Applied Soft Computing 62 (2018) 923–932

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Wavelet Neural Network Prediction Method of Stock Price Trend Based on Rough Set Attribute Reduction

Lei Lei

School of Business Administration, Henan University of Economics and Law, Zhengzhou, Henan 450046, China

ARTICLE INFO

Article history:
Received 9 December 2016
Received in revised form
16 September 2017
Accepted 17 September 2017
Available online 29 September 2017

Keywords:
Wavelet Neural Network
Rough Set
Attribute reduction
Stock price
Prediction

ABSTRACT

To improve the prediction capacity of stock price trend, an integrated prediction method is proposed based on Rough Set (RS) and Wavelet Neural Network (WNN). RS is firstly introduced to reduce the feature dimensions of stock price trend. On this basis, RS is used again to determine the structure of WNN, and to obtain the prediction model of stock price trend. Finally, the model is applied to prediction of stock price trend. The simulation results indicate that, through RS attribute reduction, the structure of WNN prediction model can be simplified significantly with the improvement of model performance. The directional symmetry values of prediction, corresponding to SSE Composite Index, CSI 300 Index, All Ordinaries Index, Nikkei 225 Index and Dow Jones Index, are 65.75%, 66.37%, 65.97%, 65.52% and 66.75%, respectively. The prediction results are better than those obtained by other neural networks, SVM, WNN and RS-WNN, which verifies the feasibility and effectiveness of the proposed method of predicting stock price trend.

1. Introduction

The purpose of stock price prediction is to explore the development law of stock market so as to provide a scientific basis for stock investments. As the stock price volatility is caused by many factors, it is difficult to grasp the uncertainty of these factors affecting stock prices. Therefore, accurate prediction of stock prices is a difficult task in the finance field [1,2].

In the existing stock price prediction methods, time series [3], gray [4], prosperity [5] and other methods are usually used. Since White first used the neural network to predict the daily return rate of IBM ordinary stock [6], the use of neural network in stock price prediction has become a hot research theme [7–15]. These studies make full use of the advantages of neural network such as self-organizing, self-learning, self-adapting, distributed processing, and can overcome the shortcomings of conventional prediction methods. However, for neural networks, there are still problems such as low precision, slow convergence and easy inclination to local minimum. Although some improvements have been made, it cannot solve the problem fundamentally. This affects wider application and popularization of this method. Wavelet Neural Network (WNN) is a new neural network developed on the basis of wavelet analysis. Compared with conventional neural networks, WNN has strict theory foundations and good mapping capabilities, and is suitable for various application areas of neural network [16]. Nevertheless, there are still some problems for WNN to predict stock prices.

In this paper, our research will focus on reducing input dimensions and optimizing the structure for a better WNN prediction model.

Based on the above analysis, a WNN method of predicting stock price trend is proposed based on Rough Set (RS), which can reduce the input dimensions and optimize the structure for WNN. The feasibility of the method is verified by experiments.

The rest of the paper is organized as follows. Section 2 gives the problem statement. RS attribute reduction is introduced in Section 3. Section 4 describes an adaptive WNN learning algorithm. The integration of WNN and RS is discussed in Section 5. The experiment on stock price prediction and result analysis are presented in Section 6. Section 7 has some discussions on incremental learning. The last section gives concluding remarks.

2. Problem statement

The stock prediction is that, based on accurate survey statistics and stock market information, scientific methods are used to predict the future tendency of stock market from the history, current situation and stock market laws. There are many factors that affect the stock market development. Uncertain interactions between these influence factors are very complex, which will lead to pre-

E-mail address: gxsh15934896556@qq.com

https://doi.org/10.1016/j.asoc.2017.09.029
1568-4946/

diction deviations. With development of the national economy and in-depth understanding of stock market, it becomes increasingly important for stock investment and risk management to use a reasonable and effective method to accurately predict the stock price trend. At present, the prediction of stock price trend mainly has the following problems.

2.1. Stock price trend has mutability and variability.

Stock price trend often show mutability and variability because of some risks. The risks can be divided into systemic risks and non-systemic risks qualitatively, which can cause the mutation of stock price trend or index. Sometimes, the factors affecting the mutation of stock price trend cannot be observed, such as human manipulation; sometimes, although they can be observed, they are actually false information, such as fraud and rumors. These factors may cause abnormal fluctuations and structure mutations of stock market, manifesting as the mutability and variability of stock price trend. This makes the stock price trend data show a non-normal distribution, resulting in low accuracy for stock price trend prediction.

2.2. There is a certain amount of redundant information in stock price trend data

The stock price trend data, obtained by trading after multi-party game, often contain a lot of valuable information such as investment intentions and behaviors. At the same time, with larger stock trading data volume, there are usually more irrelevant indices, making the data contain a certain amount of redundant information. This affects effective prediction of stock price trend.

The above two problems may lead to low prediction accuracy for stock price trend. Both the conventional prediction method and the single intelligent prediction method cannot solve these problems effectively. In this context, the hybrid prediction method is preferred. Therefore, a hybrid method based on WNN and RS is proposed to predict the stock price trend. RS plays two roles in this method: (1) Reduce the input dimensions of WNN to remove redundant information in the sampled data; (2) Reduce redundant nodes in WNN hidden layers to determine a relatively optimal WNN structure. The proposed method is to enhance the prediction accuracy for stock price trend, providing references for stock investments and scientific researches.

3. RS attribute reduction

In RS theory, knowledge can be regarded as the division of universe of discourse. The roughness of knowledge can be defined from the view point of indiscernibility relation, so as to achieve rigorous analysis and processing of knowledge based on classification mechanism. As the basis of knowledge discovery, attribute reduction can reduce the overall number of attributes under the same classification ability, and has become the core content of RS theory [17,18].

Minimum attribute reduction is to obtain the reduction containing the least attributes under the same classification ability. It has been proved that searching all reduction and the minimum reduction is a NP hard problem, so some heuristic attribute reduction algorithms were proposed [19–21]. Most of them achieve the reduction by constructing the relationship between knowledge and information entropy. In these algorithms, the compatible classes obtained by compatibility relation constitute the coverage, rather than the division to universe of discourse. That is, the knowledge is regarded as the coverage of universe of discourse, which is clearly a lack of rationality. In this paper, based on the concept of hierarchical structure and approximate precision, a heuristic attribute reduction algorithm is introduced. By adding the heuristic information different from the information entropy, the search space of the problem can be reduced to obtain the optimal and near-optimal solution [22,23].

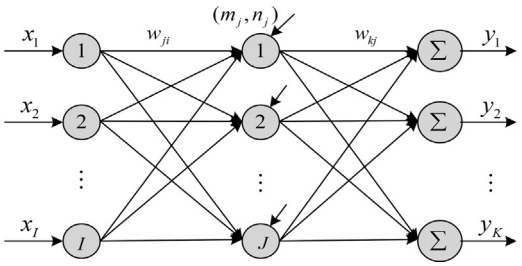

Fig. 1. WNN based on orthogonal wavelet.

Consider a decision information system $S=(U, R, V, f)$, where $U=\{x_1, x_2, \ldots, x_n\}$ denotes the object set (universe of discourse); $R=C\cup D=\{r_1, r_2, \ldots, r_m\}$ denotes the attribute set; C and D are the subset of condition attributes and the subset of decision attributes, respectively; $V=\bigcup_{r\in R} V_r$ (V_r is the value range of attribute $r\in R$); f is the information function between U and R ($f: U\times R\to V, f(x,r)\in V_r$). Suppose that X is a subset of U($X\subset U$). For each X and the indiscernibility relation B($B\subseteq A$), the lower approximation set $B_*(X)$, the set composed of the objects in U that must belong to X, is determined by the knowledge B. That is, $B_(X)=\{X\in U: B(X)\subseteq X\}$. The upper approximation set $B^*(X)$, the set composed of the objects in U that may belong to X, is determined by the knowledge B. That is, $B^*(X)=\{X\in U: B(X)\cdot\cap X\neq\emptyset\}$.

