



# Trading strategy design in financial investment through a turning points prediction scheme

Xiuquan Li, Zhidong Deng\*, Jing Luo

State Key Laboratory of Intelligent Technology and Systems, Tsinghua National Laboratory for Information Science and Technology, Department of Computer Science, Tsinghua University, Beijing 100084, China

## ARTICLE INFO

### Keywords:

Trading strategy  
Ensemble artificial neural network  
Chaotic analysis  
Turning points prediction  
Financial time series

## ABSTRACT

Turning points prediction has long been a tough task in the field of time series analysis due to its strong nonlinearity, and thus has attracted many research efforts. In this study, the turning points prediction (TPP) framework is presented and further employed to develop a novel trading strategy designing approach to financial investment. The TPP framework is a machine learning-based solution incorporating chaotic dynamic analysis and neural network modeling. It works on the ground of a nonlinear mapping deduced in financial time series through chaotic analysis. An event characterization method is created in TPP framework to characterize trend patterns in ongoing financial time series. The main contributions of this paper are (1) it presents an ensemble learning based TPP framework, within which the nonlinear mapping is approximated by the ensemble artificial neural network (EANN) model with a new parameters learning algorithm; (2) a genetic algorithm (GA) based threshold optimization procedure is described with a newly defined performance measure, named TpMSE, which is used as a cost function; and (3) a trading strategy designing approach is proposed based on the TPP framework. The proposed approach was applied to the two real-world financial time series, i.e., an individual stock quote time series and the Dow Jones Industrial Average (DJIA) index time series. Experimental results show that the proposed approach can help investors make profitable decisions.

© 2008 Elsevier Ltd. All rights reserved.

## 1. Introduction

Financial investment activities have become an indispensable component of the everyday life in our modern societies globally. Investors, institutional or individual, are now provided with a tool of financing via multiple channels, i.e., stocks, bonds, commodities, currencies, options and futures, and so forth. However, the high nonlinearity inherent in the behavior of financial markets and the needed professional knowledge make it substantially difficult for investors to make the right investment decisions promptly.

In this context, many efforts have been made to provide decision-making support systems for investors so as to facilitate their analysis work (Chun & Park, 2005; Kodogiannis & Lolis, 2002; Li & Kuo, 2008; Povinelli, 2001; Skabar, 2005; Sun et al., 2005). Among these studies, time series prediction has been proved to be an effective way to help decision making. These studies mainly focused on the issue of next-value prediction, which is forecasting the future value of time series at the oncoming time step, given the historical observation until the current time (Chen, Ji, Zhao, & Nian, 2005; Kodogiannis & Lolis, 2002; Sun et al., 2005). Most of the players in financial markets, however, do not care much about the

exact value of next time step; instead, they show great interest in how to predict the future trend of a financial time series and when it is time to alter trading strategy.

Turning points prediction, the prediction of peaks and troughs, can assist us to judge market trend and capture profitable opportunities. It is of great use to both macroeconomic policy-makers and operators in finance world. Some research endeavors trying to tackle this problem have been made with statistical approach (Kling, 1987; Poddig & Huber, 1999; Wecker, 1979). They are mainly developed from the Monte Carlo-based regression approach introduced by Wecker. These methods, based on linear statistical models of ARIMA or VAR, work well with turning points prediction of some macro-economic time series, e.g., a nation's GDP time series. Nevertheless, nonlinear dynamic systems emerge extensively in financial fields. The constraints of stationarity, residual normality, and independence are generally not met for many cases. In fact, turning points are often claimed to be essentially nonlinear phenomena. Nonlinear specifications are better than simpler linear models at reproducing the cycle features of real economic time series, such as GDP (Camacho & Quiros, 2002). Nonlinear models seem to be the natural one to forecast turning points, especially for chaotic time series.

Chaotic time series is usually a sequence of observed values from a complex nonlinear dynamic system with chaotic characteristics.

\* Corresponding author. Tel./fax: +86 10 62796830.  
E-mail address: [michael@tsinghua.edu.cn](mailto:michael@tsinghua.edu.cn) (Z. Deng).

The existence of chaotic characteristics intrinsic in financial time series, such as in stock market, has been well studied. Previous research has shown the chaotic phenomenon existing in real-world economic and financial time series and proved their chaotic dynamic characteristics (Catherine, Walter, & Michel, 2004; Harrison, Yu, Oxley, Lu, & George, 1999; Hesieh, 1989).

Furthermore, greater success has been achieved in next-value prediction of financial time series by harnessing the chaotic characteristics intrinsic in them (Chen et al., 2005; Jaeger & Haas, 2004). But few efforts have been made on how to predict turning points in financial time series, which is especially important to making trading decisions in financial investment, by exploring the rules of the underlying dynamic system of the chaotic time series.

We hereby propose a machine learning solution based on chaotic analysis and neural network modeling, called TPP framework, for turning points prediction. The theoretical fundamental supporting this proposed method has been provided in our previous work, where a new nonlinear mapping between different data points in primitive time series has been derived and proven by theoretical analysis in reconstructed phase space (Li & Deng, 2007). We have also given the definition of a characteristic function, named turning indicator, to quantitatively represent the event occurrence degree (expression that is used for how likely a point in time series was a turning point) of a turning point and identify the meaningful turning points in time series.

This paper is organized as follows. The main concepts that the TPP scheme has built on are at first described in Sections 2.1 and 2.2. In Section 2.3, the ensemble learning based TPP scheme is introduced. A threshold optimization procedure is presented in Section 2.4, incorporating genetic algorithm with a newly defined performance measure, namely TpmSE, as a cost function. In Section 2.5, we give the parameter learning algorithm for the EANN model. A trading strategy designing approach is brought up in Section 3 based on the TPP scheme. The proposed approach is applied to two real-world financial time series in Section 4, where the experimental results are also given. The performance of the methods is evaluated in Section 5. Finally in Section 6, we draw the conclusion.

## 2. Turning points prediction scheme

### 2.1. Theoretical analysis

Our turning points prediction (TPP) scheme is established on the ground of a nonlinear mapping in financial time series deduced through chaotic analysis. The discovered mapping shed light on our model construction process.

The analysis of the dynamic characteristics of one-dimensional time series is traditionally conducted in the reconstructed phase space according to the Taken's theorem. The objective of phase space reconstruction on time series is to rebuild the chaotic attractor in a high dimensional space so as to unveil the dynamic behavior of chaotic time series. Takens has proved that a suitable embedding dimension  $m$  can be used to set up a reconstructed space using the delayed coordinate method (Takens, 1981). The reconstructed space is able to resume the tracks of a chaotic attractor when  $m$  is properly chosen and  $m \geq 2d + 1$ , where  $d$  is the dimension of the dynamic system. The phase space reconstruction theory and Taken's theorem provide a solid theoretical basis for the dynamic analysis of chaotic time series.

