Available online at www.sciencedirect.com
SCIENCE DIRECT®
Physica A 363 (2006) 481–491
PHYSICA A
www.elsevier.com/locate/physa

The application of neural networks to forecast fuzzy time series

Kunhuang Huarng, Tiffany Hui-Kuang Yu*

Department of International Trade, Feng Chia University, 100 Wenhwa Rd., Seatwen, Taichung 407, Taiwan, ROC
Department of Public Finance, Feng Chia University, 100 Wenhwa Rd., Seatwen, Taichung 407, Taiwan, ROC

Received in revised form 21 March 2005
Available online 6 September 2005

Abstract

Fuzzy time series models have been applied to handle nonlinear problems. To forecast fuzzy time series, this study applies a backpropagation neural network because of its nonlinear structures. We propose two models: a basic model using a neural network approach to forecast all of the observations, and a hybrid model consisting of a neural network approach to forecast the known patterns as well as a simple method to forecast the unknown patterns. The stock index in Taiwan for the years 1991–2003 is chosen as the forecasting target. The empirical results show that the hybrid model outperforms both the basic and a conventional fuzzy time series models.

Keywords: Backpropagation; Forecasting; Nonlinear; Stock index

1. Introduction

Fuzzy time series models have been applied to handle nonlinear problems, such as enrollment [1–5], stock index forecasting [6–10], and temperature [11], etc. A fuzzy time series essentially consists of steps such as fuzzification, the establishment of fuzzy relationships, and defuzzification. Some studies have focused on the fuzzification alone [7,9,12,13]. Nevertheless, many studies have focused on the establishment of fuzzy relationships [1–6,8,10,14,15] because it is directly relevant to forecasting. Therefore, this study also targets the establishment of fuzzy relationships.

Artificial neural network approaches (hereafter referred to more simply as neural networks) have been successfully applied to various applications [16–18]. The nonlinear structures of neural networks have, for instance, been very useful in forecasting [19]. In order to take advantage of its nonlinear capabilities [20], this study has chosen a neural network to establish fuzzy relationships in fuzzy time series, which are also nonlinear. Hence, the rationale and motivation for applying a neural network are self-explanatory.

We have followed the suggestions regarding constructing a neural network for forecasting [21], including data preparation, the network setup, and model selection and evaluation. To investigate the forecasting capabilities of the neural network approach, we propose two models, a basic model and a hybrid model. The basic model simply uses a neural network approach to forecast all of the observations, while the hybrid model

*Corresponding author. Tel.: +886 4 24517250 ext. 4309.
E-mail addresses: khhuarng@fcu.edu.tw (K. Huarng), hkyu@fcu.edu.tw (T.H.-K. Yu).

0378-4371/$ - see front matter
doi:10.1016/j.physa.2005.08.014

used the same neural network approach to forecast the known patterns as well as a simple method to forecast the unknown patterns. The Taiwan Stock Exchange Capitalization Weighted Stock Index (TAIEX) for the years 1991–2003 is chosen as the forecasting target. The forecasting results of both models are then compared with a fuzzy time series model. To show these things, the remainder of this paper is organized as follows. Section 2 briefly reviews fuzzy time series. Section 3 describes the setup for the two models. Section 4 explains the forecasting process. Section 5 compares the empirical results from using the different models. Finally, Section 6 concludes the paper.

2. Fuzzy time series

Let U be the universe of discourse, where $U = \{u_1, u_2, \ldots, u_b\}$. A fuzzy set A_i of U is defined as $A_i = f_{Ai}(u_1)/u_1 + f_{Ai}(u_2)/u_2 + \cdots + f_{Ai}(u_b)/u_b$, where f_{Ai} is the membership function of the fuzzy set A_i; f_{Ai}: $U \to [0, 1]$. u_a is an element of fuzzy set A_i; $f_{Ai}(u_a)$ is the degree of belongingness of u_a to A_i; $f_{Ai}(u_a) \in [0,1]$ and $1 \leqslant a \leqslant b$.

Definition 1. $Y(t)$ $(t = \ldots, 0, 1, 2, \ldots)$ is a subset of a real number. Let $Y(t)$ be the universe of discourse defined by the fuzzy set $f_i(t)$. If $F(t)$ consists of $f_i(t)(i = 1, 2, \ldots)$, $F(t)$ is defined as a fuzzy time series on $Y(t)$ $(t = \ldots, 0, 1, 2, \ldots)$ [22].

Following Definition 1, fuzzy relationships between two consecutive observations can be defined as follows:

Definition 2. If there exists a fuzzy relationship $R(t-1, t)$, such that $F(t) = F(t-1) \times R(t-1, t)$, where $\times$ represents an operation, then $F(t)$ is said to be caused by $F(t-1)$.

