Available online at www.sciencedirect.com
ScienceDirect
Expert Systems with Applications 34 (2008) 2945–2952
Expert Systems with Applications
www.elsevier.com/locate/eswa

A bivariate fuzzy time series model to forecast the TAIEX

Tiffany Hui-Kuang Yu [a,*], Kun-Huang Huarng [b,1]

[a] *Department of Public Finance, Feng Chia University, 100 Wenhwa Road, Seatwen, Taichung 407, Taiwan, ROC*
[b] *Department of International Trade, Feng Chia University, 100 Wenhwa Road, Seatwen, Taichung 407, Taiwan, ROC*

Abstract

Fuzzy time series models have been applied to forecast various domain problems and have been shown to forecast better than other models. Neural networks have been very popular in modeling nonlinear data. In addition, the bivariate models are believed to outperform the univariate models. Hence, this study intends to apply neural networks to fuzzy time series forecasting and to propose bivariate models in order to improve forecasting. The stock index and its corresponding index futures are taken as the inputs to forecast the stock index for the next day. Both in-sample estimation and out-of-sample forecasting are conducted. The proposed models are then compared with univariate models as well as other bivariate models. The empirical results show that one of the proposed models outperforms the many other models.

Keywords: Bivariate models; Futures index; Fuzzy time series; Neural networks; Stock index

1. Introduction

Conventional time series models have for a long time been applied to forecasting. Yet different fuzzy time series models have been proposed to forecast nonlinear data (Chen, 1996; Chen & Hwang, 2000; Huarng & Yu, 2005, 2006; Hwang, Chen, & Lee, 1998; Song & Chissom, 1993a, 1993b, 1994; Sullivan & Woodall, 1994; Yu, 2005a, 2005b). These fuzzy time series models have been applied to various applications, such as enrollment (Chen, 1996; Hwang et al., 1998; Song & Chissom, 1993a, 1993b, 1994; Sullivan & Woodall, 1994), temperature (Chen & Hwang, 2000), the stock index (Huarng, 2001; Huarng & Yu, 2005, 2006; Yu, 2005a, 2005b), etc. However, they have been limited to one variable applications (referred to as AR(1) in conventional terms).

The objective of this study is to apply neural networks to forecast fuzzy time series, and to propose two bivariate models to improve forecasting results. The reasons for this are threefold. First, fuzzy time series have been applied to several domain problems and have been shown to forecast better. Second, neural networks have been very popular for modeling nonlinear data (Indro, Jiang, Patuwo, & Zhang, 1999; Wasserman, 1989). They have been applied to forecasting fuzzy time series and have performed better than some other models (Huarng & Yu, 2006). Third, bivariate fuzzy time series models have recently been proposed (Hsu, Tse, & Wu, 2003; Huarng, 2001), and have rendered better forecasting results than univariate models. This supports the view that the bivariate models are supposed to outperform the univariate models.

Some neural-fuzzy systems have been applied to forecasting (Jang, 1993; Kim & Kasabov, 1999; Nauck & Kruse, 1999). However, these systems may suffer from the size of fuzzy rules when there are many intervals. A two factor method to forecast temperature and stock index was proposed, based on high-order fuzzy logical relationships and genetic simulated annealing techniques (Lee, Wang, & Chen, 2008). Another two factor high-order method was proposed to forecast temperature and stock index based on genetic algorithms (Lee, Wang, & Chen,

* Corresponding author. Tel.: +886 4 24517250x4309.
E-mail addresses: hkyu@fcu.edu.tw (T.H.-K. Yu), khhuarng@fcu.edu.tw (K.-H. Huarng).
[1] Tel.: +886 4 24517250x4002.

0957-4174/$ - see front matter © 2007 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2007.05.016

2007). However, both studies provided only the estimation results. More sophisticated forecasting results are needed to manifest their advantages.

The Taiwan Stock Exchange Capitalization Weighted Stock Index (TAIEX) and its corresponding index futures, the Taiwan Futures Exchange (TAIFEX), are used as inputs to forecast the TAIEX for the next day. This study will compare the performance of various models, including univariate models (the fuzzy time series model, conventional regression model, neural network model, neural network-based fuzzy time series model (Huarng & Yu, 2006) and neural network-based fuzzy time series model with substitutes (Huarng & Yu, 2006)) and bivariate models (the conventional regression model, neural network model, and the two proposed models: the neural network-based fuzzy time series model and the neural network-based fuzzy time series model with substitutes).