Our work adopts the Cao method to determine the minimum embedding dimension of chaotic time series (Cao, 1997). It was proved experimentally to be more stable and efficient than the G-P algorithm does (Grassberger & Procaccia, 1983), which has been used in our previous research as well. The Lyapunov exponent quantitatively measures the chaotic characteristic of a chaotic sys-

tem, named the sensitive dependence on initial condition (SDIC). A lot of experiments have proved that when the trajectories in phase space diverge to a distance  $\epsilon$  times wider than an initial one, the track is no longer determinable (Huang, 2000; Liu, Zhang, & Yu, 2004). In light of this, the Lyapunov time, also known as the critical predictable interval, determines the short-term predictability of the given chaotic system and is defined as  $t_0 = 1/\lambda_1$ , where  $\lambda_1$  is the largest Lyapunov exponent. The numerical computation method proposed by Rosenstein, Collins, and De luca (1993) is employed to compute Lyapunov exponent.

Based on the phase space reconstruction theory and Takens's proof for the differential homomorphism between the reconstructed dynamic system in  $\mathcal{R}^m$  space and the primitive one (Takens, 1981), we discovered that a smooth mapping exists in dynamic evolution of chaotic time series. It further extracts nonlinear dynamic properties of the primitive dynamic system and inspires us to develop a new framework for turning points prediction.

**Theorem 1.** *Provided that a compact manifold of fractal dimension  $d$  in reconstructed  $\mathcal{R}^m$  space is built from chaotic time series through the phase space reconstruction with a properly chosen  $m$ , where  $m \geq 2d + 1$ , such that the manifold keeps differential homomorphism with the primitive dynamic system, there must be a smooth mapping  $\Phi: \mathcal{R}^{m\tau} \rightarrow \mathcal{R}^{p\tau}$*

$$[x_{t+1}, x_{t+2}, \dots, x_{t+p\tau}] = \Phi(x_{t-m\tau+1}, x_{t-m\tau+2}, \dots, x_t), \quad (1)$$

where  $m$  is the embedding dimension,  $\tau$  the reconstructive delay, and  $p$  Lyapunov time.

The proof for Theorem 1 can be found in Li and Deng (2007).

The derived nonlinear mapping is regarded as an evidence of the existence of certain rules intrinsic in the fluctuation of the values of chaotic time series. It indicates that the rebuilt dynamic system is capable of recalling the track trend within certain time steps by exploring self-similar fractal characteristics of chaotic attractor, which makes it possible to predict turning points of chaotic time series. The TPP framework is thus designed through further research on this nonlinear mapping as described in the following sections. The input layer structure of the neural network model and the computation span of the event characterization function are also expressed in (1). To be more concrete, the input layer of the neural network model should include  $m\tau$  neurons and the turning indicator is computed over  $p\tau + m\tau$  time steps as described in the next section.

### 2.2. Event characterization function

A new definition for turning points of time series has been given in our previous work, which leads to a new concept about the criteria for identifying meaningful turning points in financial application, and its reasonableness has also been demonstrated (Li & Deng, 2007).

**Definition 1.** For a given time series  $x_t \in \mathbb{R}$ ,  $t = 1, 2, \dots, m$ , a turning point, i.e. peak or trough, is defined as a time step  $t$ , such that  $t$  is neither located on the upward nor downward side of the time series, and meanwhile, the following value variation (decrease or increase) within  $p\tau$  steps exceeds a specific percentage  $\gamma$ .

Having obtained the turning points from the time series defined as in Definition 1, an event characterization function, namely turning indicator, can be defined as a function over the continual  $m\tau + p\tau$  time steps, i.e.,

$$\Gamma_\gamma(t) = \mathfrak{I}(x_{t-m\tau+1}, \dots, x_t, \dots, x_{t+p\tau}) \in [0, 1], \quad (2)$$

$\Gamma_\gamma(t) = 1$  if  $x_t$  is a peak while  $\Gamma_\gamma(t) = 0$  if  $x_t$  is a trough. The rest points fall between 0 and 1. The detailed description for the computation method of the turning indicator can be found in Appendix.

The proposed event characterization function is used to quantitatively measure the event occurrence degree of a turning point according to the nearby values and therefore provides a tool of modeling the trend patterns in financial time series.

### 2.3. Ensemble neural network modeling method

Considering the nonlinear mapping  $\Phi: \mathcal{R}^{m\tau} \rightarrow \mathcal{R}^{p\tau}$  described in (1), it holds that

$$\begin{aligned} [x_{t-m\tau+1}, \dots, x_t, x_{t+1}, \dots, x_{t+p\tau}] &= [x_{t-m\tau+1}, \dots, x_t, \Phi(x_{t-m\tau+1}, \dots, x_t)] \\ &= \Phi'(x_{t-m\tau+1}, \dots, x_t), \end{aligned} \quad (3)$$

where  $\Phi'$  is actually a transformation of  $\Phi$ .

Combining (3) with (2), we have

$$\begin{aligned} \Gamma_\gamma(t) &= \mathfrak{I}(x_{t-m\tau+1}, \dots, x_t, \dots, x_{t+p\tau}) = \mathfrak{I}(\Phi'(x_{t-m\tau+1}, x_{t-m\tau+2}, \dots, x_t)) \\ &= \tilde{\Phi}(x_{t-m\tau+1}, x_{t-m\tau+2}, \dots, x_t). \end{aligned} \quad (4)$$

Now a nonlinear mapping  $\tilde{\Phi}: \mathcal{R}^{m\tau} \rightarrow \mathcal{R}^1$  is established, which can be approximated through an ensemble neural network model described below.

Ensemble methods are learning algorithms that construct a set of classifiers and then classify new samples by taking a weighted vote of their predictions. Learning of continuous valued functions using neural network ensembles can give improved accuracy, reliable estimation of the generalization error, and active learning (Hansen & Salamon, 1990).

The 10 independent BP neural networks (also called members later in this paper) are comprised in our ensemble neural network model. Each member has different initial connection weights and different numbers of hidden neurons for diversity. Each member is trained independently using  $m\tau$  dimension training samples as input and corresponding turning indicator as output.

How to combine the outputs of each member, in test stage, into one output? Averaging of the outputs has been shown a powerful procedure which improves effectively on single network performance and its superiority over other methods, such as using a hierarchical voting structure with a tree of majority gates, has been demonstrated (Freund, 1995; Naftaly, 1997). In this paper, the output of ensemble model is produced through a linear combination method by weighted average, i.e.,

$$f_{ens}(\vec{x}) = \sum_{j=1}^N w_j f_j(\vec{x}), \quad (5)$$

where  $f_{ens}(\vec{x})$  is the combined output of the ensemble model,  $w_j$  the weight associated with the  $j$ th network's output, and  $f_j(\vec{x})$  the output of the  $j$ th network.

The weight  $w_j$ , the respective contribution of each member being taken account of, is trained through a parameter learning algorithm described later in Section 2.5.

The whole framework of the ensemble learning based TPP scheme is illustrated in Fig. 1.