Definition 3. Let $F(t-1) = A_i$ and $F(t) = A_j$. The relationship between two consecutive observations, $F(t)$ and $F(t-1)$, referred to as a fuzzy logical relationship (FLR) [4], can be denoted by $A_i \to A_j$, where A_i is called the left-hand side (LHS) and A_j the right-hand side (RHS) of the FLR.

Song and Chissom proposed a fuzzy time series model, that included the following steps [4]: (1) define and partition the universe of discourse; (2) define fuzzy sets for the observations; (3) fuzzify the observations; (4) establish the fuzzy relationship, R; (5) forecast; and (6) defuzzify the forecasting results. It is their study that laid the foundation for the consecutive models.

Among these steps, the establishment of fuzzy relationships directly affects the forecasting results; hence, this has become the target of many relevant studies. Some of them are introduced as follows. In Song and Chissom's model [4], the relationship of each FLR was calculated by the min operator of its transposed LHS and RHS. Then, the overall fuzzy relationship was calculated by the union operator. Subsequently, Song and Chissom proposed another model [5], where a window base was chosen first. The relationship of each FLR was calculated using the Cartesian product of the fuzzy sets in the window base.

Sullivan and Woodall proposed a Markov model based on the estimated probability distribution functions of the transitions of observations [15]. In that model, the relationship of each FLR was calculated by means of the probability density functions of its transposed LHS and RHS using the matrix multiplication operator. The relationship was then added using the matrix addition operator. Chen proposed a model based on arithmetic operations [1]. The fuzzy relationship was established by putting the same LHS of the FLRs together into fuzzy logical relationship groups (FLRGs). For example, there are FLRs with the same LHSs (A_i): $A_i \to A_{j1}$, $A_i \to A_{j2}$, ... These FLRs can be grouped into an FLRG as $A_i \to A_{j1}, A_{j2}, \ldots$ Hwang et al. proposed a model based on the variations of observations [3]. In that model, a window size was set. A variation matrix was formed by $F(t-2) \ldots F(t-w+1)$. Then, the overall relationship was calculated using the variation matrix and $F(t-1)$ using the matrix multiplication operator.

3. Construction of the forecasting models

The setup of the forecasting model consists of the following steps [21], namely, data preparation, the neural network setup (input variable selection, the choice of structure, the transfer function, etc.), and evaluation and selection.

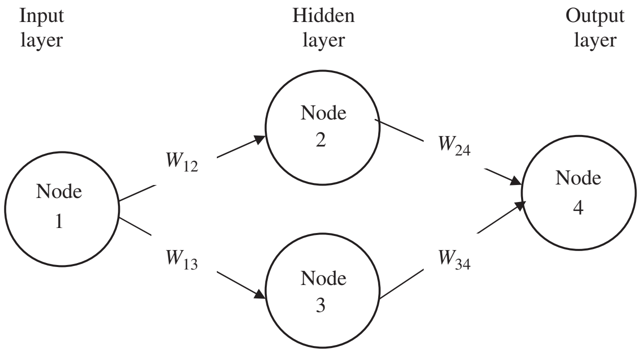

Fig. 1. Neural network structure.

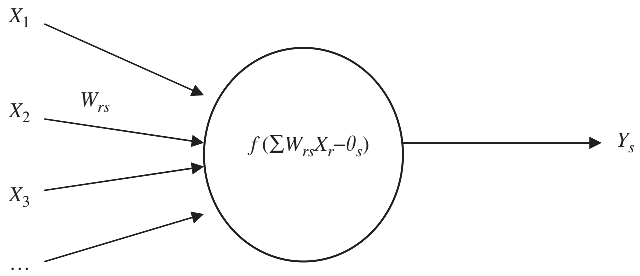

Fig. 2. A node in a neural network.

3.1. Data preparation

A neural network needs a large amount of data for training. Many studies have used a convenient ratio to separate in-samples from out-of-samples ranging from 70%:30% to 90%:10% [21]. Hence, we chose the TAIEX as the forecasting target. Meanwhile, we selected the data from January to October for our estimation (the in-sample) and November and December for forecasting (the out-of-sample). In other words, the ratio is about $\frac{10}{12} : \frac{2}{12} = 83\% : 17\%$, a ratio lying in between the two types of sample.

3.2. Neural network setup

A multilayer feedforward structure is the model most widely applied in forecasting [19]. Hence, backpropagation [23] was chosen as the model; the institutional version of PC Neuron,[1] backpropagation software, was chosen as the tool for developing the forecasting model.

