time length of trading time in each trading day, we denote the stock price at time $s$ in the $t$-th trading day by $S_t(s)$, where $s \in \{0,1\}$. Suppose that this stock consists of $2m_2 + 1$ ($m_2$ is large enough) investors, who are located in a line $\{-m_2,\ldots,-1,0,1,\ldots,m_2\} \subset \mathbb{Z}$ (similarly for $d$-dimensional lattice $\mathbb{Z}^d$). At the beginning of trading in each day, suppose that only the investor at the site $\{0\}$ receives some news. We define a random variable $\xi_t(\{0\})$ for this investor, suppose that this investor taking buying position ($\xi_t(\{0\}) = 1$), selling position ($\xi_t(\{0\}) = -1$) or neutral position ($\xi_t(\{0\}) = 0$) with probability $p_1$, $p_2$, or $1 - p_1 - p_2$ respectively. Then this investor sends bullish, bearish or neutral signal to his nearest neighbors. According to $d$-dimensional voter process system, investors can affect each other or the news can be spread, which is assumed as the main factor of price fluctuations for the investors. Moreover, here the investors can change their buying positions or selling positions to neutral positions independently at a constant rate. More specifically: (a) When $\xi_t(\{0\}) = 1$ and if $\eta_s^{(u)}(u) = 1$, we say that the investor at $u$ takes buying position at time $s$, and this investor recovers to neutral position 0 at a rate equal to the number of the vacant neighbors; if $\eta_s^{(u)}(u) = 0$, we...
think the investor at \( u \) takes neutral position at time \( s \), and this investor is changed to take buying position by his neighbors at rate \( \lambda \sum_{v \in \text{vacant}} \eta_v^\text{BM}(v) \). In this case, the more investors with taking buying positions, the more possible the stock price goes up. (b) When \( \xi_v(0)=1 \) and \( \eta_v^\text{BM}(u)=-1 \), we say that the investor at \( u \) takes selling position at time \( s \), also this investor recovers to neutral position \( 0 \) at a rate equal to the number of the vacant neighbors; if \( \eta_v^\text{BM}(u)=0 \), the investor is changed to take selling position by his neighbors at rate \( \lambda \sum_{v \in \text{vacant}} \eta_v^\text{BM}(v) \). In this case, the more investors with taking selling positions, the more possible the stock price goes down. (c) When the initial random variable \( \xi_v(0)=0 \), the process \( \eta_v^\text{BM}(u) \) is ignored, this means that the investors don’t affect the fluctuation of the stock price. For \( l \) large enough, the normalized trend function is given by

\[
C_i(s) = \xi_i(0) \left[ \frac{\eta_i^\text{BM}}{2m_s} \right], \quad s \in [0,l]
\]  

where \( \eta_i^\text{BM} = \sum_{v \in \text{vacant}} \eta_v^\text{BM}(v) \). From the above definitions and Refs. 28-31, we define the stock price at \( t \)-th trading day \( t \in [1,2,\ldots,n] \) as

\[
S_i(s) = e^{\alpha \cdot C_i(s)}, \quad s \in [0,l]
\]  

where \( \alpha > 0 \) represents the depth parameter of the market. Then we have

\[
S_i(s) = S_0 \exp\left(\alpha \sum_{k=1}^{n} C_i(s)\right), \quad s \in [0,l]
\]  

where \( S_0 \) is the stock price at time 0.

2.2. Random jump time effective neural network model

First we introduce the three-layer BP neural network model in Fig. 2, for the details see Refs. 17 and 19. For any fixed neuron \( n \ (n=1,2,\ldots,N) \), the model has the following structure: let \( \{x_i(n) : i=1,2,\ldots,p\} \) denote the set of input of neurons, \( \{y_j(n) : j=1,2,\ldots,m\} \) denote the set of output of hidden layer neurons; \( \theta_{ij} \) is weight that connects the node \( i \) in the input layer neurons to the node \( j \) in the hidden layer; \( W_{jk} \) is weight that connects the node \( j \) in the hidden layer neurons to the node \( k \) in the output layer; and \( \{o_k(n) : k=1,2,\ldots,q\} \) denote the set of output of neurons. Then the output value for a unit is given by the following function