### 2.4. Threshold optimization

When the trained ensemble neural network model is evaluated feeding with test data as input, we can get an integrated prediction output  $f_{ens}(\vec{x})$ , which is continuous and predictive of the occurrence of turning points. The higher value of  $f_{ens}(\vec{x})$ , the more likely current time step becomes a peak when the time series continues to evolve within  $p\tau$  time steps. Similarly a relatively low value of  $f_{ens}(\vec{x})$  votes for a trough. Given a set of thresholds,  $\tilde{\theta}$  for peak and  $\theta$  for trough, we can define an indicative function  $T(x_i)$  to decide whether a time step  $i$  could be forecasted as a turning point, i.e.,

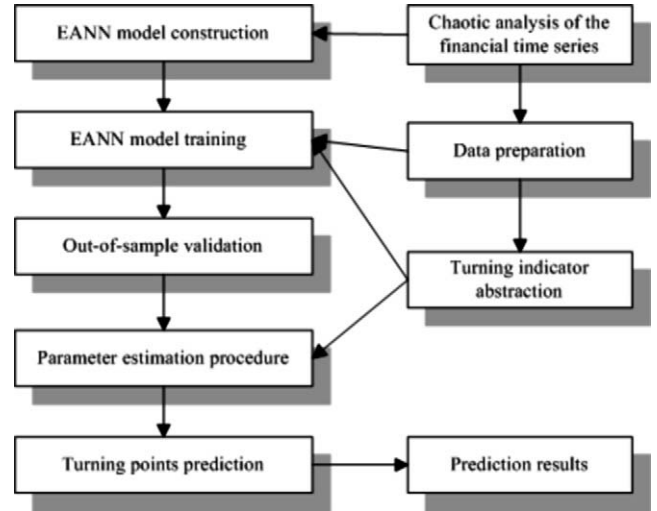


Fig. 1. The ensemble learning based TPP framework. EANN is the abbreviation of ensemble artificial neural network. Data preparation means the phase space reconstruction process, i.e., to produce the reconstructed time-delay vectors from the raw time series using the embedding dimension and reconstructive delay estimated in the chaotic analysis phase.

$$Tx_i = \begin{cases} 1, & \text{if } f_{ens}(x_i) \geq \tilde{\theta}, \\ -1, & \text{if } f_{ens}(x_i) \leq \theta, \\ 0, & \text{otherwise,} \end{cases} \quad (6)$$

where  $T(x_i) = 1$  indicates that the time step  $i$  is a peak,  $T(x_i) = -1$  a trough, and  $T(x_i) = 0$  means no turning point.

For the purpose of estimating proper values of the thresholds  $\tilde{\theta}$  and  $\theta$ , an out-of-sample validation procedure is conducted in our framework after the EANN model being well trained as shown in Fig. 1. In this stage, the EM-like parameters learning algorithm takes place as described in the next section, where the ensemble parameters learning and threshold estimation will be realized simultaneously.

In the parameters learning algorithm, a GA-based threshold optimization procedure is first explored in this section. It runs as a component in the parameters learning algorithm.

Given an out-of-sample validation dataset and the predictive output  $f_{ens}(\vec{x})$  from EANN model, the goal of the GA-based threshold optimization procedure is to search for the optimal thresholds  $\tilde{\theta}^*$  and  $\theta^*$ , with which the  $f_{ens}(\vec{x})$  would get a maximum prediction performance using (6).

To evaluate the performance of prediction, a problem-specific cost function is defined through comparing the ensemble model output  $f_{ens}(\vec{x})$  with the reference signal time series, and we compute the root mean square error, i.e.,

$$\text{TpMSE} = \left( \sum_{i=1}^N \beta_i (d(i) - y(i))^2 / N \right)^{1/2}, \quad (7)$$

where  $y(i)$  is the actual continuous output of the ensemble model, i.e., the  $i$ th component of one-dimensional vector  $f_{ens}(\vec{x})$ ,  $d(i)$  the reference signal time series, and  $\beta_i$  the discount factor.

The reference signal  $d(i)$  is constructed from the extracted turning points binary series and selected thresholds  $\tilde{\theta}$  and  $\theta$ , i.e.,

$$d(i) = \begin{cases} \tilde{\theta}, & \text{if } z_i^p = 1 \text{ and } y(i) < \tilde{\theta} \text{ or } z_i^p \neq 1 \text{ and } y(i) > \tilde{\theta}, \\ \theta, & \text{if } z_i^t \text{ and } y(i) > \theta \text{ or } z_i^t \neq 1 \text{ and } y(i) < \theta, \\ y(i), & \text{otherwise,} \end{cases}$$

$z_i^p$  and  $z_i^t$  are sequences of binary values indicative of the positions of turning points as described in Appendix:  $z_i^p = 1$  means a peak

occurred at time step  $i$  and  $z_i^T = 1$  means a trough at time step  $i$ . According to (7), each incorrect prediction  $y(i)$  will be punished in proportional to their error degree.

It is usually the case that the number of turning points in a given time series is much less than that of other points. In an even-weighted root mean square error, the prediction error produced on positions of true turning points often is deemphasized by errors on other less important positions. So a discount factor  $\beta_i$  is introduced in TpMSE in response to this problem, i.e.,

$$\beta_i = 1 + 3Z_{tp}(i) - \sum_{p=-s}^s \frac{Z_{tp}(i+p)^2}{|p|+2}, \quad (8)$$

where

$$Z_{tp}(i) = \begin{cases} 1, & \text{if } z_i^p = 1, \\ -1, & \text{if } z_i^T = 1, \\ 0, & \text{otherwise.} \end{cases}$$

As we can see in (8), the error produced on positions of true turning points is emphasized by biggest weights, while the points tightly adjacent to the true turning points have smallest weights. The reason behind this is, if a time step  $t$  very close to a true turning point, say one step ahead or behind of it, is predicted incorrectly as a turning point, this prediction result is still useful to us in real world and such points are more tolerable than an incorrectly predicted turning point far from a true one. Such manner encourages an incorrect prediction tightly adjacent to the real turning points over one with the same error value  $\varepsilon = d(i) - y(i)$  but distanced from the true turning points. Additionally, in (8),  $\beta_i$  increases asymptotically to 1 while the distance between time step  $i$  and the true turning points increases to  $s$ , where  $s = 2$  is chosen in our experimentally study.

The definition of TpMSE provides us with a method to search for the optimal thresholds, i.e.,

$$[\tilde{\theta}^*, \theta^*] = \arg \min_{\tilde{\theta}, \theta} \text{TpMSE}, \quad \tilde{\theta} \in [0.5, 1], \theta \in [0, 0.5].$$

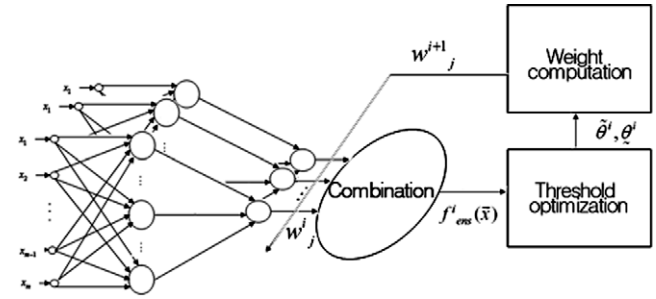
Thus, the problem of threshold optimization in validation stage becomes an optimization problem on two-dimensional space. In our work, a roulette wheel based genetic algorithm with elitism (Goldberg & David, 1989) is used for threshold optimization.