Section 2 reviews the definitions of fuzzy time series and neural networks. Section 3 describes how we deal with the data. Section 4 introduces one neural network bivariate model and proposes two bivariate neural network-based fuzzy time series models. The empirical analyses are elaborated in Section 5. Section 6 concludes the paper.

2. Methodology

We apply neural networks to fuzzy time series forecasting. Hence, we briefly introduce fuzzy time series as well as neural networks.

2.1. 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}: $U \rightarrow [0, 1]$. 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; $f_{Ai}(u_a)[0, 1]$ 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 $Y(t)$ $(t = \ldots, 0, 1, 2, \ldots)$ (Song & Chissom, 1993a, 1993b).

In a univariate model, fuzzy relationships between two consecutive observations can be defined as follows by following Definition 1:

Definition 2. 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) (Song & Chissom, 1993a, 1993b), can be denoted by $A_i \rightarrow A_j$, where A_i is called the LHS (left-hand side) and A_j the RHS (right-hand side) of the FLR.

Song and Chissom proposed a univariate fuzzy time series model that included the following steps (Song & Chissom, 1993a, 1993b): (1) define and partition the universe of discourse; (2) define fuzzy sets for the observations; (3) fuzzify the observations; (4) establish the fuzzy relationship; (5) forecast; and (6) defuzzify the forecasting results. Many fuzzy time series studies, for univariate problems, have followed steps (1) to (3) for fuzzification. However, various models have applied different methods to establish fuzzy relationships; for example, union-min (Song & Chissom, 1993a, 1993b), Cartesian product (Song & Chissom, 1994), matrix multiplication of probability density functions (Sullivan & Woodall, 1994), arithmetic operations (Chen, 1996), and matrix multiplication of the variation matrix (Hwang et al., 1998).

This study applies neural networks to establish the fuzzy relationships, and also targets the bivariate problems. Hence, a bivariate fuzzy time series is defined as follows:

Definition 3. Let F and G be two fuzzy time series. Suppose $F(t-1) = A_i$, $G(t-1) = B_k$, and $F(t) = A_j$. A bivariate FLR is defined as $A_i, B_k \rightarrow A_j$, where A_i, B_k are referred to as the LHS (left-hand side) and A_j as the RHS (right-hand side) of the bivariate FLR.

2.2. Neural networks

Neural networks have been successfully applied to the forecasting of different applications (Smith & Gupta, 2002; Widrow, Rumelhart, & Lehr, 1994; Zhang, Patuwo, & Hu, 1998). The nonlinear structures of neural networks have been very useful in forecasting (Indro et al., 1999; Wasserman, 1989). Hence, this study chooses the neural network to establish the fuzzy relationships in a bivariate fuzzy time series, which is also nonlinear.

A simple neural network is listed in Fig. 1. The leftmost layer is the input layer, consisting of input nodes. Each input node is for an input variable. Hence, the number of input variables is equal to the number of input nodes. The rightmost layer is the output layer, consisting of output nodes. Similarly, each output node is for an output variable, with the number of output variables being equal to the number of output nodes. In this study, because there are two input and one output variables, respectively, there are two input and one output nodes.

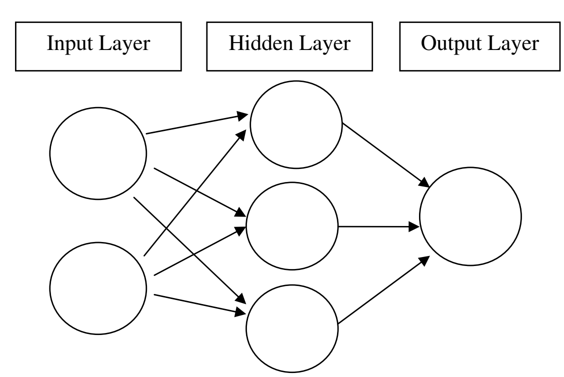

Fig. 1. A neural network structure.

There can also be multiple hidden layers, with each layer consisting of hidden nodes. Many rules have been set to decide the number of hidden layers and hidden nodes in each hidden layer. However, many forecasting applications have used only one hidden layer (Cybenko, 1989; Funahashi, 1989; Hornik, 1991, 1993; Hornik, Stinchcombe, & White, 1989). Hence, in Fig. 1, we also choose one hidden layer. The number of hidden nodes is set as the sum of the input and the output nodes (Huang, Shih, & Liu, 1996). Hence, there are three hidden nodes in Fig. 1.