For the condition attributes, if the decision value of the objects in the indiscernibility relation set is the same, the decision system is compatible; otherwise, it is incompatible. In the attribute reduction algorithm, the equivalence relation of all condition attributes firstly need to be determined in compatible decision information system. Then, the upper and lower approximation sets with respect to the decision value are calculated for all equivalence relations. The heuristic parameter can be obtained by

$$\lambda=(\left|B^*(X)\right|-\left|B_*(X)\right|)/\left|U\right| \tag{1}$$

The attribute with the smallest λ is selected. For a given universe of discourse, the lower approximate value can be used as the basis for pruning

$$U=U-POS_{red}\{d\} \tag{2}$$

where $POS_{red}\{d\}$ is the object set of decision attribute d whose condition attributes of reduction set *red* can be classified into U. Repeat the process and select the attribute with the smallest λ. In each time, the remaining attributes are combined with the attributes in the reduction set *red* to form a new equivalence relation. The algorithm is recursive and the calculation is terminated until the given universe of discourse is empty [24].

For a decision information system $DS=(U, C\cup D, V, f)$, the recursive algorithm is as follows:

1. The reduction set *red* can be initialized as $\emptyset$ (empty set), and C is all condition attributes.
2. λ of each condition attribute in C is calculated by Eq. (1).
3. All λ are sorted in a descending order, and the attribute with the smallest λ is selected. If two attributes have the same λ, the attribute with the least attribute value can be selected. If λ is the same and meanwhile the attribute value is also the same, the attribute can be selected at random.
4. The selected attribute a_i is put into *red* and is reduced from C, from Eq. (2).

5. If U is Ø, go to step 7; otherwise, go to the next step.
6. The attributes in *red* are combined with the remaining attributes in C. Repeat steps 2–5.
7. Output *red*, the reduction ends.

4. Wavelet Neural Network

WNN combines the time-frequency localization characteristics of wavelet transform and the self-learning function of neural network, which can significantly improve the performance [25].

For the discrete wavelet transform, through the wavelet function

$$\psi_{m,n}(t) = 2^{-\frac{m}{2}}\psi(2^{-m}t - n), \quad m, n \in Z \tag{3}$$

a set of orthogonal basis of the space can be constructed, $L^2(R) = \oplus_m W_m$. Using the multi-resolution analysis for the space $L^2(R)$, a series of closed subspaces $\cdots \subset V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \cdots$ can be given. $V_m = span\{2^{-\frac{m}{2}}\varphi(2^{-m}t - k)\}$, $V_{m+1} = V_m \oplus W_m$, and $\varphi(t)$ is the scaling function corresponding to the wavelet. Then, the function $f(t)$ in $L^2(R)$ can be decomposed into

$$f(t) = \sum_{m,n} \langle f, \psi_{m,n} \rangle \psi_{m,n}(t) \tag{4}$$

It can be further expressed as

$$\hat{f} = \sum_{m,n}^{N} w_{m,n}\psi_{m,n}(t) = \sum_{m,n}^{N} w_{m,n}2^{-\frac{m}{2}}\psi(2^{-m}t - n) \tag{5}$$

The above formula can be realized directly by a feed forward WNN with a single hidden layer, as shown in Fig. 1. In this figure, $x_i (i = 1, \ldots, I)$ is the network input, $y_k (k = 1, \ldots, K)$ the network output, w_{ji} the connection weight from the input layer to the hidden layer, w_{kj} the connection weight from the hidden layer to the output layer, and I, J, K are the numbers of input layer nodes, hidden layer nodes and output layer nodes, respectively. The orthogonal wavelet function $\psi_{m,n}(t)$ is the activation function of hidden layer nodes.

In fact, the wavelet in Eq. (3) is also called the binary wavelet. The wavelet analysis has the focusing effect. In general, the selected magnification 2^{-m} corresponds to a part of the observed signal. To know the details of the signal in depth, the magnification can be increased (i.e. decrease m); on the contrary, to roughly observe the signal, the magnification can be decreased (i.e. increase m).

For the network in Fig. 1, for different selected wavelet basis functions, the constituted learning algorithms are also different. In this paper, Mexico hat basis function $\psi(t) = (1 - t^2)e^{-t^2/2}$ is used to derive BP learning algorithm based on the orthogonal wavelet.

Suppose that the learning sample set is $\{X_i, \hat{Y}_i\}_i^N$, where X is the input vector $(x_1, x_2, \ldots, x_I)$, $\hat{Y}$ the expected output vector $(\hat{y}_1, \hat{y}_2, \ldots, \hat{y}_K)$, and N is the number of learning sample. Then, the input of the jth node in the hidden layer is $s_j = \sum_{i=1}^{I} w_{ji}x_i (j = 1, 2, \ldots, J)$, and the output of the jth node in the hidden layer is

$$h_j = \psi_{m_j,n_j}(s_j) = 2^{-\frac{m_j}{2}} e^{-\frac{(2^{-m_j}s_j - n_j)^2}{2}} \cos(5 * 2^{-m_j}s_j - 5n_j), \quad (j = 1, 2, \ldots, J) \tag{6}$$

The output is the linear combination of wavelet frame. The kth output is

$$y_k = \sum_{j=1}^{J} w_{kj}h_j, (k = 1, 2, \ldots, K) \tag{7}$$

If the error function of WNN is defined as $E = \frac{1}{2}\sum_{k=1}^{K}(\hat{y}_k - y_k)^2$, the instantaneous gradient vector for each parameter can be obtained by

$$\delta m_j = \frac{\partial E}{\partial m_j} = \sum_{k=1}^{K}(\hat{y}_k - y_k)w_{kj}\frac{\partial h_j}{\partial m_j}, \quad (j = 1, 2, \ldots, J) \tag{8}$$

$$\delta n_j = \frac{\partial E}{\partial n_j} = \sum_{k=1}^{K}(\hat{y}_k - y_k)w_{kj}\frac{\partial h_j}{\partial n_j}, \quad (j = 1, 2, \ldots, J) \tag{9}$$

$$\delta w_{ji} = \frac{\partial E}{\partial w_{ji}} = \sum_{k=1}^{K}(\hat{y}_k - y_k) \cdot w_{kj} \cdot \frac{\partial h_j}{\partial s_j} \cdot \frac{\partial s_j}{\partial w_{ji}}, \quad (i = 1, 2, \ldots, I; \quad j = 1, 2, \ldots, J) \tag{10}$$

$$\delta w_{kj} = \frac{\partial E}{\partial w_{kj}} = (\hat{y}_k - y_k) \cdot h_j, \quad (k = 1, 2, \ldots, K; j = 1, 2, \ldots, J) \tag{11}$$

where

$$\frac{\partial h_j}{\partial m_j} = 2^{-\frac{m_j}{2}} e^{-\frac{(2^{-m_j}s_j - n_j)^2}{2}} [2\ln 2 \cdot s_j(2^{-m_j}s_j - n_j) \cos(5 * 2^{-m_j}s_j - 5n_j) - \frac{1}{2}\ln 2 \cdot \cos(5 * 2^{-m_j}s_j - 5n_j) + 10\ln 2 \cdot s_j \cdot \sin(5 * 2^{-m_j}s_j - 5n_j)] \tag{12}$$