\[
y_j(n) = f\left(\sum_{i=1}^{p} W_{ij} x_i(n) - \theta_{ij}\right), \quad o_k(n) = f\left(\sum_{j=1}^{m} W_{jk} y_j(n) - \theta_{jk}\right)
\]  

where \( \theta_{ij}, \theta_{jk} \) are the neutral thresholds, \( f(x) = 1/(1 + e^{-x}) \) is Sigmoid activation function. Let \( T_i(n) \) be the actual value of data sets, then the error of the corresponding neuron \( k \) to the output is defined as \( e_k = T_k - o_k \).

Fig. 2. The plot of three-layer neural network.

In stock markets, the empirical research shows that the fluctuations of price changes are believed to follow a Gaussian distribution for long time intervals but to deviate from it for short steps, and the actual price changes usually exhibit excess kurtosis and fatter tails than the normal distribution, which is called the “fat-tail” phenomenon. This phenomenon is usually caused by the drastic fluctuations of stock prices. In view of the above reality status, the error of the output is defined as follows. Let \( e = e_k^2 / 2 \), then the error of the sample \( n \ (n=1,2,\ldots,N) \) is defined as

\[
e(n,t) = \frac{1}{2} \phi(t) \sum_{k=1}^{q} \left(T_k(n) - o_k(n)\right)^2
\]  

where \( \phi(t) \) is the random jump time effective function. Now we defined \( \phi(t) \) as following

\[
\phi(t) = \frac{1}{t} \exp\left(-\int_0^t \mu(s) ds - \frac{1}{2} \int_0^t \sigma(s)^2 ds \right) \prod_{t < \tau < t_i} e^\tau
\]  

where \( \tau (\tau > 0) \) is the time strength coefficient, \( t_i \) is the current time or the time of newest data in data set, \( t_i \) is an arbitrary time point in data set. \( J \ (l=1,2,\ldots,N(t)) \) are independent and identically distributed jump processes and \( \mu(t) \) obey the normal distribution with mean \( \mu_j \) and variance \( \sigma_j \). \( N(t) \ (t > 0) \) is a stochastic Poisson process with the intensity \( \gamma \). \( \mu(t) \) is the drift function (or the trend term), and \( \sigma(t) \) is the volatility function, \( B(t) \) is the standard Brownian motion, see Refs. 29 and 31. The random jump time effective function implies that: (a)
The recent information has a stronger effect on the investors than the old information. In details, the nearer the events happen, the greater the investors and market are affected. (b) The information (or data) at the jump time may have a stronger (or weaker) effect on the investors. Then the total error of all data training set in the set output layer with the random jump time effective function are defined as

\[ E = \frac{1}{N} \sum_{n=1}^{N} E(n, t) = \frac{1}{N} \sum_{n=1}^{N} \frac{1}{\tau} \left[ \sum_{i} e_{i}^{(t)} \sum_{j} \left( T_{ij} (n) - \delta_{ij} (n) \right)^{2} \right] \]

(10)

The training algorithms procedures of the neural network is described as follows:
Step 1: Normalizing the data as follows:
Step 2: At the beginning of data processing, connective weights, and calculate the actual value of the neuron.
Step 3: Introducing the jump stochastic time effective function in the error function.
Step 4: Establishing an error acceptable model and transfer function from hidden layer to output layer.
Step 5: Modifying connective weights: calculate pre-set minimum error, go to Step 6, otherwise go to where 

\[ k = \frac{1}{N} \sum_{n=1}^{N} \left[ \delta_{ij} (n) \right] \]

and calculate \( \delta \) backward for the node in hidden layer:

\[ \delta_{ij} (n) = \frac{1}{\tau} \left[ \sum_{i} e_{i}^{(t)} \sum_{j} \left( o_{ij} (n) - o_{ij} (n) \right) \right] \sum_{i} W_{ij} (n) \]  

(12)

where \( o(n) \) is the output of the neuron \( n \), \( T(n) \) is the actual value of the neuron \( n \) in data sets, \( o(n)[1-o(n)] \) is the derivative of the sigmoid activation function and \( h' \) is each of the node which connect with the node \( h \) and in the next hidden layer after node \( h \). Modifying the weights from the layer to the previous layer:

\[ W_{j} (n+1) = W_{j} (n) + \eta \delta_{j} (n) y(n) \quad \text{or} \]
\[ V_{j} (n+1) = V_{j} (n) + \eta \delta_{j} (n) x(n) \]

(13)

where \( \eta \) is learning step, which usually takes constant between 0 and 1.