The thresholds  $\tilde{\theta}$  and  $\theta$  are two independent variables of a fitness function, namely the cost function TpMSE here. The ranges of two variables  $\tilde{\theta}$  and  $\theta$  are  $[0.5, 1]$  and  $[0, 0.5]$ , respectively. The population size is set to be 20. The most elite individual is maintained from generation to generation without changes.

### 2.5. Parameter learning algorithm

In our ensemble learning based TPP scheme, a parameter learning algorithm is needed to train the combination weights in (5).

The experimental data is divided into three parts, i.e. the training dataset, the validation dataset and the test dataset. After an ensemble neural network model is constructed and trained using training dataset, the out-of-sample validation process should be



**Fig. 2.** Schematic representation of the parameter learning algorithm. The ten feedforward neural networks are used, whose outputs are combined using different weights  $w_j^i$  to generate the output of the ensemble model.  $i$  means the  $i$ th iteration and  $j$  means the  $j$ th neural network.  $\tilde{\theta}^i$  and  $\theta^i$  indicate the estimated thresholds in the  $i$ th iteration.

conducted, where the parameters learning procedure takes place as shown in Fig. 1. The threshold optimization procedure introduced in Section 2.4 is also incorporated in this algorithm as a component. The ensemble parameters and thresholds estimated here will be used for ensemble model in the test stage.

Inspired by the expectation-maximization algorithm (EM) (Dempster, Laird, & Rubin, 1977), which is an excellent method widely used in parameter learning with incomplete data, we put forward an algorithm that realize ensemble parameter learning and threshold estimation simultaneously, as described in Table 1.

The combination weights  $w_j^i$  can be computed as

$$w_j^i = [Q_j^i - Q_{\min}^i] / \sum_{m=1}^N [Q_m^i - Q_{\min}^i], \quad Q_{\min}^i = \min_{p=1}^N (Q_p^i), \quad (9)$$

where  $Q_j^i$  is the evaluated prediction performance of the  $j$ th member in the  $i$ th iteration by means of TpMSE, and  $w_j^i$  the computed weight of the  $j$ th member in the  $i$ th iteration.

The schematic representation of parameter learning algorithm is shown in Fig. 2.

### 3. Trading strategy design

The TPP scheme presented in Section 2 provides an effective way to model and predict the trend of the financial time series. The indicative function  $T(x_i)$  given in (6) is used to transform the continuous output of the EANN model into discrete points 1, -1, 0 to indicate whether a corresponding time step in the time series is predicted as a peak, a trough or none. This process can also be considered to label the trading signal along with the evolvement of the financial time series. This section gives instructions on how the foreseen signal about the trend turning can be utilized in practical financial investments.

The proposed trading strategy, called TPP-based strategy in this paper, involves simply the turning points prediction results, and is substantially effective for investors to save time and energy consumed in monitoring financial data variations. It can be formulated as

**Table 1**

Parameter learning algorithm.

- Step 1. Train each member of the ensemble model independently using the training dataset.
- Step 2. Feed each member with the validation dataset as input and then get the outputs  $f_j^1(\bar{x})$ . Assign weights  $w_j^1 \in [0, 1]$  randomly. Compute  $f_{ens}^1(\bar{x})$  according to (5).
- Step 3. Search for optimal thresholds  $\tilde{\theta}^1, \theta^1$  through the GA-based threshold optimization procedure.
- Step 4. Evaluate performance of each member using  $\tilde{\theta}^1, \theta^1$  on the validation dataset. These performance evaluation results are used as confidence factors to compute weights  $w_j^2$  as described in (9).
- Step 5. Compute  $f_{ens}^2(\bar{x})$  using  $w_j^2$ . Find current optimal thresholds  $\tilde{\theta}^2, \theta^2$  through the threshold optimization procedure.

$f_j^i(\bar{x})$  is the individual output of a single neural network and  $f_{ens}^i(\bar{x})$  is the computed output of the ensemble model using combination weight  $w_j^i$ .  $i$  means the  $i$ th iteration and  $j$  means the  $j$ th member in ensemble model.  $\tilde{\theta}^i$  and  $\theta^i$  indicate the estimated thresholds in the  $i$ th iteration.



$$Sig(t) = \begin{cases} \text{Buy,} & \text{if } T(x_t) = -1 \text{ and } Stat(t-1) = 0, \\ \text{Sell,} & \text{if } T(x_t) = 1 \text{ and } Stat(t-1) = 1, \\ \text{Keep,} & \text{otherwise,} \end{cases}$$

$$Stat(t) = \begin{cases} 0, & \text{if } t = 0, \\ (Stat(t-1) + 1) \bmod 2, & \text{if } Sig(t) = \text{Sell} \\ Stat(t-1), & \text{if } Sig(t) = \text{Buy or Keep,} \end{cases}$$

$Sig(t)$  denotes the trading instruction produced in this strategy designing process, while  $Stat(t)$  records the current trading position of the investor. We do not consider selling short here because short-selling mechanism is not universal in all kinds of financial markets. According to this strategy, investors buy in when they get the first trough signal and their current trading position is empty, sell when they are in holding position and receive the first peak signal.

#### 4. Experimental results

To evaluate the performance of the TPP-based trading strategy designing approach, the two representative real-world financial time series in stock market were investigated, i.e., a stock quote time series of TESCO PLC and the Dow Jones Industrial Average (DJIA) index time series. The former is a FTSE international ingredient stock and is employed to validate the effectiveness of the introduced method for individual stock price movement prediction while the latter is used as a demonstration of overall market trend judgment. This demonstration is not lack of generality because the basic laws of those financial time series of different kinds, such as of bonds and commodities trade, are essentially similar.

The datasets we used are shown in Fig. 3.

For the TESCO dataset, the chaotic analysis process was conducted on time series segment from 18 November 2004 to 31 May 2006. According to the Cao method, the value of  $E1$  for this stock quote time series was calculated with the reconstructive de-

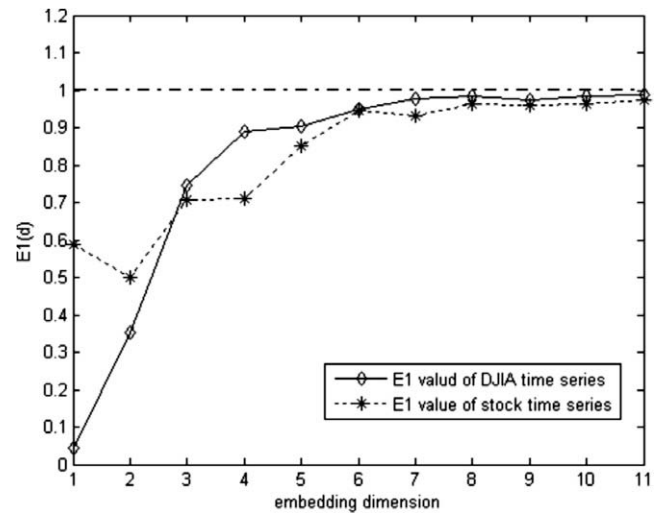


Fig. 4. The computation of embedding dimension using the Cao method.  $E1(d)$  is a quantity defined in Cao method to investigate the average variation between distances of nearest neighbors in the reconstructed phase space when the embedding dimension increases from  $d$  to  $d+1$ .  $E1(d)$  stops changing when  $d$  is greater than some value  $d_0$  if the time series comes from an attractor. Then  $d_0 + 1$  is the minimum embedding dimension we look for.

lay  $\tau = 1$ . A minimum embedding dimension of eight was selected as shown in Fig. 4.