$$\frac{\partial h_j}{\partial n_j} = (2^{-m_j}s_j - n_j)2^{-\frac{m_j}{2}} e^{-\frac{(2^{-m_j}s_j - n_j)^2}{2}} \cos(5 * 2^{-m_j}s_j - 5n_j) + 5 \cdot 2^{-\frac{m_j}{2}} e^{-\frac{(2^{-m_j}s_j - n_j)^2}{2}} \sin(5 * 2^{-m_j}s_j - 5n_j) \tag{13}$$

$$\frac{\partial h_j}{\partial s_j} = -2^{-\frac{m_j}{2}} [2^{-m_j}(2^{-m_j}s_j - n_j) \cos(5 * 2^{-m_j}s_j - 5n_j) + 5 \cdot 2^{-m_j} \sin(5 * 2^{-m_j}s_j - 5n_j)] \tag{14}$$

$$\frac{\partial s_j}{\partial w_{ji}} = x_i \tag{15}$$

For conventional BP algorithm, the network parameters can be adjusted by

$$\theta(k + 1) = \theta(k) + \lambda\Delta\theta(k) + (1 - \lambda)\eta(k)\frac{\partial E_k}{\partial\theta(k)} \tag{16}$$

where θ represents the parameter vector $\{w_{ji}, w_{kj}, m_j, n_j\}$, and $\Delta\theta(k)$ is the parameter difference after last adjustment. λ is the momentum factor that can reduce the sensitivity of network to the error surface details, so as to effectively avoid falling into local minimum. $\eta(k)$ is the learning rate. If $\eta(k)$ is too small, the algorithm has a slow convergence rate and may easily fall into local minimums; if too large, the algorithm has a fast convergence rate, but it may oscillate. Therefore, the adaptive learning rate can be introduced to dynamically adjust $\eta(k)$. The specific adjustment formula is

$$\eta(k) = \begin{cases} 1.05\eta(k - 1) & E(k) < E(k - 1) \\ 0.7\eta(k - 1) & E(k) > 1.04E(k - 1) \\ \eta(k - 1) & \text{otherwise} \end{cases} \tag{17}$$

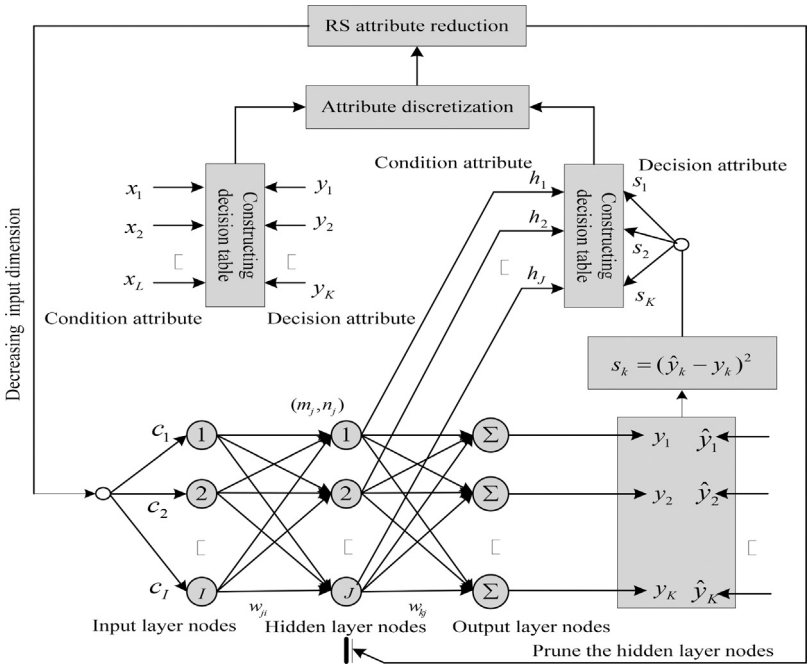

Fig. 2. Integrating principle of 2RS-WNN.

where 1.05, 0.7, 1.04 are empirical values.

WNN derived by orthogonal wavelet transform is actually on the basis of orthogonal compactly supported wavelets constructed by Daubechies and multiresolution analysis. Using the scaling function, the most appropriate scaling and translation parameters are selected to adjust the wavelet coefficient for achieving the best approximation to the function.

5. Integrating realization based on RS and WNN

For WNN modeling, when the input dimension is large, the network structure is very complex, which may result in long training time and low accuracy. Especially, the sample data collected from the complex system not only have a large dimension, but also contain many redundant variables. This may result in an unsatisfactory result for WNN modeling. Theoretical studies have shown that, RS attribute reduction ability can be used to pre-process the sample data for WNN modeling, which can not only remove the redundant information, but also reduce the dimension of the input space by analyzing the internal relation between the sample data.

Network structure design is a hard problem in WNN modeling. Generally, it can be solved by the top-down approach. That is, a network containing enough hidden layer nodes is firstly constructed, and then the nodes are reduced by experience. In this case, the network is redundant, time-consuming, more unstable since it is strongly dependent on personal experience. Hence, there is room for improvement. Research shows that, in WNN, the contribution of some hidden layer nodes to the error is small, with less impact on the network. On the contrary, some hidden layer nodes have great contribution to the error, which can cause the network to fluctuate drastically. To obtain a relatively optimal WNN structure with improved performances, it is desirable to retain the hidden layer nodes with great contribution and to remove the hidden layer nodes with less contribution. Considering RS reduction ability, RS can be used to optimize WNN structure, which can remove redundant nodes and obtain simple topology network.

In summary, the integration method based on RS and WNN (2RS-WNN) uses RS attribute reduction ability to reduce the complexity of WNN modeling and structure. The integration principle is shown in Fig. 2. In 2RS-WNN, RS plays two roles:

1. RS is introduced to reduce the input dimension of WNN, i.e. the input variables are taken as the condition attributes and the output variables as the decision attributes, so as to construct the decision table. RS attribute reduction is conducted on the decision table to remove redundant variables to decreasing input dimensions of WNN.
2. RS is used to optimize the network structure of WNN, i.e. the outputs of each hidden layer node are taken as the condition attributes and the output error squares of each output layer node as the decision attributes, so as to construct the decision table. RS attribute reduction is conducted on the decision table to prune the corresponding hidden nodes and their connection weights for an optimal WNN structure.

According to the above analysis, the steps of 2RS-WNN integration method are as follows:

1. The input variable $x_1, x_2, \ldots, x_L$ are taken as the condition attributes and the output variable $y_1, y_2, \ldots, y_K$ as the decision attributes, to construct the initial decision table for decreasing input dimensions of WNN.
2. The initial decision table obtained by Step 1 is discretized and then RS attribute reduction is conducted.
3. The number of input layer nodes of WNN can be determined by the attribute reduction result obtained in Step 2; the number of output layer nodes can be determined by the practical problems to be solved; the number of hidden layer nodes can be preliminarily determined by the empirical formula

$$J \geq \sqrt{I+K} + a \quad \text{or} \quad J \geq \log_2 I \tag{18}$$

   where a is a positive integer in [1,10].
4. After WNN model is trained, the output $h_1, h_2, \ldots, h_J$ of hidden layer nodes and the squared output error $s_1, s_2, \ldots, s_K$ of output layer nodes are recorded for each sample, forming the initial decision table for the optimal WNN structure, where

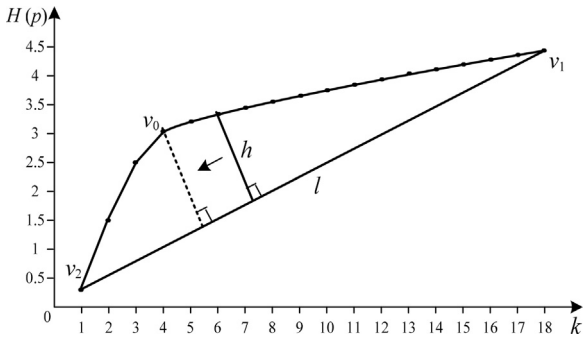

Fig. 3. Determination diagram of best stop point.