3. Data

Since closing prices have been used in many previous studies (Kwon & Shin, 1999; Mavrides, 2003; Peláez, 2003), they are also used in this study. The data used are the daily closing prices of the stock index, the TAIEX, and its corresponding index futures, the TAIFEX, in Taiwan. Conceptually, both the TAIEX and TAIFEX at time $t-1$ are taken as the input variables and TAIEX at time t is taken as the output variable (the forecast target).

The data are extracted from the TEJ Database (January 1999–December 2004). In addition, the data are divided into training and testing sets. Many studies have applied a convenient ratio to separate training (in-sample) from testing (out-of-sample) data between the ratios $\frac{7}{10}:\frac{3}{10}$ and $\frac{9}{10}:\frac{1}{10}$ (Peter Zhang, 2004). This study follows the choice presented in (Huarng & Yu, 2006): the data covering January to October are used for training, while the data for November and December are used for testing. Hence, the ratio adopted is $\frac{10}{12}:\frac{2}{12}$, which is a ratio that lies in-between.

4. Models

This study proposes two bivariate models: a bivariate neural network-based fuzzy time series model and a bivariate neural network-based fuzzy time series model with substitutes. A bivariate neural network model is used as a counterpart for comparison. For neural network training, we set the epochs as 1000 and the testing period as 10 for all the neural network-based models. The setup is summarized in Table 1.

4.1. A bivariate neural network model

A bivariate neural network model is depicted in Fig. 2. A simple algorithm is listed as follows:

Table 1
Neural network setup

Number of nodes in the input layer	2
Number of hidden layers	1
Number of nodes in the hidden layer	3
Number of nodes in the output layer	1
Epochs	1000
Testing periods	10

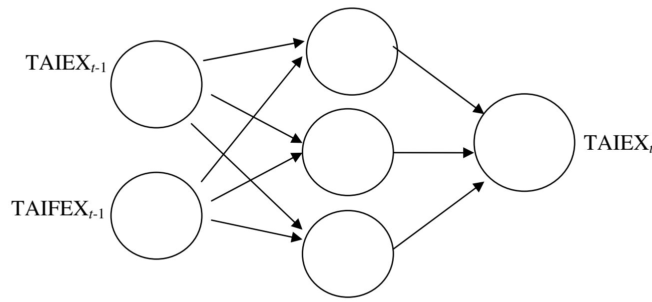

Fig. 2. A bivariate neural network model.

The Algorithm for a Bivariate Neural Network Model

1. Prepare the data for training: the *actual* TAIEX and TAIFEX at time $t-1$ are taken as the inputs and the *actual* TAIEX at time t is taken as the output.
2. Prepare the data for forecasting (or testing in neural network terminology): the *actual* TAIEX and TAIFEX at $t-1+k$ (where there are $k+1$ data for training) are taken as the inputs, and the output from the model is the *actual* forecast for the TAIEX at $t+k$.

For example, the TAIEX at the market close on 1/5/1999 was 6152.43, and that at the market close on 1/6/1999 was 6199.91. In addition, the TAIFEX at the market close on 1/5/1999 was 6120. Hence, 6152.43 and 6199.91 become the inputs of the neural network and 6120 is the corresponding output for training. After training, we can proceed to forecast. For example, on 28/12/1999, the TAIEX was 8448.84 and the TAIFEX was 8564. The output (or forecast) from the trained neural network is 8347.8. The forecasts for November 1999 are listed in Table 2.

4.2. A bivariate neural network-based fuzzy time series model

A bivariate neural network-based fuzzy time series model is depicted in Fig. 3. The algorithm is listed as follows.

The Algorithm for a Bivariate Neural Network-Based Fuzzy Time Series Model

1. Fuzzify the actual data to fuzzy data.
2. Prepare data for training: the *fuzzy* TAIEX, $F(t-1)$, and TAIFEX at $t-1$, $G(t-1)$, are taken as the inputs and the *fuzzy* TAIEX at t, $F(t)$, is taken as the output.
3. Prepare data for forecasting: both the *fuzzy* TAIEX, $F(t+k-1)$, and the *fuzzy* TAIFEX at $t-1+k$, $F(t+k-1)$, are taken as the inputs to the trained neural network and the output from the model is the *fuzzy* forecast for the TAIEX at $t+k$, $F(t+k)$.
4. Defuzzify each fuzzy forecast, $F(t+k)$.