A reconstructed phase space  $V^8$  consequently was built with the embedding dimension  $m = 8$  and the reconstructive delay  $\tau = 1$ . We calculated the average divergence degree of the reconstructed dynamic system, as illustrated in Fig. 5. Thus the maximum Lyapunov exponent  $\lambda_1 = 0.1284$  and the Lyapunov time  $t_0 = 7.7881$  were derived from the numerical computation method for Lyapunov exponent (Rosenstein et al., 1993).

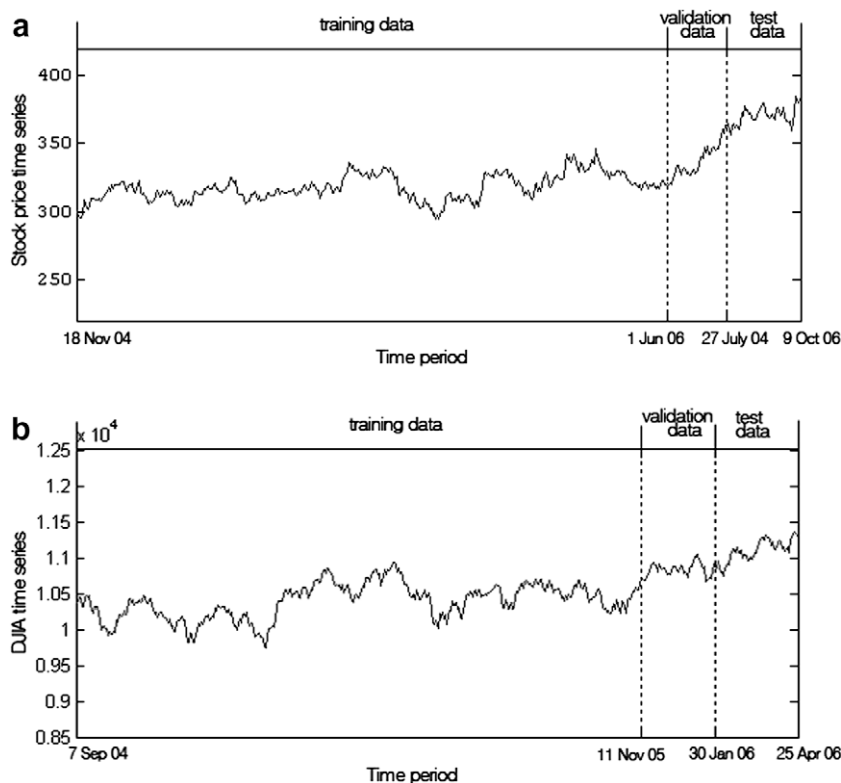
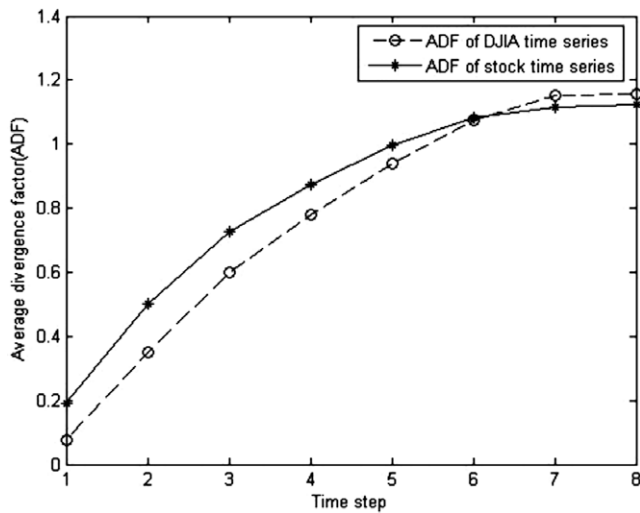


Fig. 3. Experimental time series datasets and their divisions. (a) An individual stock time series dataset and (b) a Dow Jones Industrial Average (DJIA) time series dataset.



**Fig. 5.** The average divergence factors. ADF is used in the numerical computation method to quantitatively measure the average divergence degree of adjacent trajectories of the dynamic system over time in the reconstructed phase space.

Similar chaotic analysis was then conducted on Dow Jones Industrial Average (DJIA) time series between 7 September 2004

and 10 November 2005. We obtained the results of  $\tau = 1$ ,  $m = 8$ ,  $\lambda_1 = 0.1564$ , and  $t_0 = 6.3938$ .

In both cases, the maximum Lyapunov exponents are positive, i.e.,  $\lambda_1 = 0.1284 > 0$  and  $\lambda_1 = 0.1564 > 0$ , which testify the chaotic characteristic of the given time series. Thus, the proposed method can be applied to these two financial time series.

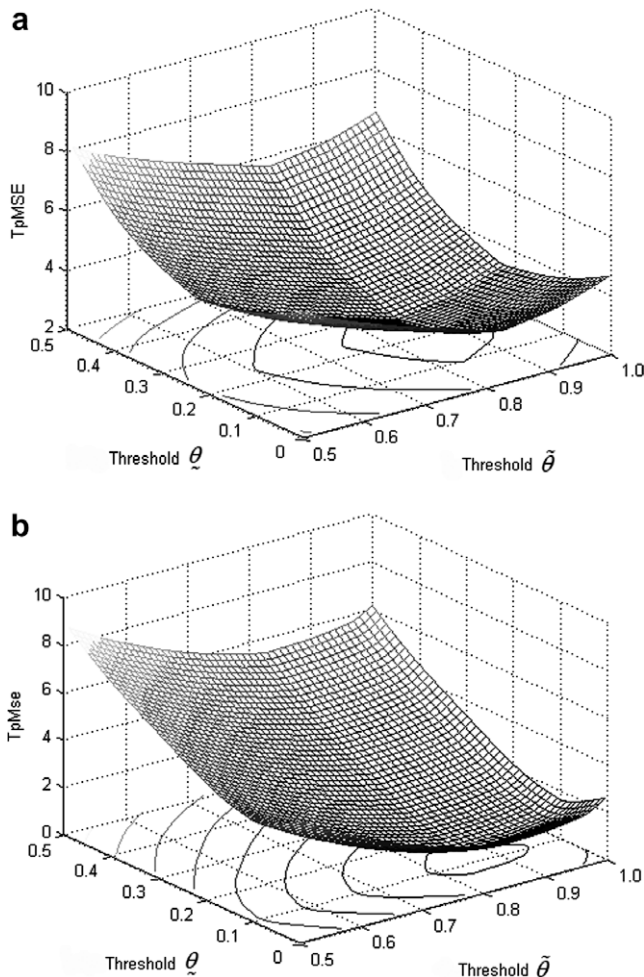
Based on the analysis above, the experimental samples can be constructed from original time series through delayed coordinate embedding with the embedding dimension  $m = 8$  and the reconstructive delay  $\tau = 1$ . The ensemble models were trained on training datasets and then interrogated using validation samples. The initial weights  $w_j^1$  were assigned randomly and the error surfaces of the ensemble model output are shown in Fig. 6.