Step 6: Outputting the predictive value.

3. Statistical methodology

3.1. Detrended fluctuation analysis

Detrended fluctuation analysis (DFA) is a scaling analysis method providing the scaling exponent \( \alpha \) to represent the correlation properties. Briefly, for a given stochastic time series \( S(i), i = 1, 2, \ldots, N \) with the sampling period \( \Delta t \), the DFA method can be implemented as follows:
Step 1: Compute the mean \( \bar{S} = \frac{1}{N} \sum_{i} S(i) \), and obtain an integrated time series \( y(j) = 1/\Delta t \sum_{i=1}^{j} (S(i) - \bar{S}) \). Then divide the integrated time series into boxes of equal size, \( n \).
Step 2: In each box, fit the integrated time series by using a polynomial function, \( y_{\mu}(j) \). For order-\( l \) DFA, \( l \) order polynomial function should be applied to the fitting, and we choose \( l = 2 \) in this paper. Then calculate the detrended fluctuation function as follows:

\[ y(i) = y(i) - y_{\mu}(i). \]

(14)

Step 3: For a given box size \( n \), calculate the root mean square fluctuation:

\[ F(n) = \left( \frac{1}{N} \sum_{j} [y(j)]^{2} \right)^{1/2}. \]

(15)

A power-law relation between \( F(n) \) and the box size \( n \) indicates the presence of scaling: \( F(n) \propto n^{\alpha} \) (where \( F(n) \) is calculated by \( \log F(n) = \beta + \alpha \log n \)). The parameter \( \alpha \), called the scaling exponent or correlation exponent, represents the correlation properties of the time series: if \( \alpha = 0.5 \), there is no correlation and the time series is uncorrelated; if \( 0 < \alpha < 0.5 \), it indicates the power-law anti-correlations; if \( 0.5 < \alpha < 1 \), the time series has the persistent long-range power-law correlations; if \( 1 < \alpha \), it indicates that the correlations exist but not in a power-law form.

3.2. Rescaled Range Analysis

The evaluation of the long memory of one time series can be made through various methodologies. Hurst exponent \( H \) is an important measure of the long memory, which can be calculated through the classic Rescaled Range (R/S) Analysis. The research shows...
that, when $H = 0.5$, the time series belongs to Brownian motion and the variables are independent and the correlation coefficients are 0; when $0 < H < 0.5$, the variables are negative correlation and are anti-persistence; when $0.5 < H < 1$, that means the time sequence is persistent and has the quantification of long-term memory. In order to calculate the Hurst exponent $H$, the time series $x_t$ with $N$ numbers is divided by $A$ parts. And each part marks as $x_{a,i}$, $a = 1, 2, \ldots, A$. For $t = 1, 2, \ldots, n$, let

$$x_a = \frac{1}{N} \sum_{j=1}^{N} x_{a,j}, \quad X_a = \sum_{j=1}^{N} (x_{a,j} - \bar{x}_a)$$

$$R_a = \max_{i \leq t \leq i + A} X_{a,i} - \min_{i \leq t \leq i + A} X_{a,i}, S_a = \left[ \frac{1}{A} \sum_{j=1}^{A} (x_{a,j} - \bar{x}_a)^2 \right]^{1/2}$$

By computing $R_a / S_a$ by each part data, then we have the mean as follows

$$\langle R / S_a \rangle = \frac{1}{A} \sum_{a=1}^{A} (R_a / S_a).$$

The Hurst exponent $H$ is obtained by the expression

$$\ln \langle R / S_a \rangle = \ln C + H \ln n,$$

which is given by Mandelbrot & Wallis, and the values of $H$ can be calculated by the least ordinary squares.