In the setup, we intended to establish (or train) the fuzzy relationships of all the FLRs and then to forecast. As in Definition 3, an FLR is a 1–1 relationship. Hence, there is one input layer and one output layer with one node each. Although there are different guidelines for choosing the number of hidden layers and the number of hidden nodes in each layer, most forecasting applications have used only one hidden layer and a sufficiently large number of hidden nodes [24–28]. Meanwhile, to prevent over-fitting, a small neural network was preferred [29]. Accordingly, we used one hidden layer and two hidden nodes. Hence, we set up a neural network structure as in Fig. 1.

The function of each node in the hidden and output layers is depicted in Fig. 2. There is (are) input(s) to the node s from the node(s) r of the previous layer, such as X_r in the figure. Each connection from node r to s is

[1] http://www.chu.edu.tw/~icyeh/.

associated with a weight, W_{rs}, representing the connection strength in between. The output of node s, Y_s, is computed as follows [30].

$$Y_s = f\left(\sum W_{rs} \times X_r - \theta_r\right), \tag{1}$$

$$f(z) = \frac{1}{1 + e^{-z}}, \tag{2}$$

where $f(z)$ is a sigmoid function.

3.3. Model selection and evaluation

A relevant study has suggested that no single approach has always been the best for all kinds of applications [31,32], including neural networks [21]. Thus, in addition to a basic model, we also proposed a hybrid model. As mentioned in Section 3.1, all the observations were categorized as being either in-sample or out-of-sample. In the latter, the observations were further divided into known and unknown patterns. The observations in the out-of-sample that appeared in the in-sample were referred to as known patterns, while the others were referred to as unknown patterns.

The forecasting results from these two models were then compared with those from a fuzzy time series model, Chen' s model, which was easy to implement and performed better than those of many studies [1]. For evaluation purposes, the forecast error and root mean squared error (RMSE) were used to measure performance:

$$error_t = \left|actual_t - forecast_t\right|, \tag{3}$$

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{t=1}^{c} error_t^2}{c}}, \tag{4}$$

where there are c forecasts.

4. Neural network-based fuzzy time series models

This study applied a backpropagation neural network to forecast fuzzy time series. The TAIEX for the year 2000 was used for purposes of illustration.

Step 1: Defining and partitioning the universe of discourse.

According to the problem domain, the universe of discourse for observations, $U = [starting, ending]$, is defined. After the length of intervals, l, is determined, the U can be partitioned into equal-length intervals u_1, u_2, u_3, $\ldots, u_b$, $b = 1, \ldots$ and their corresponding midpoints m_1, m_2, m_3, $\ldots, m_b$, respectively.

$$u_b = [starting + (b-1) \times l_a, \quad starting + b \times l_a], \tag{5}$$

$$m_b = \frac{1}{2} \times [starting + (b-1) \times l_a + starting + b \times l_a]. \tag{6}$$

According to the TAIEX for the year 2000, the universe of discourse for observations, U, was defined as [4600,10300]. The length of the intervals was determined as 100. The U was partitioned into equal-length intervals $u_1 \sim u_{57}$. The midpoints of these intervals were $m_1 \sim m_{57}$, respectively. Based on Eq. (5), the intervals were set as $u_1 = [4600, 4700]$, $u_2 = [4700, 4800]$, ... Using Eq. (6), the midpoints were set as $m_1 = 4650$, $m_2 = 4750$, ...

Step 2: Defining fuzzy sets for observations.

Each linguistic observation, A_i, can be defined by the intervals u_1, u_2, u_3, $\ldots$, u_b.

$$A_i = f_{Ai}(u_1)/u_1 + f_{Ai}(u_2)/u_2 + \cdots + f_{Ai}(u_b)/u_b.$$

For the TAIEX for the year 2000, each linguistic observation, A_i, was defined as follows:

$$
\begin{aligned}
A_1 &= 1.0/u_1 + 0.5/u_2 + 0/u_3 + 0/u_4 + 0/u_5 + 0/u_6 + \cdots + 0/u_{55} + 0/u_{56} + 0/u_{57} \\
A_2 &= 0.5/u_1 + 1.0/u_2 + 0.5/u_3 + 0/u_4 + 0/u_5 + 0/u_6 + \cdots + 0/u_{55} + 0/u_{56} + 0/u_{57} \\
&\vdots \\
A_{56} &= 0/u_1 + 0/u_2 + 0/u_3 + 0/u_4 + 0/u_5 + 0/u_6 + \cdots + 0.5/u_{55} + 1.0/u_{56} + 0.5/u_{57} \\
A_{57} &= 0/u_1 + 0/u_2 + 0/u_3 + 0/u_4 + 0/u_5 + 0/u_6 + \cdots + 0/u_{55} + 0.5/u_{56} + 1.0/u_{57}
\end{aligned}
$$

Step 3: Fuzzifying the observations.