We take the year 1999 as our example. Because the minimal and maximal TAIEX data in 1999 were 5474.79 and 8608.91, respectively, we set the universe of discourse for

Table 2
Forecasts from the bivariate neural network model

Date (t)	Actual TAIEX (t)	Inputs		Output
		TAIEX ($t-1$)	TAIFEX ($t-1$)	TAIEX (t)
2/11/1999	7721.59	7814.89	7911	7872.50
3/11/1999	7580.09	7721.59	7820	7783.70
4/11/1999	7469.23	7580.09	7678	7647.90
5/11/1999	7488.26	7469.23	7565	7533.00
6/11/1999	7376.56	7488.26	7619	7569.60
8/11/1999	7401.49	7376.56	7400	7397.20
9/11/1999	7362.69	7401.49	7441	7433.70
10/11/1999	7401.81	7362.69	7382	7381.50
11/11/1999	7532.22	7401.81	7443	7433.70
15/11/1999	7545.03	7532.22	7575	7574.80
16/11/1999	7606.20	7545.03	7582	7585.20
17/11/1999	7645.78	7606.20	7629	7637.50
18/11/1999	7718.06	7645.78	7669	7679.20
19/11/1999	7770.81	7718.06	7741	7747.10
20/11/1999	7900.34	7770.81	7800	7799.40
22/11/1999	8052.31	7900.34	7980	7940.40
23/11/1999	8042.19	8052.31	8139	8071.00
24/11/1999	7921.85	8042.19	8130	8065.80
25/11/1999	7904.53	7921.85	7965	7945.60
26/11/1999	7595.44	7904.53	8016	7956.10
29/11/1999	7823.90	7595.44	7700	7663.60
30/11/1999	7720.87	7823.90	7855	7851.60

fuzzification as [5400, 8700]. Then, we set the length of the intervals as 100. Hence, we have the intervals $u_1 = [5400, 5500]$, $u_2 = [5500, 5600]$, $u_3 = [5600, 5700], \ldots,$ of which the midpoints are 5450, 5550, 5650,…, respectively. The midpoints are the corresponding defuzzified forecasts for the fuzzy forecasts. Each fuzzy set, A_i, is defined by the intervals $u_1, u_2, u_3, \ldots, u_{33}$.

$$A_1 = 1/u_1 + 0.5/u_2 + 0/u_3 + \cdots + 0/u_{32} + 0/u_{33}$$
$$A_2 = 0.5/u_1 + 1/u_2 + 0.5/u_3 + \cdots + 0/u_{32} + 0/u_{33}$$
$$\ldots$$
$$A_{32} = 0/u_1 + 0/u_2 + 0/u_3 + \cdots + 1/u_{32} + 0.5/u_{33}$$
$$A_{33} = 0/u_1 + 0/u_2 + 0/u_3 + \cdots + 0.5/u_{32} + 1/u_{33}$$

Following (Chen, 1996), the fuzzification can be conducted. The TAIEX at the market close on 1/5/1999, for example, was 6152.43, which is fuzzified to A_8. The TAIEX at the market close on the next day (1/6/1999) was 6199.91, which is fuzzified to A_8. We conduct a similar process for the TAIFEX. We set the universe of discourse for fuzzification as [5500, 8700] and the length of the intervals as 100. For example, the TAIFEX at the market close on 1/5/1999 was 6120, which is fuzzified to B_7. Then, we can fuzzify all the TAIFEX data.

After fuzzification, we can establish a bivariate FLR, such as $A_8, B_7 \rightarrow A_8$. Some other FLRs are listed in Table 3. All the FLRs from the training data are taken as the inputs. For example, from $A_8, B_7 \rightarrow A_8$, 8 and 7 are the fuzzy inputs in the neural network, and 8 is its corresponding output. After training, the fuzzy inputs from the testing data are taken as the inputs in the neural network for forecasting. For example, on 28/12/1999, the TAIEX was 8448.84 and the TAIFEX was 8564. Both are fuzzified to A_{31}, B_{31}, respectively. Then, the fuzzy forecast from the neural network is 30; i.e. A_{30}.

Then we defuzzify A_{30} to 8350 (the midpoint of A_{30}), which is the forecast. It is worth noting that some input patterns may not appear in the training sample. However, we still apply the neural network to perform the forecast using this model. The forecasts for November 1999 are listed in Table 4.