4. Results and discussions

In this section, we study the statistical properties of returns and absolute returns of the simulated data from the financial stock model by voter interacting system, the predictive data from the random jump time effective neural network model and the actual SHCI data.

For the simulated prices, in Fig. 3, we do the computer simulation of the financial model for different values of the intensity $\lambda (\lambda = 5, 10, 15)$ and the finite-range $R (R = 1, 2, 3)$. We let the initial price $S_0$ be 5320, the price of the index SHCI on January 2, 2008. For the predictive price, in the random jump time effective neural network model, we input five kinds of stock prices of SHCI: daily open price, daily closed price, daily input layer is 5, the number of neural nodes in the hidden layer is 13 and the number of neural nodes in output layer is 1. In this paper, we take $\mu_j$ and $\sigma_j$ to be the mean and the highest price, daily lowest price and daily trade volume; and one price of stock prices in the output layer: the closed price of the next trade day. The number of neural nodes in variance of reality historical data of SHCI, and let the intensity $\gamma$ be 1/30. That is to say, jump will happen 10 times a year in average. Moreover, we suppose that the values of vector $(\mu(t), \sigma(t))$ are (1, 1). We divide data into two sections: we use the data from 2003 to 2007 for training and forecast the SHCI from 2008 to 2009 for each trading day. In the experiment, we perform 15 times of the random jump time effective neural network model and report the averaged predictive data as the final predictive result. The empirical results show that the model exhibits stable training and predicting ability despite the random fluctuations from the improved algorithm. Fig. 4 presents the predictive data of the model and the actual data.

![Fig. 3(a)](image1)

![Fig. 3(b)](image2)

![Fig. 3(c)](image3)

Fig. 3. Plots of simulation data of the financial model by finite-range voter system.

![Fig. 4](image4)

Fig. 4. SHCI forecasted by random jump time effective neural network.

We denote the daily price at time $t$ by $S(t)$ ($t = 1, 2, \ldots$), then we have the formula of stock logarithmic return:

$$R(t) = \ln S(t + 1) - \ln S(t).$$

The corresponding absolute return is defined as
\[ R(t) = \ln S(t+1) - \ln S(t). \]  

(19)

In the following, we begin to investigate and compare the returns and the absolute returns of the simulated data, the predictive data and the actual data by using the detrended fluctuation analysis (DFA) approach and the rescaled range (R/S) statistic. Table 1 and Table 2 show the basic statistical properties of the returns and the absolute returns.

According to the DFA method, we analyze the statistical behaviors of returns of the simulated data, the actual data and the predictive data, see Fig. 5 and Table 3 for empirical results. Specially, the small figures in the upper left and lower right corner of Fig. 5 represent the scaling exponent \( \alpha \) value of the actual data and its closest situation. The empirical research shows that the smallest exponent corresponds to the simulated data when \( \{ R = 1, \lambda = 5 \} \), and the largest one is for the predictive index. In Table 3, the results reveals that five \( \alpha \) values are bigger than 0.5, including the actual data, which indicates that these five corresponding time series are long-range auto-correlated according to the concept of \( \alpha \). And when \( \{ R = 2, \lambda = 15 \} \), the result of simulation by the finite-range financial model is the closest to the actual situation.

By applying the method of R/S test, the exponent \( H \) of the returns for the simulated data, the actual data and the predictive data are investigated in Fig. 6 and Table 4. The \( H \) of three groups’ data are all larger than 0.5, which indicate that these three group financial time series are all long-term memory sequences. In addition,