Each observation is fuzzified to a fuzzy set. As in [1,4], an observation is fuzzified to A_i if the maximal degree of membership of that observation is in A_i.

For example, the TAIEX for 2000/10/2 was 6024.07, which was mapped to A_{15}, because A_{15} had a degree of membership of 1.0. Similarly, the TAIEX for 2000/10/3 was 6143.44, which was mapped to A_{16}. Some of the observations are listed in Table 1.

Step 4: Establishing the fuzzy relationship (neural network training).

We used the backpropagation neural network to establish (or train) the fuzzy relationships in these FLRs. For each FLR, $A \rightarrow A_j$, i became the input and j its corresponding output. Based on the fuzzy sets in Step 3, some FLRs were established as follows:

$$A_{15} \rightarrow A_{16}, \ldots, A_{15} \rightarrow A_{18}, \ldots, A_{15} \rightarrow A_{13}, \ldots, A_{15} \rightarrow A_{14}, \ldots$$

These FLRs became the input and output patterns for neural network training. Some of them are listed in Table 2.

Table 1
Fuzzy TAIEX

Date	TAIEX	Fuzzy TAIEX
…	…	…
2000/10/2	6024.07	A_{15}
2000/10/3	6143.44	A_{16}
2000/10/4	5997.92	A_{14}
2000/10/5	6029.65	A_{15}
2000/10/6	6353.67	A_{18}
2000/10/7	6352.03	A_{18}
2000/10/9	6209.42	A_{17}
2000/10/11	6040.55	A_{15}
2000/10/12	5805.01	A_{13}
2000/10/13	5876.11	A_{13}
2000/10/16	5630.95	A_{11}
2000/10/17	5702.36	A_{12}
2000/10/18	5432.23	A_9
2000/10/19	5081.28	A_5
2000/10/20	5404.78	A_9
2000/10/21	5599.74	A_{10}
2000/10/23	5680.95	A_{11}
2000/10/24	5918.63	A_{14}
2000/10/25	6023.78	A_{15}
2000/10/26	5941.85	A_{14}
2000/10/27	5805.17	A_{13}
2000/10/30	5659.08	A_{11}
2000/10/31	5544.18	A_{10}

Table 2
FLRs

…	$A_{15}\to A_{18}$	$A_{15}\to A_{13}$	$A_{12}\to A_9$	$A_{10}\to A_{11}$	$A_{14}\to A_{13}$
$A_{15}\to A_{16}$	$A_{18}\to A_{18}$	$A_{13}\to A_{13}$	$A_9\to A_5$	$A_{11}\to A_{14}$	$A_{13}\to A_{11}$
$A_{16}\to A_{14}$	$A_{18}\to A_{17}$	$A_{13}\to A_{11}$	$A_5\to A_9$	$A_{14}\to A_{15}$	$A_{11}\to A_{10}$
$A_{14}\to A_{15}$	$A_{17}\to A_{15}$	$A_{11}\to A_{12}$	$A_9\to A_{10}$	$A_{15}\to A_{14}$	

Step 5: Forecasting

The basic and hybrid models are realized below. The basic model uses a neural network approach to forecast all of the observations, while the hybrid model used the same neural network approach to forecast the known patterns as well as a simple method to forecast the unknown patterns.

Basic model: Suppose $F(t-1) = A_{i'}$. To facilitate calculation, we set i' as the input for forecasting. Suppose the output from the neural network is j'. We say that the fuzzy forecast is $A_{j'}$. In other words,

$$F(t) = A_{j'}. \tag{7}$$

Hybrid model: Suppose $F(t-1) = A_{i'}$. If $A_{i'}$ is a known pattern, we follow the basic model to get the fuzzy forecast. If $A_{i'}$ is an unknown pattern, then following Chen's model [1] we simply use $A_{i'}$ as the fuzzy forecast for $F(t)$. That is,

$$F(t) = A_{i'}. \tag{8}$$

Some examples are used to illustrate the forecasting:

Basic model: For a known pattern, for example, the TAIEX for 2000/11/8 was 6067.94, mapped to A_{15}. In other words, $F(t-1) = F(2000/11/8) = A_{15}$. Corresponding to the input, 15, the output from the neural network was 14. According to Eq. (7), $F(t) = F(2000/11/9) = A_{14}$.

For the unknown patterns, we also adopted the outputs from the neural network. For example, the TAIEX for 2000/11/17 was 5351.36, mapped to A_8. In other words, $F(t-1) = F(2000/11/17) = A_8$. Corresponding to the input, 8, the output from the neural network was 9. According to Eq. (7), $F(t) = F(2000/11/18) = A_9$.