4.3. A bivariate neural network-based fuzzy time series model with substitutes

The bivariate neural network-based fuzzy time series model with substitutes is very similar to that of the bivariate neural network-based fuzzy time series model. The neural network structure of the former model is the same as that of the latter model. Only the forecasting of the empty fuzzy relationships is different. The algorithm is listed as follows:

The algorithm for a bivariate neural network-based fuzzy time series model with substitutes

1. Fuzzify the actual data to fuzzy data.
2. Prepare data for training: the *fuzzy* TAIEX, $F(t-1)$, and TAIFEX at time $t-1$, $G(t-1)$, are taken as the inputs and the *fuzzy* TAIEX at time t, $F(t)$, is taken as the output.
3. Prepare data for forecasting:
   IF the *fuzzy* TAIEX, $F(t+k-1)$, and the *fuzzy* TAIFEX at time $t+k-1$, $G(t+k-1)$, together appear in the training data,

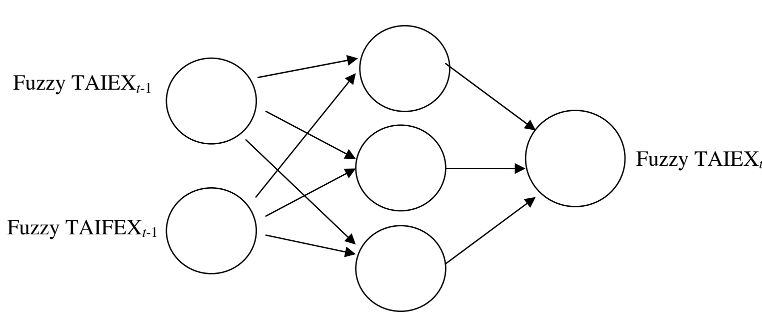

Fig. 3. A bivariate neural network-based fuzzy time series model.

Table 3
Fuzzy sets and fuzzy logic relationships

Date	Stock closing	Fuzzy set A	Futures closing	Fuzzy set B	Fuzzy logic relationship
5/1/1999	6152.43	8	6120	7	$A8,\ B7 \to A8$
6/1/1999	6199.91	8	6245	8	$A8,\ B8 \to A11$
7/1/1999	6404.31	11	6510	11	$A11,\ B11 \to A11$
8/1/1999	6421.75	11	6452	10	$A11,\ B10 \to A11$
11/1/1999	6406.99	11	6435	10	$A11,\ B10 \to A10$
12/1/1999	6363.89	10	6390	9	$A10,\ B9 \to A10$
13/1/1999	6319.34	10	6352	9	$A10,\ B9 \to A9$
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.

Table 4
Forecasts from the bivariate neural network-based fuzzy time series model

Date (t)	Actual TAIEX (t)	Inputs		Output	
		TAIEX (A_{t-1})	TAIFEX (B_{t-1})	TAIEX (A_t)	TAIEX (defuzzified A_t)
2/11/1999	7721.59	25	25	25	7850
3/11/1999	7580.09	24	24	25	7850
4/11/1999	7469.23	22	22	23	7650
5/11/1999	7488.26	21	21	22	7550
6/11/1999	7376.56	21	22	22	7550
8/11/1999	7401.49	20	20	21	7450
9/11/1999	7362.69	21	20	21	7450
10/11/1999	7401.81	20	19	20	7350
11/11/1999	7532.22	21	20	21	7450
15/11/1999	7545.03	22	21	22	7550
16/11/1999	7606.20	22	21	22	7550
17/11/1999	7645.78	23	22	23	7650
18/11/1999	7718.06	23	22	23	7650
19/11/1999	7770.81	24	23	22	7550
20/11/1999	7900.34	24	24	25	7850
22/11/1999	8052.31	26	25	26	7950
23/11/1999	8042.19	27	27	27	8050
24/11/199	7921.85	27	27	27	8050
25/11/1999	7904.53	26	25	26	7950
26/11/1999	7595.44	26	26	26	7950
29/11/1999	7823.90	22	23	23	7650
30/11/1999	7720.87	25	24	25	7850

THEN both are taken as the inputs to the trained neural network and the output from the model is the *fuzzy* forecast for TAIEX at time $t+k$, $F(t+k)$, or
ELSE the *fuzzy* TAIEX at time $t+k-1$ becomes the *fuzzy* forecast for TAIEX at time $t+k$; i.e., $F(t+k) = F(t+k-1)$.