$s_k = (\hat{y}_k - y_k)^2$, $\hat{y}_k$ is the expected output and y_k is the actual output of WNN.

5. The initial decision table obtained by Step 4 is discretized and then RS attribute reduction is conducted.
6. For the decision table reduced by Step 5, the structure of WNN is pruned to delete the hidden layer nodes and their connection weights corresponding to the redundant attributes, so as to simplify the structure of WNN.

RS is mostly used to analyze the discrete data. However, for many practical problems, the values of some condition attributes and decision attributes are usually continuous values that need to be discretized before RS analysis. This is an important problem in the application of RS [26].

In essence, the discretization of condition attributes can be reduced to the problem where the condition attribute space is divided by the selected breakpoint. Obviously, in compatibility, the decision table discretized using the different division method may be different from the original decision table. If an attribute has m attribute values, m-1 can be treated as the number of candidate breakpoints. With an increase in the number of attributes and the sample size, the number of candidate breakpoints increases exponentially, so the efficiency of the breakpoint selection algorithm is very important for discretization.

For a certain continuous attribute, its value range can be divided into several intervals, and each interval contains one sample at least. So n samples can be divided into n intervals at most ($O(n)$). That is, the continuous attribute variables can be converted into $O(n)$ discrete variables. Studies show that the maximum entropy (or information) can be achieved when the frequency-probability distribution is with the maximum number of attribute values. Since each non-repetitive attribute value of continuous attributes corresponds to one discrete interval, it can be considered as no information (or entropy) loss in the discretization process. If the discrete random variable is $X \subseteq U$, its entropy can be defined as

$$H(X) = -\sum_{j=1}^{r(d)} p_j \log_2 p_j,\ p_j = k_j / \left|X\right| \tag{19}$$

It can also be recorded as $H(p)$, where $|X|$ is the number of instances, and k_j is the number of instances for decision attribute j($j = 1, 2, \ldots, r(d)$). Generally, the smaller $H(p)$ ($H(p) \geq 0$), the smaller the degree of confusion, this indicates that the individual decision attribute values in the set X are dominant, especially if and only if the decision attributes of instance in X are the same, $H(p) = 0$. This ensures that the following discretization process does not change the compatibility of the decision table.

The key of discretizing continuous attributes is to reasonably determine the number and position of discrete division points. Discretization should meet the following two requirements as much as possible: (1) The space dimension should be as small as possible after the discretization of continuous attributes. (2) The information loss should be as low as possible after the discretization of continuous attributes.

According to the above entropy properties of continuous attributes, an entropy-based discretization method can be proposed. That is, the value range of each continuous attribute is firstly divided into several intervals and each interval corresponds to one non-repetition value; then, two adjacent intervals are selected to merge in order to minimize the entropy difference before and after the merger. In this process, if there is more than one pair of adjacent intervals with the smallest entropy difference before and after the merger, one pair of them are randomly merged. Repeat this merger process until the stop point is obtained, and the division points for the defined intervals are stored [27,28].

It can be found from the analysis that, $H(p)$ is a concave function, which increases monotonically with the increase of k. The increase rate of $H(p)$ decreases when k approaches the maximum. On the function curve of the entropy shown in Fig. 3, the maximum of $H(p)$ is the point v_1 and the corresponding k is also the maximum. If a chord l is drawn from the starting point v_2 to the ending point v_1 of the curve, all points on the concave function curve are above the chord l. It can be concluded from the characteristic of concave function curve that, at the farthest point v_0 on $H(p)$ from the chord l (namely, at the inflection point of the curve), the change of k is larger than that of entropy. It is also at this point that the best balance is achieved between loss of entropy and moderate interval number. In this way, the point v_0 can be used as the best stop point for merging adjacent intervals.

A point is arbitrarily selected on the function curve and a vertical line segment h is created from this point to the chord l. The length of h is proportional to $k_{\max}H(p) - H_{\max}(p)(k-1)$, then the formula can be expressed as

$$h = k_{\max}H(p) - H_{\max}(p)(k-1) \tag{20}$$

where $k_{\max}$ is the maximum number of intervals, and $H_{\max}(p)$ is the corresponding maximum entropy. It is obvious in Fig. 3 that, h at the point v_0 is the maximum value. Accordingly, the points v_0 and corresponding interval numbers can be obtained.

Based on the above analysis, the pseudo code of the entropy-based discretization algorithm can be given as below:

```
Input: data X
Output: discretization value and division points of continuous
attributes in X
For (each continuous attribute) {
  For k non-repeated attribute value, calculate the probability p_k
and entrophy H(p_k) by formula (19), and save the division points;
  Suppose that Symbol = 1, k_0 = k, T_k = 0;
While (Symbol) {
  Select 2 adjacent intervals and combine to minimize the
difference of the entrophy before and after combination, and reset
the division points, save the entrophy after combination;
  Calculate: T_{k-1} = k_0 H(p_{k-1}) - H(p_k)(k-1)
  If (H_{k-1} > H_k)
    {Recalculate the interval probability}
  else
    {Symbol = 0, k = k - 1, save the division points}
}}
Output the continuous value and division points of continuous
attribute.
```

6. Experiment simulation

The experiment environments are: Pentium IV 2.4 GHz CPU, 4GB DDR RAM, 120GB + 7200 RPM hard drive, Windows XP operating system. In the experiment, all simulations for the algorithms were implemented in MATLAB programming environment.

Table 1
Features and calculation formula.

ID	Feature	Formula	Illustration
1	Random %K	$1/3 \times \mathrm{RSV}_t + 2/3 \times \%t\ K_{t-1}$	$\mathrm{RSV}_t = 100 \times (C_t - L_n)/(H_n - L_n)$, C_t is the day closing price, L_n and H_n are the lowest price and highest price within the cycle respectively.
2	Random %D	$1/3 \times \%K_t + 2/3 \times \%D_{t-1}$	It is the smoothed moving average line of %K, %K and %D cycle can be taken as 5 days.
3	W&R	$100 \times (H_n - C_n)/(H_n - L_n)$	L_n and H_n are the lowest price and highest price within the cycle respectively. C_n is the day closing price, and the cycle n can be taken as 5 days.
4	MTM	$\mathrm{C}_t - \mathrm{C}_{t-n}$	C_t and C_{t-n} are the day closing price and first n day closing price respectively, and the cycle n can be taken as 5 days.
5	RSI	$100 \times \Sigma\mathrm{UP}_n/(\Sigma\mathrm{UP}_n + \Sigma\mathrm{DW}_n)$	$\Sigma\mathrm{UP}_n$ is the sum of the rising of daily closing price within the cycle. $\Sigma\mathrm{DW}_n$ is the absolute value of the sum of the falling of daily closing price within the cycle, and the cycle n can be taken as 5 days.
6	BBI	$(\mathrm{MA}_3 + \mathrm{MA}_6 + \mathrm{MA}_{12} + \mathrm{MA}_{24})/4$	It is the mixed moving average, MA_3, MA_6, MA_{12}, MA_{24} denote the moving average within 3, 6, 12, 24 days respectively.
7	5BIAS	$100 \times (C_t - \mathrm{MA}_5)/\mathrm{MA}_5$	C_t is the day closing price. MA_5 denotes the moving average within 5 days.
8	10BIAS	$100 \times (C_t - \mathrm{MA}_{10})/\mathrm{MA}_{10}$	C_t is the day closing price. MA_{10} denotes the moving average within 10 days.
9	MB	$\mathrm{MA}_3 - \mathrm{MA}_6$	MA_3 and MA_6 denote the moving average within 3 and 6 days respectively.
10	PSY	$100 \times \mathrm{DUP}_n/n$	DUP_n denotes the number of rising days within the selected cycle, and the cycle n can be taken as 10 days.
11	AR	$100 \times \mathrm{HO}_n/\mathrm{OL}_n$	HO_n is the sum of distance from highest price to opening price within n days. OL_n is the sum of distance from opening price to lowest price within n days.
12	BR	$100 \times \mathrm{HC}_n/\mathrm{CL}_n$	HC_n is the sum of distance from highest price to closing price within n days. CL_n is the sum of distance from closing price to lowest price within n days.
13	MV	$(V_1 + V_2 + \ldots + V_n)/n$	$V_1, V_2, \ldots, V_n$ denote the daily trading volume within a given cycle, the cycle n can be taken as 5 days.
14	TAPI	V_n/C_n	V_n is the day trading volume. C_n is the day closing price.
15	VR	$100 \times \Sigma\mathrm{VUP}_n/\Sigma V_n$	$\Sigma\mathrm{VUP}_n$ is the sum of rising daily trading volume within the cycle, ΣV_n is the total volume within the cycle, and the cycle n can be taken as 5 days.