Hybrid model: For the known patterns, we also adopted the outputs from the neural network. The fuzzy forecasts for 2000/11/9, 2000/11/10, and 2000/11/13 were obtained like those in the basic model, which were all A_{13}.

For the unknown patterns, for example, the TAIEX for 2000/11/17 was 5351.36, mapped to A_8. In other words, $F(t-1) = F(2000/11/17) = A_8$. According to Eq. (8), $F(t) = F(2000/11/18) = A_8$. A comparison of the fuzzy forecasts of both models is presented in Table 3.

Step 6: Defuzzifying.

Regardless of either model, the defuzzified forecast is equal to the midpoint of the fuzzy forecast.

Suppose the fuzzy forecast of $F(t)$ is $= A_{k'}$. The defuzzified forecast is equal to the midpoint of $A_{k'}$; i.e.,

$$forecast_t = m_{k'}. \tag{9}$$

For example, $F(2000/11/9) = A_{14}$; $forecast_{2000/11/9} = m_{14} = 5950$.

5. Empirical analysis

The empirical analysis was conducted and the respective performances of the models compared.

5.1. Data

We used the daily TAIEX closing prices covering the period from the year 1991–2003 for our empirical analysis, where the data from January to October in each year were used for estimation (training), while those for November and December were used for forecasting. Following the fuzzy time series model, we first needed to determine the starting value for the universe of discourse. Hence, we rounded down the minimal data each year to the nearest hundred as the starting. All of the information is listed in Table 4.

Table 3
A comparison of forecasts in the year 2000

Date	TAIEX	Basic model		Hybrid model		Chen
		Fuzzy set	Forecast	Fuzzy set	Forecast	Forecast
2000/11/2	5626.08	A_{10}	5550	A_{10}	5550	5300
2000/11/3	5796.08	A_{11}	5650	A_{11}	5650	5750
2000/11/4	5677.30	A_{12}	5750	A_{12}	5750	5450
2000/11/6	5657.48	A_{11}	5650	A_{11}	5650	5750
2000/11/7	5877.77	A_{11}	5650	A_{11}	5650	5750
2000/11/8	6067.94	A_{13}	5850	A_{13}	5850	5750
2000/11/9	6089.55	A_{14}	5950	A_{14}	5950	6075
2000/11/10	6088.74	A_{14}	5950	A_{14}	5950	6075
2000/11/13	5793.52	A_{14}	5950	A_{14}	5950	6075
2000/11/14	5772.51	A_{12}	5750	A_{12}	5750	5450
2000/11/15	5737.02	A_{12}	5750	A_{12}	5750	5450
2000/11/16	5454.13	A_{12}	5750	A_{12}	5750	5450
2000/11/17	5351.36	A_{10}	5550	A_{10}	5550	5300
2000/11/18	5167.35	A_9	5450	A_8	**5350**	5350
2000/11/20	4845.21	A_8	5350	A_6	**5150**	5150
2000/11/21	5103.00	A_7	5250	A_3	**4850**	4850
2000/11/22	5130.61	A_8	5350	A_6	**5150**	5150
2000/11/23	5146.92	A_8	5350	A_6	**5150**	5150
2000/11/24	5419.99	A_8	5350	A_6	**5150**	5150
2000/11/27	5433.78	A_{10}	5550	A_{10}	5550	5300
2000/11/28	5362.26	A_{10}	5550	A_{10}	5550	5300
2000/11/29	5319.46	A_9	5450	A_8	**5350**	5350
2000/11/30	5256.93	A_9	5450	A_8	**5350**	5350
2000/12/1	5342.06	A_9	5450	A_7	**5250**	5250
2000/12/2	5277.35	A_9	5450	A_8	**5350**	5350
2000/12/4	5174.02	A_9	5450	A_7	**5250**	5250
2000/12/5	5199.20	A_8	5350	A_6	**5150**	5150
2000/12/6	5170.62	A_8	5350	A_6	**5150**	5150
2000/12/7	5212.73	A_8	5350	A_6	**5150**	5150
2000/12/8	5252.83	A_9	5450	A_7	**5250**	5250
2000/12/11	5284.41	A_9	5450	A_7	**5250**	5250
2000/12/12	5380.09	A_9	5450	A_7	**5250**	5250
2000/12/13	5384.36	A_9	5450	A_8	**5350**	5350
2000/12/14	5320.16	A_9	5450	A_8	**5350**	5350
2000/12/15	5224.74	A_9	5450	A_8	**5350**	5350
2000/12/16	5134.10	A_9	5450	A_7	**5250**	5250
2000/12/18	5055.20	A_8	5350	A_6	**5150**	5150
2000/12/19	5040.25	A_8	5350	A_8	5350	5450
2000/12/20	4947.89	A_8	5350	A_8	5350	5450
2000/12/21	4817.22	A_7	5250	A_4	**4950**	4950
2000/12/22	4811.22	A_7	5250	A_3	**4850**	4850
2000/12/26	4721.36	A_7	5250	A_3	**4850**	4850
2000/12/27	4614.63	A_6	5150	A_2	**4750**	4750
2000/12/28	4797.14	A_5	5050	A_1	**4650**	4650
2000/12/29	4743.94	A_6	5150	A_2	**4750**	4750
2000/12/30	4739.09	A_6	5150	A_2	**4750**	4750

Note: The figures in bold were the forecasts in the hybrid model that were different from those in the basic model.