**Table 2**

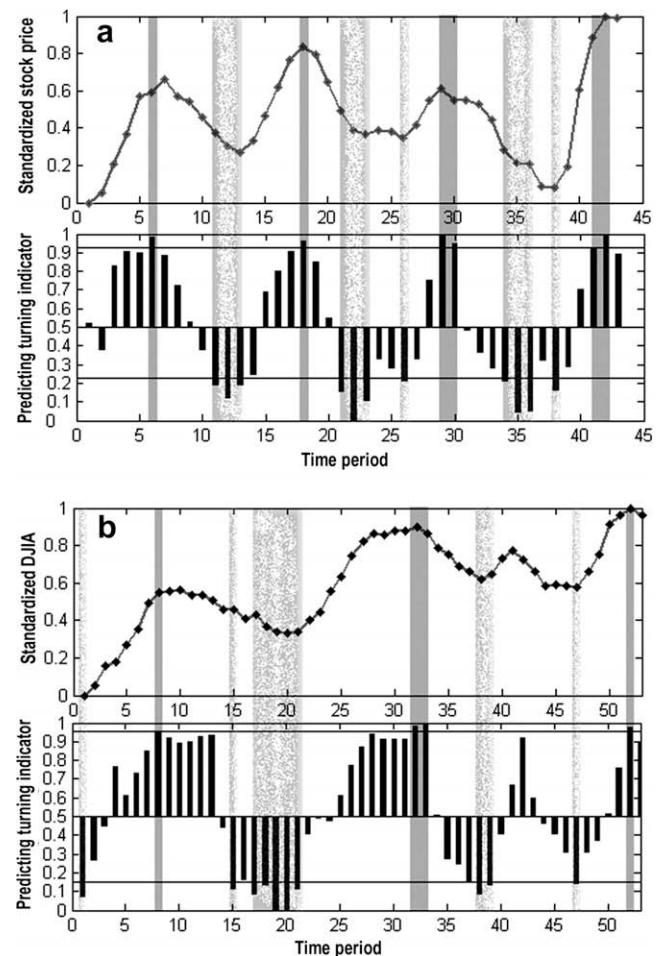
Estimated probability thresholds.

	Experimental dataset	$\tilde{\theta}$	$\theta$	Optimal TpMSE
1st iteration	TESCO	0.81765	0.24754	2.5177
	DJIA	0.90227	0.13034	0.9713
2nd iteration	TESCO	0.91984	0.22876	2.3262
	DJIA	0.95485	0.15054	0.7312

$\tilde{\theta}$  is the peak threshold and  $\theta$  is the trough threshold. Optimal TpMSE is the corresponding values of cost function in different  $\tilde{\theta}$  and  $\theta$ .



**Fig. 6.** The error surfaces of the ensemble model output. The error surface is drawn observing TpMSE evolving alongside the variation of both the trough threshold  $\theta$  and the peak threshold  $\tilde{\theta}$ . (a) For the stock time series dataset and (b) for the Dow Jones Industrial Average (DJIA) time series dataset.



**Fig. 7.** Prediction results on test datasets. The standardized test time series is shown in the upper subfigure and the predicting turning indicator is shown in the lower. When the predicting turning indicator exceeds the thresholds visualized by the two horizontal lines in the lower subfigure, the corresponding time period is predicted as turning points, as highlighted by shadow bars above (dark gray for peaks and light gray for troughs). (a) For the stock time series dataset and (b) for the Dow Jones Industrial Average (DJIA) time series dataset.

**Table 3**

Trading process of the stock time series dataset.

Trading number	Trading time	Buy price	Sell price
1	2006-8-9	361.0	0.0
2	2006-8-16	0.0	374.5
3	2006-8-23	367.25	0.0
4	2006-9-4	0.0	380.0
5	2006-9-7	370.5	0.0
6	2006-9-19	0.0	371.25
7	2006-9-26	367.0	0.0
8	2006-10-5	0.0	382.5
End time	2006-10-9	0.0	384.0

**Table 4**

Trading process of the DJIA time series dataset.

Trading number	Trading time	Buy price	Sell price
1	2006-2-8	10858.62	0.0
2	2006-2-17	0.0	11115.32
3	2006-3-1	11053.53	0.0
4	2006-3-24	0.0	11279.97
5	2006-4-3	11144.94	0.0
6	2006-4-24	0.0	11336.32
End time	2006-4-25	0.0	11283.25

**Table 5**

Classification of turning points prediction errors.

Prediction results	Actual outcome		
	TP	Non-TP	Marginal
TP	$O_{ii}$	$O_{ij}$	$O_{i.}$
Non TP	$O_{ji}$	$O_{jj}$	$O_{.j}$
Marginal	$O_{.i}$	$O_{.j}$	$O$

The numbers in the table are counted as happened according to our prediction result and the reference. TP stands for turning points.

Starting with this set of initial parameters, the ensemble parameter learning and the threshold estimation algorithm described in Section 2.5 were done. The probability thresholds are estimated as listed in Table 2.

After the validation and parameter learning process, we examined the well-trained ensemble model on the test datasets. The prediction results are given in Fig. 7.

We can then design, according to the method introduced in Section 3, a trading strategy for instance on test datasets of both TESCO and DJIA. The trading process is illustrated in Tables 3 and 4, respectively.

## 5. Performance evaluations

In this section, we conduct a detailed performance evaluation on the experimental results from the TPP model and the proposed trading strategy designing approach. A variety of performance metrics are adopted to quantify the forecast accuracy of TPP model and the profitability of generated trading strategies.

### 5.1. Evaluation of the TPP model

The quality of out-of-sample forecast in the field of time series prediction is typically assessed by such measures as mean squared error (MSE), mean absolute error (MAE) or the statistics test (Diebold & Mariano, 1995). Nevertheless, in our research of turning points prediction, because the prediction results are sequences of indicative symbols over a time period, other measures will be employed to evaluate the prediction performance. Besides the evaluation measure of root mean square error TpMSE defined in Section 2.4 which plays a key role in the threshold optimization and ensemble parameter learning, we also perform such tests as described in (Fritzsche, Kuzin, Berlin, & Frankfurt, 2005). The forecasting results are thus classified in Table 5.

The information content of the forecast can be summarized as  $I = \frac{O_{ii}}{O_{ii}+O_{ji}} + \frac{O_{jj}}{O_{ji}+O_{jj}}$ . The value of  $I$  should asymptotically be bound between 1 and 2. In a “coin flip” case, we have  $O_{ii} \approx O_{ji}$ ,  $O_{jj} \approx O_{ij}$ , and thus  $I \rightarrow 1$ . If the forecast is “perfect”, we should have  $O_{ji} = O_{ij} = 0$  and  $I = 2$ . Thus, any value of  $1 < I \leq 2$  indicates a positive information content (compared to the “coin flip”).