---

### Table 1. Returns statistics of the simulated data, the predictive data and the actual data.

<table>
<thead>
<tr>
<th>( R )</th>
<th>( \lambda )</th>
<th>Mean</th>
<th>Variance</th>
<th>Skewness</th>
<th>Kurtosis</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>-8.8809e-5</td>
<td>1.9539e-6</td>
<td>1.8258e-2</td>
<td>7.5903e-1</td>
<td>-4.0000e-3</td>
<td>5.0000e-3</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>1.1550e-4</td>
<td>5.6737e-6</td>
<td>1.0601e-2</td>
<td>8.8434e-1</td>
<td>-8.2500e-3</td>
<td>8.0000e-3</td>
</tr>
<tr>
<td>1</td>
<td>15</td>
<td>8.3162e-4</td>
<td>9.8055e-6</td>
<td>-1.1185e-1</td>
<td>8.0303e-1</td>
<td>-1.0000e-2</td>
<td>9.7500e-2</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>-3.4907e-4</td>
<td>1.9365e-5</td>
<td>4.9578e-2</td>
<td>3.7003e-1</td>
<td>-1.2500e-2</td>
<td>1.2750e-2</td>
</tr>
<tr>
<td>2</td>
<td>15</td>
<td>-1.3280e-3</td>
<td>1.1827e-4</td>
<td>1.4396e-2</td>
<td>-6.0905e-1</td>
<td>-2.5000e-2</td>
<td>2.5000e-2</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>3.7833e-4</td>
<td>7.7932e-5</td>
<td>6.3248e-2</td>
<td>-1.4102e-1</td>
<td>-2.3750e-2</td>
<td>2.3750e-2</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>-3.7422e-4</td>
<td>1.9203e-4</td>
<td>-3.9950e-2</td>
<td>7.5850e-1</td>
<td>-2.5000e-2</td>
<td>2.5000e-2</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td>-6.1704e-4</td>
<td>7.6169e-4</td>
<td>1.4385e-1</td>
<td>-5.8214e-1</td>
<td>-5.0000e-2</td>
<td>5.0000e-2</td>
</tr>
</tbody>
</table>

Table 2. Absolute returns statistics of the simulated data, the predictive data and the actual data.

<table>
<thead>
<tr>
<th>( R )</th>
<th>( \lambda )</th>
<th>Mean</th>
<th>Variance</th>
<th>Skewness</th>
<th>Kurtosis</th>
<th>Minimum</th>
<th>Maximum</th>
</tr>
</thead>
<tbody>
<tr>
<td>predictive</td>
<td>5</td>
<td>1.0539e-3</td>
<td>8.4881e-7</td>
<td>1.2842</td>
<td>1.4928</td>
<td>0.0000</td>
<td>5.0000e-3</td>
</tr>
<tr>
<td>predictive</td>
<td>10</td>
<td>1.8085e-3</td>
<td>2.4096e-6</td>
<td>1.3946</td>
<td>2.0432</td>
<td>0.0000</td>
<td>8.2500e-3</td>
</tr>
<tr>
<td>predictive</td>
<td>15</td>
<td>2.3963e-3</td>
<td>4.0584e-6</td>
<td>1.3883</td>
<td>1.9593</td>
<td>0.0000</td>
<td>9.7500e-2</td>
</tr>
<tr>
<td>actual</td>
<td>5</td>
<td>3.4671e-3</td>
<td>7.4413e-6</td>
<td>1.2401</td>
<td>1.2645</td>
<td>0.0000</td>
<td>1.2750e-2</td>
</tr>
<tr>
<td>actual</td>
<td>10</td>
<td>6.2402e-3</td>
<td>2.3165e-5</td>
<td>1.0006</td>
<td>6.2121e-1</td>
<td>0.0000</td>
<td>2.4500e-2</td>
</tr>
<tr>
<td>actual</td>
<td>15</td>
<td>9.0210e-3</td>
<td>3.8492e-5</td>
<td>6.2217e-1</td>
<td>-5.0543e-1</td>
<td>0.0000</td>
<td>2.5000e-2</td>
</tr>
<tr>
<td>predictive</td>
<td>5</td>
<td>1.4110e-3</td>
<td>8.2033e-4</td>
<td>4.5136e-1</td>
<td>2.1941</td>
<td>-8.5685e-1</td>
<td>1.4191e-1</td>
</tr>
<tr>
<td>predictive</td>
<td>10</td>
<td>1.2906e-3</td>
<td>6.4592e-4</td>
<td>-4.1636e-1</td>
<td>3.1190</td>
<td>-1.0063e-1</td>
<td>1.0063e-1</td>
</tr>
<tr>
<td>predictive</td>
<td>15</td>
<td>-1.4110e-3</td>
<td>8.2033e-4</td>
<td>4.5136e-1</td>
<td>2.1941</td>
<td>-8.5685e-1</td>
<td>1.4191e-1</td>
</tr>
<tr>
<td>actual</td>
<td>5</td>
<td>-1.4110e-3</td>
<td>8.2033e-4</td>
<td>4.5136e-1</td>
<td>2.1941</td>
<td>-8.5685e-1</td>
<td>1.4191e-1</td>
</tr>
<tr>
<td>actual</td>
<td>10</td>
<td>-1.2906e-3</td>
<td>6.4592e-4</td>
<td>-4.1636e-1</td>
<td>3.1190</td>
<td>-1.0063e-1</td>
<td>1.0063e-1</td>
</tr>
</tbody>
</table>