5.2. Neural network statistics

The neural network training statistics are depicted in Fig. 3. The x-axis represents the amount of training, starting from 0 to 2990. The y-axis represents the forecast error and RMSE, ranging from 0 to 0.5. We found

Table 4
Setup for TAIEX forecasting

	1991	1992	1993	1994	1995	1996	1997	1998	1999	2000	2001	2002	2003
Estimation	1/1–10/30	1/4–10/30	1/5–10/30	1/5–10/29	1/5–10/30	1/4–10/30	1/4–10/30	1/3–10/31	1/5–10/30	1/4–10/31	1/2–10/31	1/2–10/31	1/2–10/31
Forecasting	11/1–12/28	11/2–12/29	11/2–12/31	11/1–12/31	11/1–12/30	11/1–12/31	11/3–12/31	11/2–12/31	11/1–12/28	11/1–12/30	11/1–12/31	11/1–12/31	11/3–12/31
Starting	3300	3300	3100	5100	4500	4600	6800	6200	5400	4600	3400	3800	4100

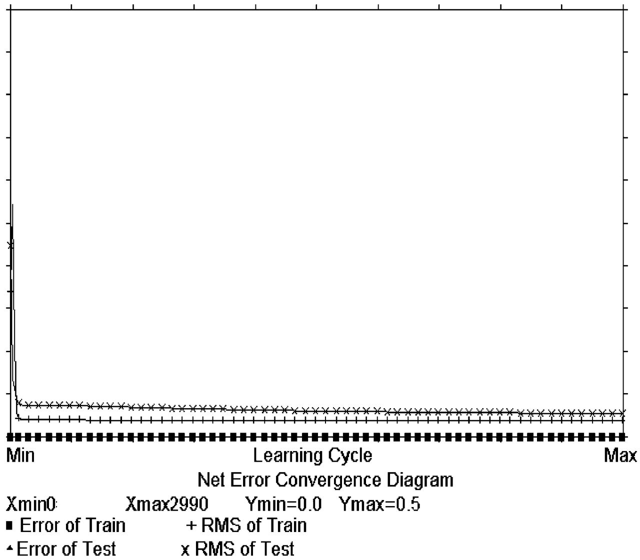

Fig. 3. Training convergence.

Table 5
Comparison of performance (RMSE)

	1991	1992	1993	1994	1995	1996	1997	1998	1999	2000	2001	2002	2003	average
Basic model	54	54	228	79	74	133	141	121	109	255	130	84	56	117
Hybrid model	54	54	**107**	79	74	**73**	141	121	109	**152**	130	84	56	95
Chen	71	61	107	107	65	64	154	134	120	176	148	101	74	106

Note: The figures in bold are used where there were unknown patterns in that year.

that the RMSE for training decreased drastically and converged quickly. This indicates that the training was quite successful.

5.3. Overall performance

The respective performances of the two models were compared with that of Chen's model in Table 5. First, from the averages of the RMSEs, we found that the hybrid model performed the best and the basic model the worst. In terms of the yearly comparison, the hybrid model performed better than Chen's model in 10 out of 12 years, while the basic model performed better than Chen's model in 8 out of 12 years. (Note that the RMSEs for the year 1993 were the same.)

Second, concerning the known patterns, the basic and hybrid models outperformed Chen's model in 9 out of 10 years. This indicates that the backpropagation neural network is good at formulating known patterns. For the unknown patterns, the hybrid model and Chen's model performed about the same, while the basic model was worse than both the hybrid model and Chen's model in all 3 years. This indicates that the neural network is not suitable for handling unknown patterns.