The statistical significance of the information content can be systematically tested. The consistent estimator for the cell counts is given by  $\hat{E}_{ij} = O_{i.}O_{.j}/O$ . According to Pearson's  $\chi^2$  test, a measure  $C$  is constructed as

$$C = \sum_{i=1}^2 \sum_{j=1}^2 \frac{(O_{ij} - \hat{E}_{ij})^2}{\hat{E}_{ij}} \sim \chi^2(1).$$

The null hypothesis that the turning points forecast has no value is that the forecast and reality are independent. This measures the quadratic distance between real and expected values in relation to the expected probabilities. We then calculate the  $p$ -value of the test supposing that both series are independent.

The relation between the forecast and reality can be evaluated by the contingency coefficient proposed by Pearson. This contingency coefficient is a normalization of reported  $\chi^2$  statistic which is given by  $\sqrt{\frac{\min(i,j)}{\min(i,j)-1} \frac{C}{C+O}}$ . It is bound between 0 and 1 where a higher value indicates stronger association. We also report the Yule coefficient which measures the association between concordant and discordant pairs of attributes. The Yule coefficient ( $Y$ ) is given by  $Y = \frac{O_{ii}O_{jj}-O_{ij}O_{ji}}{O_{ii}O_{jj}+O_{ij}O_{ji}}$  and bounded between  $-1$  (negative association) and 1 (positive association).

The performance evaluations on the prediction output of TPP model are listed in Table 6.

### 5.2. Evaluation of trading strategy designing approach

One way to evaluate the performance of our proposed trading strategy designing approach is to look at the trading profits it generates. Meanwhile, the typical performance estimators of a trading strategy are total profit and rate of return (Li & Kuo, 2008; Povinelli, 2001), which are defined as follows.

Total profit (TP) represents the profitability of total trades. TP can be negative when the loss is greater than the gain. The higher the TP is, the better the performance is.

$$TP = G - L, \quad (10)$$

**Table 6**

Prediction performance of TPP model on experimental financial time series.

Experimental dataset	TpMSE	$I$	Pearson's $\chi^2$	$p$ -Value	Contingency coefficient	Yule coefficient
Stock time series	2.0462	1.4643	4.1849	0.04079	0.4212	0.7647
DJIA time series	1.6822	1.5076	7.8373	0.00512	0.5076	0.8272

**Table 7**

Profits analysis on the trading strategy.

Trading strategy	Stock time series		DJIA time series	
	Buy-and-hold	TPP-based strategy	Buy-and-hold	TPP-based strategy
Total profit (TP)	23	42	424.63	674.54
Rate of return (RR)	6.37%	11.63%	3.91%	6.21%

where  $G$  is the gross gain and  $L$  the gross loss. One caveat to this is that it ignores trading costs.

Rate of return (RR) is the net profit expressed as a percentage of average capital employed. The higher the RR is, the better the performance is.

$$RR = TP/INV * 100, \quad (11)$$

where INV is the capital employed in an investment.

In Table 7, the trading profits are calculated and compared with another popular trading strategy, i.e., buy-and-hold, where the stock is bought at start time and is held until end time of the investment period.

## 6. Conclusions

In this paper, we propose a novel trading strategy designing approach as a decision making supporting tool for financial investors. It is constructed on the ground of a turning points prediction scheme. The TPP scheme is guaranteed by a newly discovered nonlinear mapping which harnesses the hidden chaotic dynamics intrinsic in financial time series. This nonlinear mapping is then elaborated to one that could be modeled by EANN ensemble learning method, through an event characterization function which provides a tool to characterize the trend pattern in the financial time series. The EANN model using 10 artificial neural networks is employed in this machine learning solution and a parameter learning algorithm is given afterwards.

The TPP-based trading strategy designing approach consists of multiple phases including chaotic analysis, turning indicator extraction, ensemble model training, parameter learning, and trading strategy generation. The chaotic analysis is particularly important for unveiling the chaotic properties of the target financial time series, and further giving instructions on both the structure of the EANN model and the computation span of the turning indicator.

The proposed approach is applied to two real-world non-stationary financial time series, whose chaotic characteristics are proven by means of the Lyapunov exponent. Our experimental results of both individual stock investment and overall market trend judgment show how the proposed trading strategy designing approach can be effective in real world. Investors, short-term or long-term, fan of fundamental or technical analysis, can take great advantage of the proposed method in this paper making their own guess on how the market behaves and when there may be a turning in trend.

The generated trading strategy hereby solely depends on the turning points prediction results, intentionally avoiding the involvement of any experience knowledge which nevertheless may be important to making optimal decision. This helps to show more clearly the effectiveness of the TPP scheme in the trading strategy designing process. The experimental results are encouraging and we hope that our method can provide financial investors with a useful tool and give referential information on the big picture of economy for policy-makers and regulators.

In future works, a multi-classifier scheme that incorporates neural network, K-nearest neighbor, and support vector machine (SVM) could be put in place in order to improve performance and reliability of the existing scheme. In addition, more factors could be taken into consideration, such as the volume vibration patterns and valuable priori knowledge from expertise chartists,

to further advance the trading strategy designing approach presented in this paper.

## Acknowledgements

This work is supported in part by the National Science Foundation of China (NSFC) under Grant Nos. 60621062 and 60775040.

## Appendix. Computation method for the turning indicator

The proposed event characterization function of turning indicator is used to quantitatively measure the event occurrence degree of a turning point considering the nearby values. The computation of the turning indicator is based on the extraction of turning points from the time series according to Definition 1 in Section 2.2.

Let  $x_t \in \mathbb{R}$ ,  $t = 1, 2, \dots, n$  be a given time series, and let  $z_t^p$  and  $z_t^T$  be sequences of binary variables indicating positions of turning points in  $x_t$ .

Let  $\varepsilon_t = x_t - x_{t-1}$ ,  $z_t^p$  and  $z_t^T$  can be preliminarily calculated from time series  $x_t$  as follows

$$z_t^p = \begin{cases} 1, & \text{if } \varepsilon_t > 0 \wedge \varepsilon_{t+1} \leq 0 \wedge \min\{\delta_t(d) | d \leq p\tau\} \leq -\gamma, \\ 0, & \text{otherwise} \end{cases}$$

and

$$z_t^T = \begin{cases} 1, & \text{if } \varepsilon_t < 0 \wedge \varepsilon_{t+1} \geq 0 \wedge \max\{\delta_t(d) | d \leq p\tau\} \geq \gamma, \\ 0, & \text{otherwise} \end{cases}$$

where  $\delta_t(d) = \sum_{u=t}^{t+d} \varepsilon_u / x_t$ ,  $t = 1, 2, \dots, n-d$ ,  $p$  is the Lyapunov time,  $\tau$  the reconstructive delay, and  $\gamma$  is a given percentage of change in observed values of time series  $x_t$ , which depends on different return expectation in different trading strategy design. The chosen variable  $\gamma$  is application dependant, and thus reflects the preference of users, for example the need of a bold investor to implement an aggressive action plan. In our study  $\gamma = 0.10$  is used to more emphasize on short-term investment.