4. Defuzzify each fuzzy forecast, $F(t+k)$.

During the establishing of bivariate FLRs, some input patterns may not show up in the training data. For example, the bivariate FLR A_{24}, B_{24} never appears in the 1999 training sample. The bivariate neural network-based fuzzy time series model still applies the trained neural network to the forecasting. Hence, the output from that model for 3/11/1999 is 25 (or A_{25}), which is defuzzified to 7850. Thus, 7850 is the forecast, as shown in Table 4.

Table 5
Forecasts from the neural network-based fuzzy time series model with substitutes

Date (t)	Actual TAIEX (t)	Inputs		Output	
		TAIEX (A_{t-1})	TAIFEX (B_{t-1})	TAIEX (A_t)	TAIEX (defuzzified A_t) or substitutes
2/11/1999	7721.59	25	25	25	7850
3/11/1999	7580.09	24	24	$\emptyset$	7750
4/11/1999	7469.23	22	22	23	7650
5/11/1999	7488.26	21	21	22	7550
6/11/1999	7376.56	21	22	$\emptyset$	7450
8/11/1999	7401.49	20	20	21	7450
9/11/1999	7362.69	21	20	21	7450
10/11/1999	7401.81	20	19	20	7350
11/11/1999	7532.22	21	20	21	7450
15/11/1999	7545.03	22	21	22	7550
16/11/1999	7606.20	22	21	22	7550
17/11/1999	7645.78	23	22	23	7650
18/11/1999	7718.06	23	22	23	7650
19/11/1999	7770.81	24	23	22	7550
20/11/1999	7900.34	24	24	$\emptyset$	7750
22/11/1999	8052.31	26	25	26	7950
23/11/1999	8042.19	27	27	27	8050
24/11/1999	7921.85	27	27	27	8050
25/11/1999	7904.53	26	25	26	7950
26/11/1999	7595.44	26	26	26	7950
29/11/1999	7823.90	22	23	$\emptyset$	7550
30/11/1999	7720.87	25	24	25	7850

However, in this model, we follow the approach of treating the empty RHS in the FLRs in (Chen, 1996) by using the midpoint of the A_{t-1} as the forecast. For example, the bivariate FLR for 3/11/1999 is A_{24}, $B_{24} \to \emptyset$. Hence, instead of using the neural network for forecasting, we take the mid value of the result of A_{24} as the forecast, which is 7550. The forecasts for November 1999 are listed in Table 5. We compare the forecasts for all these models for November 1999 as seen in Fig. 4.

5. Empirical analysis

We conduct the empirical analysis to compare the performance of the proposed models with univariate models in (Huarng & Yu, 2006) as well as other bivariate models.

5.1. Performance evaluation

To facilitate the comparison, we use the root mean squared error (RMSE):

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{t=1}^{n-1}(\mathrm{forecast}_{t+1} - \mathrm{actual}_{t+1})^2}{n-1}}$$

where t represents the date and there are n forecast data; $\mathrm{forecast}_{t+1}$ is the forecast at $t+1$ from any model and actual_{t+1} is the actual stock index at $t+1$.

In addition, we rank the performance each year to show the superiority and then use the average rank to indicate the overall performance. We sort the RMSEs for all years