6.1. Selection of experimental data

To verify the performance of the method, all experimental data used is from 5 famous stock market indices (www.resset.cn), namely, SSE Composite Index (China), CSI 300 Index (China), All Ordinaries Index (Australian), Nikkei 225 Index (Japan) and Dow Jones Index (USA). The original data are in the range of: (1) 04/10/2009–06/24/2014 from SSE Composite Index; (2) 02/09/2009–04/02/2014 from CSI 300 Index; (3) 04/21/2009–03/26/2014 from All Ordinaries Index; (4) 03/15/2009–05/24/2014 from Nikkei 225 Index; (5) 10/22/2009–07/18/2014 from Dow Jones Index. The original data include the opening price, the closing price, the high price, the low price, and transaction volume. To facilitate the modeling and test, the original data need to be pre-transformed into 15 features through calculation formula in advance, as shown in Table 1. The statistics of 15 features for SSE Composite Index are shown in Table 2. These 15 features are used as the inputs of the prediction model. The output of the prediction model is the stock price trend T, which can be defined by the percent yield R, namely, Drop: $R \in (-\infty, -0.5\%]$, Stability: $R \in (-0.5\%, 0.5\%)$ and Rise: $R \in [0.5\%, \infty)$. R can be calculated by

$$R_t = (P_t - P_{t-1})/P_{t-1} \times 100\% \tag{21}$$

where P_t is the series of stock price.

In addition, the different features (attributes) usually have different value ranges, which may affect the prediction results. The normalization method needs to be used to standardize the sample data before modeling. The value a of the attribute A can be mapped into the normalization interval (L, H) through the formula

$$a' = L + (H - L) \cdot \frac{a - a_{\min}}{a_{\max} - a_{\min}} \tag{22}$$

where $a_{\min}$, $a_{\max}$ are the minimum and maximum of the attribute A, respectively. Here, the normalization interval is uniform and set as $(-0.5, 0.5)$.

Table 2
Statistics of 15 features for SSE Composite Index.

ID	Feature	Maximum	Minimum	Average	Standard deviation
1	Random %K	97.201	5.816	52.334	24.021
2	Random %D	93.160	9.308	52.302	20.733
3	W&R	100.000	0.000	47.620	33.419
4	MTM	318.570	−339.320	3.003	49.335
5	RSI	100.000	0.000	49.915	28.773
6	BBI	2081.753	376.727	1087.394	415.682
7	5BIAS	29.020	−16.731	0.102	3.172
8	10BIAS	53.285	−21.516	0.246	5.144
9	MB	128.827	−108.958	1.119	17.939
10	PSY	90.000	10.000	51.117	16.286
11	AR	473.016	10.875	96.524	56.412
12	BR	378.016	12.213	103.406	54.956
13	MV	363.553	1.393	44.394	42.289
14	TAPI	30,422.995	213.594	4493.407	3567.456
15	VR	100.000	0.000	50.232	23.172

6.2. Evaluation index of model performance

To accurately evaluate the performance of the prediction model for stock price trend, reasonable evaluation indices need to be

Table 3
Initial decision table for decreasing input dimension of WNN.

Sample ID	x_1	x_2	x_3	x_4	x_5	...	x_{15}	T
1	34.121	11.435	22.374	−11.027	32.175	...	15.142	1
2	15.460	50.127	5.166	−80.179	12.040	...	26.750	3
3	54.029	19.232	43.185	215.156	60.137	...	31.082	2
⋮	⋮	⋮	⋮	⋮	⋮	...	⋮	⋮
500	76.258	34.812	9.454	47.550	53.022	...	40.531	3

selected in advance. Through the analysis and study on relevant literatures, seven evaluation indices are included: Root Mean Square Error (RMSE), Mean Absolute Difference (MAD), Mean Absolute Percentage Error (MAPE), Directional Symmetry (DS), Correct Uptrend (CP), Correct Downtrend (CD) and Training Time (TT). RMSE, MAD, MAPE are used to measure the error between actual values and predicted values. DS, CP, CD are used to measure the accuracy of the prediction. DS is a statistical measure of model performance in predicting the direction of stock price change in percentage form. CP is the correct rate of rising trend prediction for stock price change. CD is the correct rate of dropping trend prediction for stock price change. The formulas of the six evaluation indices are:

$$\mathrm{RSME} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(P_i - T_i)^2} \tag{23}$$

$$\mathrm{MAD} = \frac{1}{n}\sum_{i=1}^{n}|P_i - T_i| \tag{24}$$

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{P_i - T_i}{P_i}\right| \times 100\% \tag{25}$$

$$\mathrm{DS} = \frac{100}{n}\sum_{i=1}^{n} d_i, \quad d_i = \begin{cases} 1 & (P_i - P_{i-1})(T_i - T_{i-1}) \geq 0 \\ 0 & \text{Other} \end{cases} \tag{26}$$

$$\mathrm{CP} = \frac{100}{n_1}\sum_{i=1}^{n} d_i,\ d_i = \begin{cases} 1 & (T_i - T_{i-1}) > 0 \text{ and } (P_i - P_{i-1})(T_i - T_{i-1}) \geq 0 \\ 0 & \text{Other} \end{cases} \tag{27}$$

$$CD = \frac{100}{n_2}\sum_{i=1}^{n} d_i,\ d_i = \begin{cases} 1 & (T_i - T_{i-1}) < 0 \text{ and } (P_i - P_{i-1})(T_i - T_{i-1}) \leq 0 \\ 0 & \text{Other} \end{cases} \tag{28}$$

where T is the actual value, P the predicted value, n the number of samples, n_1 the number of the samples with rising trend and n_2 the number of the samples with dropping trend.

6.3. Experimental scheme

6.3.1. Numerical Example A

The modeling steps of 2RS-WNN are demonstrated for stock price trend and the effects of different training set sizes on the performance of 2RS-WNN model are analyzed. Firstly, 100, 300, 500, 600, 800, 1000 samples are randomly selected from the original data of SSE Composite Index to constitute the training sets. At the same time, the samples of 20% size of the training set are randomly selected from the remaining data to construct the corresponding test set. Then, based on 10-fold cross validation, 2RS-WNN prediction model is established for each training set and is then tested.