There are still some points which satisfy the conditions above but are pseudo turning points that should be screened out. They are those turning points located on the not-yet-finished upward or downward side of the time series.

We define two functions for locating the nearest peak and nearest trough from  $m\tau - 1$  time steps before current time  $t$  to  $t$ , i.e.,

$$h^p(t) = \{u | z_u^p = 1 \wedge z_k^p = 0 \text{ for } u < k < t \wedge t - u < m\tau - 1 \text{ or } t - u = m\tau - 1\},$$

$$h^T(t) = \{u | z_u^T = 1 \wedge z_k^T = 0 \text{ for } u < k < t \wedge t - u < m\tau - 1 \text{ or } t - u = m\tau - 1\},$$

where  $m$  is the embedding dimension and  $\tau$  the reconstructive delay. Then these pseudo turning points can be excluded as follows:

$$\begin{cases} z_t^p = 0, & \text{if } z_t^p = 1 \wedge h^p(t) \geq h^T(t) \wedge x_{h^p(t)} \geq x_t, \\ z_{h^p(t)}^p = 0, & \text{if } z_t^p = 1 \wedge h^p(t) \geq h^T(t) \wedge x_{h^p(t)} < x_t \end{cases}$$

and

$$\begin{cases} z_t^T = 0, & \text{if } z_t^T = 1 \wedge h^T(t) \geq h^p(t) \wedge x_{h^T(t)} \leq x_t, \\ z_{h^T(t)}^T = 0, & \text{if } z_t^T = 1 \wedge h^T(t) \geq h^p(t) \wedge x_{h^T(t)} > x_t. \end{cases}$$



Now turning indicator  $\Gamma_{\gamma}(t)$  of each point in the time series  $x_t$  can be calculated as

$$\Gamma_{\gamma}(t) = \begin{cases} 1, & \text{if } z_t^P = 1, \\ 0, & \text{if } z_t^T = 1, \\ \frac{x_t - x_{T^*(t)}}{x_{P^*(t)} - x_{T^*(t)}}, & \text{otherwise,} \end{cases}$$

where  $P^*(t)$ ,  $T^*(t)$ , derived from  $h^P(t)$ ,  $h^T(t)$ , respectively, indicate the positions of nearest peak and nearest trough located in opposite side of time step  $t$ .

## References

- Camacho, M., & Quiros, G. P. (2002). This is what the leading indicators lead. *Journal of Applied Econometrics*, 17, 61–80.
- Cao, L. (1997). Practical method for determining the minimum embedding dimension of a scalar time series. *Physica D*, 110, 43–50.
- Catherine, K., Walter, C. L., & Michel, T. (2004). Noisy chaotic dynamics in commodity markets. *Empirical Economics*, 29, 489–502.
- Chen, F., Ji, G. R., Zhao, W. C., & Nian, R. (2005). The prediction of the financial time series based on correlation dimension. *Lecture Notes in Computer Science*, 3610, 1256–1265.
- Chun, S. H., & Park, Y. J. (2005). Dynamic adaptive ensemble case-based reasoning: Application to stock market prediction. *Expert Systems with Applications*, 28, 435–443.
- Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum-likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39, 1–38.
- Diebold, F. X., & Mariano, R. S. (1995). Comparing predictive accuracy. *Journal of Business and Economic Statistics*, 13, 253–263.
- Freund, Y. (1995). Boosting a weak learning algorithm by majority. *Information and Computation*, 121(2), 256–285.
- Fritsche, U., Kuzin Berlin & Frankfurt, V. (2005). Prediction of business cycle turning points in Germany. *Jahrbücher für Nationalökonomie und Statistik*, 225(1), 22–43.
- Goldberg & David, E. (1989). *Genetic algorithms in search, optimization and machine learning*. Boston: Addison-Wesley.
- Grassberger, P., & Procaccia, I. (1983). Measuring the strangeness of strange attractors. *Physica D*, 9, 189–208.
- Hansen, L., & Salamon, T. (1990). Neural networks ensemble. *IEEE Trans on Pattern Analysis and Machine Intelligence*, 12(10), 993–1001.
- Harrison, R. G., Yu, D., Oxley, L., Lu, W., & George, D. (1999). Non-linear noise reduction and detecting chaos: Some evidence from the S&P composite price index. *Mathematics and Computers in Simulation*, 48, 407–502.
- Hsieh, D. (1989). Testing for nonlinear dependence in daily foreign exchange rates. *Journal Business*, 62(3), 339–359.
- Huang, R. S. (2000). *Chaos and Its Application*. Wuhan: Wuhan University Press. pp. 115–135.
- Jaeger, H., & Haas, H. (2004). Harnessing nonlinearity: predicting chaotic systems and saving energy in wireless Communication. *Science*, 304, 78–80.
- Kling, J. L. (1987). Predicting the turning points of business and economic time series. *Journal of Business*, 60(2), 201–238.
- Kodogiannis, V., & Lolis, A. (2002). Forecasting financial time series using neural network and fuzzy system-based techniques. *Neural Computing and Application*, 11, 90–102.
- Li, S. T., & Kuo, S. C. (2008). Knowledge discovery in financial investment for forecasting and trading strategy through wavelet-based SOM networks. *Expert System with Applications*, 34, 935–951.
- Li, X. Q., & Deng, Z. D. (2007). A machine learning approach to predict turning points for chaotic financial time series. *The 19th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 07)*, 331–335.
- Liu, J. Q., Zhang, T. Q., & Yu, S. K. (2004). Chaotic phenomenon and the maximum predictable time scale of observation series of urban hourly water consumption. *Journal of Zhejiang University Science*, 5(9), 1053–1059.
- Naftaly, U. (1997). Optimal ensemble averaging of neural networks. *Network: Computation in Neural System*, 8, 283–296.
- Poddig, T., & Huber, C. (1999). A comparison of model selection procedures for predicting turning points in financial time series. *PKDD'99, LNAI, 1704*, 492–497.
- Povinelli, R. J. (2001). Identifying temporal patterns for characterization and prediction of financial time series events. *International Workshop on Temporal, Spatial, and Spatio-temporal Data Mining (TSDM, 2000)*, 46–61.
- Rosenstein, M. T., Collins, J. J., & De luca, C. J. (1993). A practical method for calculating largest Lyapunov exponents from small data sets. *Physica D*, 65, 117–134.
- Skabar, A. (2005). Application of bayesian techniques for MLPs to financial time series forecasting. *AI2005, LNAI, 3809*, 888–891.
- Sun, Y. F., Liang, Y. C., Zhang, W. L., Lee, H. P., Lin, W. Z., & Cao, L. J. (2005). Optimal partition algorithm of the RBF neural network and its application to financial time series forecasting. *Neural Computing and Application*, 14, 36–44.
- Takens, F. (1981). Detecting strange attractors in turbulence. *Lecture Notes in Mathematics*, 898, 361–381.
- Wecker, F. W. (1979). Predicting the turning points of a time series. *Journal of Business*, 52, 35–50.