---

![Fig. 5. The plot of DFA test statistics of returns for the simulation data, the actual data and the predictive data](image-url)
the time series which is closest to the actual situation is the simulated series by finite-range financial model when \( R = 1 \) and \( \lambda = 10 \). Furthermore, Fig. 7 and Table 4 show the exponent \( H \) of the absolute returns for the simulated data, the actual data and the predictive data. All the values are bigger than 0.5, which means all the time sequences are persistent and have the quantification of long-term memory.

<table>
<thead>
<tr>
<th>( R )</th>
<th>( \lambda )</th>
<th>( \alpha )</th>
<th>( \beta )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>0.4150</td>
<td>-3.5411</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>0.4533</td>
<td>-3.3302</td>
</tr>
<tr>
<td>1</td>
<td>15</td>
<td>0.4700</td>
<td>-3.2773</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>0.4603</td>
<td>-3.1042</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>0.4555</td>
<td>-2.8738</td>
</tr>
<tr>
<td>2</td>
<td>15</td>
<td>0.6009</td>
<td>-2.9258</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>0.5504</td>
<td>-2.9162</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>0.5178</td>
<td>-2.7212</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td>0.4477</td>
<td>-2.3100</td>
</tr>
</tbody>
</table>

actual data 0.6346 -2.5634
predictive data 0.6876 -2.4736

Table 4. Hurst exponents of the simulated data, the predictive data and the actual data.

<table>
<thead>
<tr>
<th>( R )</th>
<th>( \lambda )</th>
<th>returns</th>
<th>absolute returns</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>0.6605</td>
<td>0.6072</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>0.6793</td>
<td>0.7286</td>
</tr>
<tr>
<td>1</td>
<td>15</td>
<td>0.6604</td>
<td>0.6327</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>0.6357</td>
<td>0.6529</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>0.6581</td>
<td>0.6551</td>
</tr>
<tr>
<td>2</td>
<td>15</td>
<td>0.6293</td>
<td>0.6785</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>0.7203</td>
<td>0.7294</td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>0.6521</td>
<td>0.6468</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td>0.6331</td>
<td>0.6481</td>
</tr>
</tbody>
</table>

5. Conclusion

In the present paper, we model a stock price process by applying the finite-range voter system to study the statistical behaviors of fluctuations for the index SHCI. Further we develop the artificial neural network model based on random jump time effective function to forecast the index SHCI. And we compare the statistical properties of the returns and the absolute returns for the simulated data, the actual data and the predictive data by using DFA and classic R/S analysis. We see that scaling exponents of DFA method for these three groups’ data are all larger than 0.5, indicating the long-range correlation, and when \( \{ R = 2, \lambda = 15 \} \), the result of simulated data by the price model is the closest to the actual situation. Similarly, by R/S analysis, when \( \{ R = 1, \lambda = 10 \} \), the simulated time series is the closest to the actual index. The research results may imply that, from a micro perspective, the ANN model is closer to the actual situation, yet the financial model derived from the voter interacting system may has better performance from the macro point of view.

Acknowledgements

The authors were supported in part by National Natural Science Foundation of China Grant No. 71271026 and Grant No. 10971010.

References