Table 4
Discretized decision table for decreasing input dimension of WNN.

Sample ID	x_1	x_2	x_3	x_4	x_5	...	x_{15}	T
1	3	1	1	5	3	...	2	1
2	1	5	1	3	1	...	3	3
3	5	2	2	8	6	...	3	2
⋮	⋮	⋮	⋮	⋮	⋮	...	⋮	⋮
500	7	3	1	2	5	...	4	3

6.3.2. Numerical Example B

2RS-WNN is compared with BP-NN, RBF-NN, ANFIS-NN, SVM, WNN, RS-WNN to validate its superiority in stock price trend prediction. 500 samples are firstly randomly selected from the original data of the above-mentioned 5 stock market indices to construct the training set with 500 samples and the test set with 100 samples. Then, the prediction model based on BP-NN, RBF-NN, ANFIS-NN, WNN, RS-WNN, 2RS-WNN are trained using 10-fold cross validation and tested for stock price trend. Finally, performances of the models are compared to verify the use of 2RS-WNN in stock price trend prediction. In the experiment, except 2RS-WNN, the number of hidden layer nodes of BP-NN, RBF-NN, ANFIS-NN, WNN, RS-WNN can be determined by Exhaustive Attack Method (EAM). It should be mentioned that, since the simple SVM can only deal with two-classification problem, the multi-classification problem need to be divided into several two-classification problems. Here, the stock price trend prediction is a three-classification problem, including Drop, Stability and Rise. Hence, the problem needs to train 3 two-classification SVMs to obtain the final prediction result.

6.4. Model demonstration and result analysis

6.4.1. Numerical Example A

To demonstrate the modeling steps of 2RS-WNN, an example for SSE Composite Index is given on the basis of the training set with 500 samples and the test set with 100 samples. Parameter settings: w_{ji}, w_{kj}, m_j, n_j can be initialized as the random number in [−0.5, 0.5], and the maximum number of training can be taken as 500. The specific modeling steps are as follows:

1. For the training sample set, the feature $x_1, x_2, \ldots, x_{15}$ of stock price trend can be taken as the condition attributes, the stock price trend T as the decision attribute, so as to construct the initial decision table for decreasing input dimension of WNN, as shown in Table 3.
2. The data in Table 3 is automatically discretized by introduced entropy-based algorithm, obtaining the discretized decision table as shown in Table 4. Then, RS is used to reduce redundant condition attributes in the discretized decision table.
3. Through RS attribute reduction, the feature $x_3, x_4, x_6, x_8, x_9, x_{12}, x_{14}$ are redundant, and the feature $x_1, x_2, x_5, x_7, x_{10}, x_{11}, x_{13}, x_{15}$ are taken as the input of WNN (8 input layer nodes); the output of WNN is the stock price trend T (only one output node); the

Table 5
Initial decision table for optimal WNN structure.

Sample ID	h_1	h_2	h_3	h_4	h_5	...	h_{10}	s
1	0.1438	0.2105	0.7021	0.02706	0.0001	...	0.0201	0.2132
2	0.0300	−0.6870	−0.1548	−0.0196	0.0045	...	0.7048	0.0207
3	0.0214	0.2032	0.0023	−0.1564	0.2047	...	−0.0502	0.0325
⋮	⋮	⋮	⋮	⋮	⋮	...	⋮	⋮
500	0.1764	0.2459	0.1505	−0.2428	0.0210	...	0.0510	0.1401

Table 6
Discretized decision table for optimal WNN structure.

Sample ID	h_1	h_2	h_3	h_4	h_5	...	h_{10}	s
1	4	2	9	5	1	...	4	4
2	1	6	3	5	1	...	13	1
3	1	2	4	4	2	...	5	1
⋮	⋮	⋮	⋮	⋮	⋮	...	⋮	⋮
500	5	2	6	4	1	...	5	3

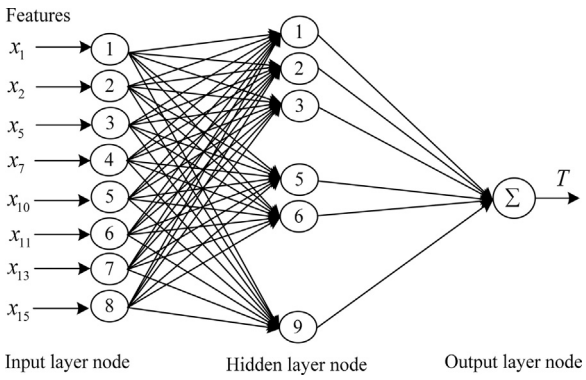

Fig. 4. WNN structure pruned for stock price trend prediction.

number of hidden layer nodes can be preliminarily taken as 10 by Eq. (18).

4. WNN prediction model is trained by BP algorithm with adaptive learning rate, and the output $h_1, h_2, \ldots, h_{10}$ of hidden layer nodes and the squared output error s of output layer node (only an output node) are recorded for each sample, so as to construct the initial decision table for optimal WNN structure, as shown in Table 5.
5. The entropy-based discretization algorithm is used to process the data in Table 5, obtaining the discretized data as shown in Table 6. Then, RS attribute reduction is conducted to the discretized data.
6. It can be known from the reduction result that, the redundant hidden node 4, 7, 8, 10, so the corresponding connection weights need to be pruned. The pruned WNN structure contains the hidden nodes 1, 2, 3, 5, 6, 9, and the prediction model is obtained for SSE Composite Index by the training, as shown in Fig. 4.

According to above demonstration steps, the prediction results and performance comparison of 2RS-WNN model based on different training sets are given in Table 7 for stock price trend of SSE Composite Index.

It can be seen from Table 7 that, for stock price trend of SSE Composite Index, with the increase of the training set size, RMSE, MAD, MAPE, DS predicted by 2RS-WNN model are improved in different degrees. This indicates that, while the number of training samples is insufficient, the patterns contained in training samples cannot ensure a satisfactory 2RS-WNN prediction model, with some improvement room for prediction performance. On the contrary, the larger the training sample set size is, the richer the patterns are in the training samples. This can improve 2RS-WNN training effect for good prediction. However, the improvement of DS index is not significant when the number of training samples is 500 or more. DS value corresponding to the training set with 500, 600, 800, 1000 samples are 65.74%, 65.96%, 66.15% and 66.32%, respectively. Considering the balance between prediction accuracy and training time, it is more appropriate to select 500 training samples for model performance comparison.

In addition, with the increase of training sample size, after the deletion of redundant features through RS pre-processing, the reduced feature set is basically composed of 1, 2, 5, 7, 10, 11, 12, 13, 15. On this basis, the number of WNN hidden layer nodes reduced by RS is generally controlled within 8. It is not difficult to understand that, through RS pre-processing, the input dimension of WNN is reduced, meanwhile the redundant hidden layer nodes and the corresponding connection weights are pruned by RS. The structure of WNN model is greatly simplified under the premise of ensuring prediction accuracy.

6.4.2. Numerical Example B

The performance comparison of different models is shown in Table 8 on the basis of 500 training samples for 5 stock market indices. These models are established by BP-NN, RBF-NN, ANFIS-NN, SVM, WNN, RS-WNN and 2RS-WNN.