5.4. Forecasting detail

After knowing the overall performance, we further probed into the details of the forecasting. To explain how the forecasts differed among the basic and hybrid models and Chen's model, we took the year 2000 as an example. For the known patterns, the fuzzy forecasts from the hybrid model were the same as those from the

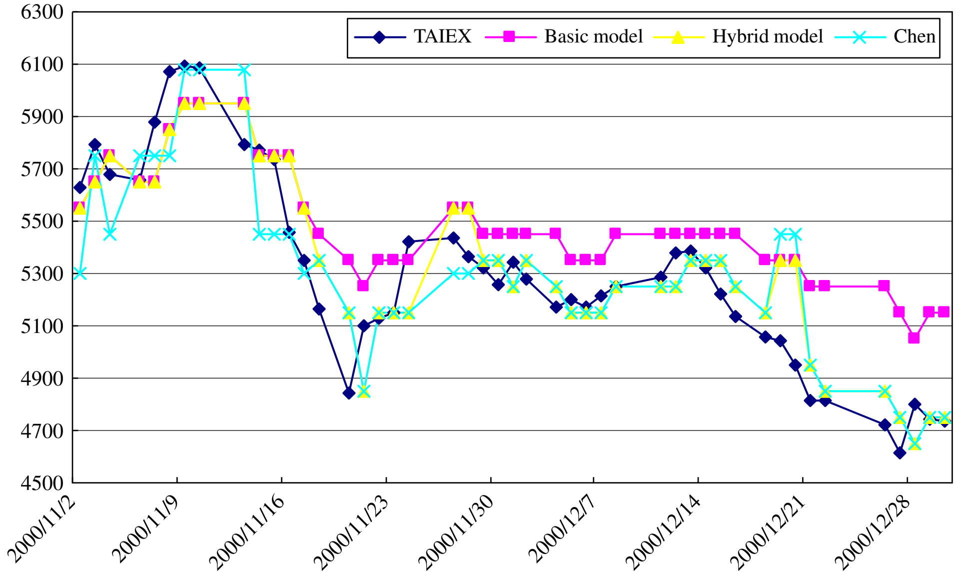

Fig. 4. Comparison of forecasts for year 2000.

basic model, while for the unknown patterns, the ones from the hybrid model were the same as those from Chen's model. For a known pattern, the actual TAIEX for 2000/11/14 was 5772.51. The defuzzified forecasts for the basic and the hybrid models were both 5750. Both absolute differences were 22.51. By contrast, the defuzzified forecast for Chen's model was 5450. The absolute difference was 322.51. Hence, the neural network forecast was better than that of Chen's model in this case.

For the unknown patterns, the actual TAIEX on 2000/11/22 was 5130.61. The defuzzified forecasts were 5150 for both the hybrid model and Chen's model; both absolute differences were 19.39. The defuzzified forecast was 5350 for the basic model. The absolute difference was 219.39. Chen's models outperformed the neural network in this case. The defuzzified forecasts for all of the models are depicted in Fig. 4.

6. Conclusions

This study applied a backpropagation neural network to assist in fuzzy time series modeling. Three important tasks were performed in terms of the model's construction, in conformity with the evaluation criteria [33]. First, we compared the forecasting results with a conventional fuzzy time series model, i.e. Chen's model, to demonstrate the superiority of the proposed model. Second, we separated the observations into in-sample (for estimation) and out-of-sample (for forecasting) observations. In addition, the ratio of both observations was 83%:17%.

We proposed two models for forecasting. The basic model applied neural networks to forecast all the out-of-sample observations. The hybrid model applied neural networks to forecast the known patterns (as in the basic model) and the simple method to forecast the unknown patterns (as in Chen's model) in the out-of-sample observations. From the empirical analysis, the neural network was found to perform better when forecasting known patterns, but not when forecasting unknown patterns. As a result, the hybrid model outperformed the basic model as well as a conventional fuzzy time series model.

Acknowledgments

The software we used was PC Neuron, developed by Prof. Yeh, I-Cheng, Department of Information Management, Chung Hua University, Taiwan. We would like to thank him for providing the institutional version.