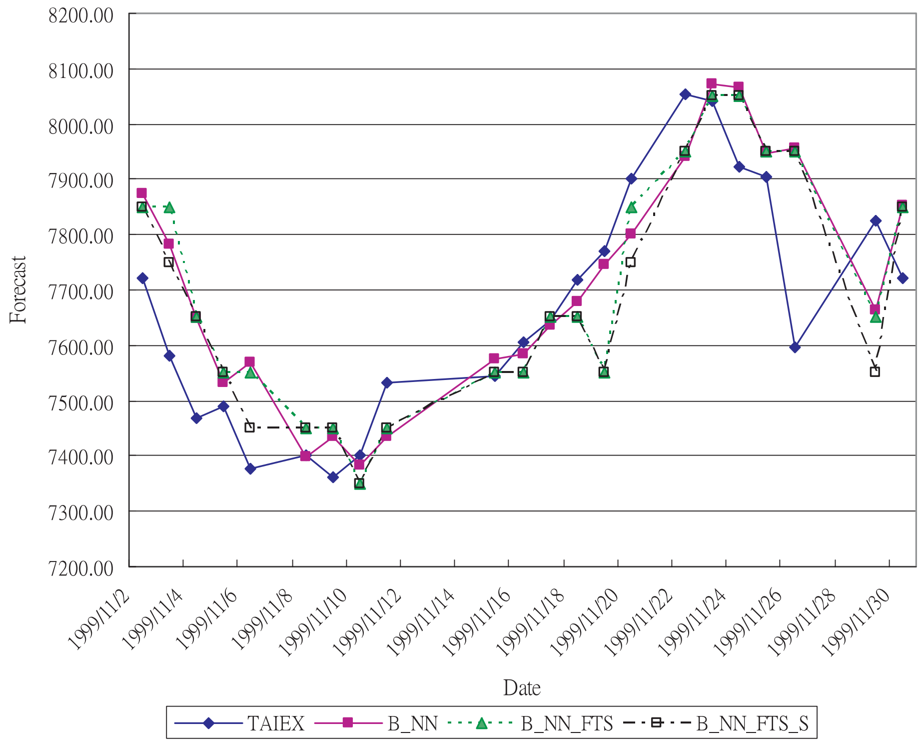

Fig. 4. A comparison of forecasts in November 1999.

in ascending order. Then, we assign a rank from 1 to p accordingly when there are p models for comparison. The average rank is defined as follows:

$$\text{average\_rank} = \frac{\sum_{r=1}^{q} \text{rank}_r}{q}$$

where there are q years for comparison.

Meanwhile, we use the standard deviation of the ranks to evaluate the consistency of relative performance.

5.2. Comparisons

To demonstrate the performance, we conduct the empirical analyses of the following models:

(1) Univariate models
- Chen's fuzzy time series model (U_FTS model) (Chen, 1996)
- conventional regression model (U_R model)
- neural network model (U_NN model)
- neural network-based fuzzy time series model (Huarng & Yu, 2006) (U_NN_FTS model)
- neural network-based fuzzy time series model with substitutes (Huarng & Yu, 2006) (U_NN_FTS_S model)

(2) Bivariate models
- conventional regression model (B_R model)
- neural network model (B_NN model)
- neural network-based fuzzy time series model (B_NN_FTS model)
- neural network-based fuzzy time series model with substitutes (B_NN_FTS_S model)

The RMSEs of these models are compared in Table 6.

In regard to the average RMSE and average rank, the B_NN_FTS_S model outperforms the others, and is followed by the B_R model in second place. As for consistency, the U_R model has with the smallest value 0.0, which means it is the most consistent in terms of relative performance. However, when interpreting the consistency in relation to the RMSE, we know that this model consistently performs the worst. The second best model in terms of consistency is the proposed B_NN_FTS_S model, whose value is 0.9. The B_R model is ranked fifth. Hence, when we compare the average RMSE, average rank, and the consistency, the proposed B_NN_FTS_S model is still the best. The ranks are compared in Fig. 5.

Among the group of univariate models, the U_NN_FTS_S model (Huarng & Yu, 2006) performs the best. Both the U_NN_FTS model (Huarng & Yu, 2006) and the U_NN model outperforms the U_FTS model (Chen, 1996). Meanwhile, the U_R model performs the worst. Among the group of bivariate models, the proposed B_NN_FTS_S model performs the best, while the proposed B_NN_FTS model performs the worst. Except for the proposed B_NN_FTS model, the other bivariate models perform better than all the univariate models. Hence, when applying neural networks to fuzzy time series forecasting, the handling of missing patterns is critical in both univariate as well as bivariate models.

Table 6
Comparison of performance (RMSE)

	1999		2000		2001		2002		2003		2004		Average		
	RMSE	Rank	RMSE	Rank	RMSE	Rank	RMSE	Rank	RMSE	Rank	RMSE	Rank	RMSE	Rank	SD
U_FTS	120	8	176	4	148	7	101	8	74	8	84	5	117.4	6.7	1.8
U_R	164	9	420	9	1070	9	116	9	329	9	146	9	374.2	9.0	0.0
U_NN	107	3	309	8	259	8	78	3	57	6	60	1	145.0	4.8	2.9
U_NN_FTS	109	5	255	7	130	3	84	5	56	4	116	7	125.0	5.2	1.6
U_NN_FTS_S	109	5	152	2	130	3	84	5	56	4	116	7	107.8	4.3	1.8
B_R	103	2	154	3	120	1	77	2	54	3	85	6	98.8	2.8	1.7
B_NN	112	7	274	6	131	5	69	1	52	1	61	2	116.4	3.7	2.7
B_NN_FTS	108	4	259	5	133	6	85	7	58	7	67	3	118.3	5.3	1.6
B_NN_FTS_S	93	1	67	1	128	2	78	3	53	2	67	3	81.1	2.0	0.9