It can be found from Table 8 that, compared with 3 neural networks (BP-NN, RBF-NN, ANFIS-NN), SVM and 3 WNNs (WNN, RS-WNN, 2RS-WNN) generally have superiorities in training time, prediction error and prediction accuracy. This is consistent with the conclusions from theoretical analysis. In other words, the located wavelet function is taken as the activation function of hidden layer in WNN, instead of Sigmoid function in conventional neural networks. The time-frequency window of wavelet function can be adjusted adaptively. This makes WNN a great advantage over conventional neural networks for nonstationary nonlinear problems like stock price trend prediction.

For SVM and 3 WNNs, the prediction accuracy of SVM is better than that of WNN, but less than that of RS-WNN and 2RS-WNN. It is not difficult to understand that, although SVM has more excellent classification ability than WNN, it cannot extract features like RS to reduce the prediction error caused by the system itself. In addition, the training time of SVM is more than that of 2RS-WNN.

In 3 WNNs, RS-WNN model has advantages over WNN model on the overall performance. DS of RS-WNN models corresponding to SSE Composite Index, CSI 300 Index, All Ordinaries Index, Nikkei 225 Index and Dow Jones Index are 64.01%, 66.24%, 65.80%, 65.41% and 66.12%, respectively, which improve by 3.78%, 5.73%, 3.45%, 3.59% and 2.67% in contrast with simple WNN model. This shows that, through RS attribute reduction, the redundant features of stock price trend are reduced. This can greatly eliminate multiple correlations between the features (indices) and decrease the input dimensions of WNN, making the WNN training more direct and effective without the interference of redundant features. It is also shown that, compared with the improvement of model performance, the negative impacts of information loss caused by entropy-based discretization process are very small.

Table 7
Prediction results and performance comparison of 2RS-WNN model based on different training sets for stock price trend of SSE Composite Index.

Samples numbers	Features reduced by RS	Model structure	TT (s)	RMSE	MAD	MAPE	DS%
100	1, 2, 5, 11, 13	5-3-1	0.5	97.05	84.11	2.54	60.36
300	1, 2, 5, 7, 11, 13, 15	7-6-1	1.1	83.62	46.35	1.97	63.83
500	1, 2, 5, 7, 10, 11, 13, 15	8-6-1	1.7	68.50	27.18	1.35	65.74
600	1, 2, 5, 7, 11, 12, 15	7-5-1	3.8	55.67	24.52	1.32	65.96
800	1, 2, 5, 7, 10, 11, 12, 13, 15	9-8-1	4.9	50.01	24.19	1.28	66.15
1000	1, 2, 4, 5, 10, 11, 12, 13, 15	9-7-1	6.2	48.24	23.05	1.17	66.32

Table 8
Performance comparison of different models for 5 stock market indices.

Data set	Algorithm	Model structure	TT (s)	RMSE	MAD	MAPE	DS%	CP%	CD%
SSE Composite Index (China)	BP-NN	15-12-1	14.3	216.2	192.5	4.4	59.8	61.6	58.4
	RBF-NN	15-7-1	9.7	175.7	141.8	3.3	61.1	59.2	62.8
	ANFIS-NN	15-8-1	12.4	153.2	137.7	3.5	61.6	62.1	58.5
	SVM	3-SVM	3.6	126.5	114.6	2.6	62.4	63.3	62.8
	WNN	15-8-1	5.5	135.4	125.2	3.2	60.2	62.2	58.4
	RS-WNN	8-7-1	4.2	107.6	92.1	2.0	64.0	63.5	65.8
	2RS-WNN	8-5-1	1.7	98.5	77.2	1.8	65.8	66.7	65.9
CSI 300 Index (China)	BP-NN	15-10-1	13.8	167.4	133.2	5.9	58.5	57.1	60.3
	RBF-NN	15-7-1	7.5	148.1	124.0	5.3	59.1	60.2	58.4
	ANFIS-NN	15-6-1	11.4	142.3	117.1	3.5	59.9	57.1	60.1
	SVM	3-SVM	3.5	106.2	79.4	2.8	62.6	63.9	62.4
	WNN	15-8-1	4.6	115.1	75.2	2.9	60.5	62.7	59.8
	RS-WNN	7-9-1	4.1	97.2	59.0	2.7	66.2	65.5	66.8
	2RS-WNN	7-5-1	1.8	84.3	45.1	1.8	66.4	67.3	66.2
All Ordinaries Index (Australian)	BP-NN	15-14-1	16.2	239.1	207.8	4.0	60.7	63.8	57.9
	RBF-NN	15-9-1	9.5	180.2	145.1	3.3	61.9	60.9	61.4
	ANFIS-NN	15-10-1	12.0	165.3	122.4	4.0	61.3	61.2	63.4
	SVM	3-SVM	4.2	114.1	97.2	1.8	64.5	62.7	65.7
	WNN	15-6-1	6.4	127.9	105.1	2.0	62.4	61.5	64.0
	RS-WNN	6-8-1	5.7	106.9	84.8	1.3	65.8	64.3	66.5
	2RS-WNN	6-3-1	2.1	97.6	70.2	1.2	65.9	64.3	67.4
Nikkei 225 Index (Japan)	BP-NN	15-17-1	28.5	293.4	228.5	8.3	57.3	59.7	58.7
	RBF-NN	15-12-1	16.7	248.5	185.4	6.8	59.1	60.2	58.4
	ANFIS-NN	15-11-1	20.3	202.7	157.1	7.6	57.4	58.3	62.5
	SVM	3-SVM	7.6	139.1	114.3	4.6	64.3	65.7	66.3
	WNN	15-12-1	12.1	165.4	148.4	7.0	61.8	64.3	64.3
	RS-WNN	9-12-1	10.4	125.3	102.9	4.0	65.4	65.9	66.9
	2RS-WNN	9-8-1	5.5	114.6	87.3	3.0	65.5	66.2	67.5
Dow Jones Index (USA)	BP-NN	15-12-1	21.5	188.2	195.7	4.8	60.9	58.7	61.8
	RBF-NN	15-14-1	12.0	152.2	170.3	4.3	63.7	62.5	62.1
	ANFIS-NN	15-9-1	16.4	165.7	192.1	4.4	61.0	59.6	63.1
	SVM	3-SVM	5.5	123.6	135.4	2.0	65.2	64.3	65.5
	WNN	15-10-1	9.3	133.8	145.2	3.8	63.4	62.8	63.2
	RS-WNN	7-10-1	7.7	116.52	127.6	1.6	66.1	65.3	66.4
	2RS-WNN	7-7-1	4.1	105.33	112.4	1.3	66.8	66.4	67.8

For 2RS-WNN model, its DS corresponding to SSE Composite Index, CSI 300 Index, All Ordinaries Index, Nikkei 225 Index and Dow Jones Index are 65.75%, 66.37%, 65.97%, 65.52% and 66.75%, respectively, and only improve by 0.74%, 0.13%, 0.17%, 0.11% and 0.63% compared with RS-WNN model. However, TT save 2.5 s, 2.3 s, 3.6 s, 4.9 s and 3.6 s, respectively. This shows that, using RS to reduce the hidden layer nodes on the basis of RS-WNN, can essentially optimize the network structure of WNN. It can greatly shorten the training time with accuracy improvement, which will promote the actual application of the proposed method. The obtained prediction results can also accurately reflect the overall trend of the stock market.