References

[1] S.-M. Chen, Forecasting enrollments based on fuzzy time series, Fuzzy Sets and Systems 81 (1996) 311–319.
[2] S.-M. Chen, Forecasting enrollments based on high-order fuzzy time series, Cybernet. Syst.: Int. J. 33 (1) (2000) 1–16.
[3] J.R. Hwang, S.-M. Chen, C.-H. Lee, Handling forecasting problems using fuzzy time series, Fuzzy Sets and Systems 100 (1998) 217–228.
[4] Q. Song, B.S. Chissom, Forecasting enrollments with fuzzy time series – Part 1, Fuzzy Sets and Systems 54 (1993) 1–9.
[5] Q. Song, B.S. Chissom, Forecasting enrollments with fuzzy time series—Part 2, Fuzzy Sets and Systems 62 (1994) 1–8.
[6] K. Huarng, Heuristic models of fuzzy time series for forecasting, Fuzzy Sets and Systems 123 (3) (2001) 369–386.
[7] K. Huarng, Effective lengths of intervals to improve forecasting in fuzzy time series, Fuzzy Sets and Systems 123 (3) (2001) 387–394.
[8] K. Huarng, H.-K. Yu, A Type 2 fuzzy time series model for stock index forecasting, Physica A 353 (2005) 445–462.
[9] H.-K. Yu, A refined fuzzy time-series model for forecasting, Physica A 346 (3–4) (2005) 657–681.
[10] H.-K. Yu, Weighted fuzzy time series models for TAIEX forecasting, Physica A 349 (2005) 609–624.
[11] S.-M. Chen, J.R. Hwang, Temperature prediction using fuzzy time series, IEEE Trans. Syst. Man Cybern. B 30 (2) (2000) 263–275.
[12] K. Huarng, H.-K. Yu, Ratio-based lengths of intervals to improve enrollment forecasting, in: Ninth International Conference on Fuzzy Theory and Technology, Cary, NC, USA, 2003.
[13] K. Huarng, H.-K. Yu, A dynamic approach to adjusting lengths of intervals in fuzzy time series forecasting, Intell. Data Anal. 8 (1) (2004) 3–27.
[14] K. Huarng, H.-K. Yu, An *N*th order heuristic fuzzy time series model for TAIEX forecasting, Int. J. Fuzzy Syst. 5 (4) (2003) 247–253.
[15] J. Sullivan, W.H. Woodall, A comparison of fuzzy forecasting and Markov modeling, Fuzzy Sets and Systems 64 (1994) 279–293.
[16] K.A. Smith, J.N.D. Gupta, Neural Networks in Business: Techniques and Applications, Idea Group Publishing, Hershey, PA, 2002.
[17] B. Widrow, D. Rumelhart, M.A. Lehr, Neural networks: Applications in industry, business and science, Commun. ACM 37 (3) (1994) 93–105.
[18] G. Zhang, B.E. Patuwo, M.Y. Hu, Forecasting with artificial neural networks: the state of the art, Int. J. Forecast. 14 (1998) 35–62.
[19] D.C. Indro, C.X. Jiang, B.E. Patuwo, G.P. Zhang, Predicting mutual fund performance using artificial neural networks, Omega Int. J. Manage. Sci. 27 (1999) 373–380.
[20] P.D. Wasserman, Neural Computing: Theory and Practice, Van Nostrand Reinhold, New York, NY, 1989.
[21] G. Peter Zhang, Business Forecasting with Artificial Neural Networks: An Overview, in: G. Peter Zhang (Ed.), Neural Networks in Business Forecasting, Idea Group Publishing, Hershey, PA, 2004, pp. 1–22.
[22] Q. Song, B.S. Chissom, Fuzzy time series and its models, Fuzzy Sets and Systems 54 (1993) 269–277.
[23] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning internal representation by error propagation, in: D.E. Rumelhart, J.L. McClelan, PDP Research Group (Eds.), Parallel Distributed Processing: Explorations in the Microstructure of Cognition, vol. 1, 1986, MIT Press, Cambridge, MA, pp. 318–362.
[24] G. Cybenko, Approximation by superpositions of sigmoidal function, Math. Control Signals Syst. 2 (1989) 303–314.
[25] K. Funahashi, On the approximate realization of continuous mappings by neural networks, Neural Networks 2 (1989) 183–192.
[26] K. Hornik, Approximation capabilities of multiplier feed-forward networks, Neural Networks 4 (1991) 251–257.
[27] K. Hornik, Some new results on neural network approximation, Neural Networks 6 (1993) 1069–1072.
[28] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, Neural Networks 2 (1989) 359–366.
[29] W. Remus, M. O'Connor, Neural networks for time-series forecasting, in: J.S. Armstrong (Ed.), Principles of Forecasting: A Handbook for Researchers and Practitioners, Kluwer Academic Publishers, Norwell, MA, 2001, pp. 245–256.
[30] R.D. Reed, R.J. Marks II, Neural Smithing: Supervised Learning in Feedforward Artificial Neural Networks, The MIT Press, Cambridge, MA, 1999.
[31] S. Makridakis, A. Andersen, R. Carbone, R. Fildes, M. Hibon, R. Lewandowski, J. Newton, E. Parzen, R. Winkler, The accuracy of extrapolation (time series) methods: Results of a forecasting competition, J. Forecast. 1 (1982) 111–153.
[32] S. Makridakis, C. Chatfield, M. Hibon, M.J. Laurence, T. Mills, K. Ord, L.F. Simmons, The M-2 competition: A real-time judgmentally based forecasting competition, J. Forecast. 9 (1993) 5–22.
[33] M. Adya, F. Collopy, How effective are neural networks at forecasting and prediction? A review and evaluation, J. Forecast. 17 (1998) 481–495.