7. Discussions on incremental learning

To improve the practicality, the incremental learning way of the proposed method needs to be discussed. It can be realized by adjusting the parameters of WNN after the determination of network structure. In order to ensure that the parameter adjustment of each incremental learning is not too large, the range of parameter adjustment need to be set in advance in the learning algorithm. (The range of parameter value and the range of parameter adjustment is not a concept. The range of parameter adjustment is generally less than the range of parameter values.) This can ensure that the known knowledge is not destroyed in the network and the new knowledge is gradually being learned. In addition, the range of parameter adjustment can also effectively overcome the shortcomings such as over-fitting and falling into local minimum. To adjust the parameters is actually to maintain a balance between the existing knowledge of the classifier and the new knowledge contained in the samples. If the existing knowledge of the classifier is more concerned, it should learn slowly; if the new knowledge is more important, it should learn quickly. For the incremental learning of BP algorithm, it can be seen as fast learning if the multiple iterations are made for a new sample, and slow learning if only an iteration or without iteration. In general, the fast learning uses the new knowledge to quickly update the network parameters, which may cause instability in network performance. Therefore, only when the old knowledge is already out of date or the network needs to quickly learn new knowledge to adapt to the changes of sample distribu-

tion, is the fast learning used. In addition, some details also need more in-depth discussion on future work.

8. Conclusion

Stock market is an important part of the economic system. The process of stock price evolution is decided by many economic individuals and economic factors working together, resulting in great uncertainty, and bringing difficulties to stock price trend prediction. For this reason, an integrated prediction method based on RS and WNN is proposed. RS has an excellent attribute reduction capability. WNN is good at nonlinear system modeling. Combining them, the proposed method can decrease the input dimension and optimize the structure for WNN modeling. This can reduce the calculation complexity with the improvement of stock price trend prediction. Simulations show that, the proposed prediction method reduces the network size, simplifies the network structure, and improves the training time and prediction accuracy. Thereby, it provides an effective solution to the prediction of stock price trend. It is believed that, with further research, through selecting more reasonable indices, the prediction accuracy can be further improved for stock price trend. In addition, WNN is used to predict the trend of stock price, but the application of typical three-layer network structure with single hidden layer is mainly studied. If it is extended to WNN structure with a number of hidden layers, its performance for stock price trend prediction will be the next focus of the study.

Conflicts of interest

The authors declare that they have no conflict of interest. All procedures performed in studies involving human participants were in accordance with the ethical standards of the institutional and/or national research committee and with the 1964 Helsinki declaration and its later amendments or comparable ethical standards. This article does not contain any studies with animals performed by any of the authors. Informed consent was obtained from all individual participants included in the study.

Acknowledgement

The author gratefully acknowledges supports of the National Social Science Foundation of China (Grant No. 16CJY060).

References

[1] Y.X. Chen, Stock Investment, Sichuan University Press, Chengdu, 2005.
[2] R.D. Edwards, J. Magee, W.H.C. Bassetti, Technical Analysis of Stock Trends, CRC Press, Boca Raton, 2007.
[3] Y.Y. Win, Time series data mining method based on ARIMA model improvements, J. Harbin Univ. Commer. 30 (6) (2014) 675–676, 681.
[4] M.A. Jin, L.C. Cheng, Grey forecasting model for the price of the stock, J. Yanbian Univ. 40 (1) (2014) 54–57.
[5] J. Xiao, Y. Fan, B. Zhu, Applied research on business forecast model, Soft Sci. 20 (4) (2014) 12–15.
[6] T. White, Economic prediction using neural networks: the case of IBM daily stock returns, Proc. IEEE Int. Conf. Neural Netw. 2 (6) (1988) 451–458.
[7] T. Kimoto, K. Asakawa, M. Yoda, et al., Stock market prediction system with modular neural networks, Proceedings of 1990 International Joint Conference on Neural Networks, vol. I (1990) 1–6.
[8] N. Baba, M. Kozaki, An intelligent forecasting system of stock price using neural networks, Proceedings of 1992 International Joint Conference on Neural Networks, vol. I (1992) 371–377.
[9] M.H. Pesaran, M.A. Timermann, Recursive Modeling Approach to Predicting Stock Returns, University of Cambridge Press, Cambridge, 1999.
[10] K. Kwong, Financial Forecasting Using Neural Network or Machine Learning Techniques (Thesis of Electrical Engineering), University of Queensland, October 2001.
[11] F. Oliveira, C.N. Nobre, L.E. Zarate, Applying artificial neural networks to prediction of stock price and improvement of the directional prediction index – case study of PETR4, Expert Syst. Appl. 40 (18) (2013) 7596–7606.
[12] C.K. Ayo, Comparison of ARIMA and artificial neural networks models for stock price prediction, J. Appl. Math. 2014 (1) (2014) 1–7.
[13] E. Hadavandi, H. Shavandi, A. Ghanbari, Integration of genetic fuzzy systems and artificial neural networks for stock price forecasting, Knowl. Based Syst. 23 (8) (2013) 800–808.
[14] Z.J. Yu, S.L. Yang, Z. Zhang, J. Jiao, Stock returns prediction based on error-correction grey neural network, Chin. J. Manag. Sci. 23 (12) (2015) 20–26.
[15] A.L. Leonel, A.S.F. Ricardo, G.L. Guilherme, Maximum and minimum stock price forecasting of Brazilian power distribution companies based on artificial neural networks, Appl. Soft Comput. 35 (2015) 66–74.
[16] Q.H. Zhang, A. Benveniste, Wavelet network, IEEE Trans. Neural Netw. 3 (6) (1992) 889–898.
[17] Z. Pawlak, Rough sets, Int. J. Comput. Inf. Sci. 1 (11) (1982) 341–356.
[18] Z. Pawlak, Rough Sets – Theoretical Aspects of Reasoning about Data, Kluwer Academic Publisher, Dordrecht, 1990.
[19] C.Z. Wang, M.W. Shao, Q. He, Y.H. Qian, Y.L. Qi, Feature subset selection based on fuzzy neighborhood rough sets, Knowl. Based Syst. 111 (1) (2016) 173–179.
[20] C.Z. Wang, Y.L. Qi, M.W. Shao, Q.H. Hu, D.G. Chen, Y.H. Qian, Y.J. Lin, A fitting model for feature selection with fuzzy rough sets, IEEE Trans. Fuzzy Syst. PP (99) (2016) 1–13.
[21] C.Z. Wang, M.W. Shao, B.Q. Sun, Q.H. Hu, An improved attribute reduction scheme with covering based rough sets, Appl. Soft Comput. 26 (C) (2015) 235–243.
[22] B.B. Qu, Y.S. Lu, Rough set-based algorithm for attribute reduction, J. Huazhong Univ. Sci. Technol. 33 (8) (2005) 30–33.
[23] B.F. Wu, Q. Li, W.Z. Song, Inductive learning approach to knowledge representation system based on rough set theory, Control Decis. 14 (3) (1999) 206–211.
[24] J. Zhao, G.Y. Wang, Z.F. Wu, H. Tang, et al., An efficient approach to compute the feature core, Mini-micro Syst. 24 (11) (2003) 1950–1953.
[25] Q.H. Zhang, Using wavelet network in nonparametric estimation, IEEE Trans. Neural Netw. 8 (2) (1997) 227–236.
[26] F.M. Usama, I.B. Keki, Multi-interval discretization of continuous-valued attributes for classification learning, in: Proceedings of the 13th International Joint Conference on Artificial Intelligence, vol. 2, Morgan Kaufman, San Mateo, CA, 1993, pp. 1022–1027.
[27] S.J. Hong, Use of contextual information for feature ranking and discretization, IEEE Trans. Knowl. Data Eng. 9 (5) (1997) 718–730.
[28] H. Xie, H.Z. Chang, D.X. Niu, Discretization of continuous attributes in rough set theory based on information entropy, Chin. J. Comput. 28 (9) (2005) 1570–1574.

Lei Lei was born in Dec. 1981. She is a doctor of economics. Her research activity mainly focuses on circulation efficiency, the development of the retail industry, pattern recognition, nonlinear system modeling.