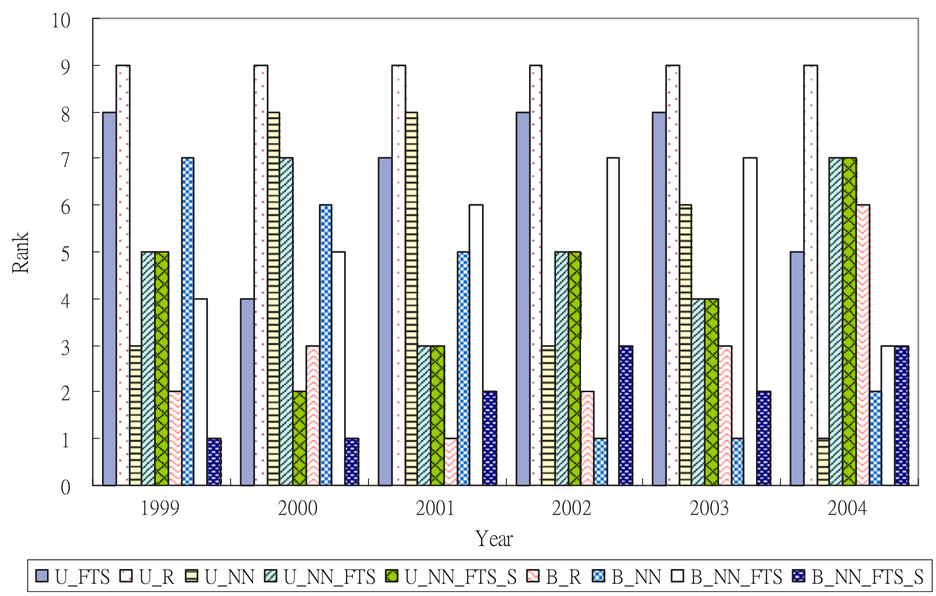

Fig. 5. A comparison of ranks.

When comparing two conventional regression models, the bivariate model greatly outperforms the univariate model. This shows that the TAIFEX is empirically relevant to the TAIEX. Choosing both variables as inputs can assist with forecasting the TAIEX for the next day.

6. Conclusion and future research

This study applies neural networks to fuzzy time series forecasting and proposes two bivariate models to improve forecasting. The stock index and its corresponding index futures are taken as the inputs to forecast the stock index for the next day. As for the average RMSE and the average rank, the proposed bivariate neural network-based fuzzy time series model with substitutes performs the best and the bivariate conventional regression model follows in second place among all the other models. However, the proposed neural network-based fuzzy time series model performs the worst among all the bivariate models. Hence, handling the missing patterns properly is critical to the application of neural networks to fuzzy time series forecasting.

Concerning the consistency of relative performance, the proposed neural network-based time series model with substitutes is ranked second, and the bivariate regression model fifth. However, when we take the average RMSE and the average rank with the consistency into consideration, it again shows that the proposed neural network-based time series model with substitutes performs best.

One future study is to extend the proposed bivariate models to multivariate models. However, some foreseeable problems need to be solved. For example, there will be more empty relationships in the multivariate models than those in the bivariate models. Hence, how to supply proper values to these empty relationships is definitely critical to the forecasting. One solution to that problem may be to propose a distance function to determine a closest LHS, whose RHS can be the values. One way to lessen the impact of this problem is to provide more data. In this case, the selection of proper empirical targets becomes another issue.

Meanwhile, the proposed bivariate models are AR(1) models (autoregressive of order 1), where the value of t is affected by that of $t-1$. If the value of t is affected by the values of $t-1$, $t-2, \ldots, t-p$, it becomes an AR(p)

problem. In this case, the proposed bivariate models can be extended to AR(p) models. However, how to determine the value of p becomes important. The integration of multivariate models with AR(p) is also feasible, but is even more complicated.

Acknowledgements

This work was supported in part by the National Science Council, Taiwan, ROC, under Grant NSC 94-2416-H-035-002. Special thanks are extended to Professor I-Cheng Yeh, Department of Information Management, Chung Hua University, Taiwan, for providing the neural network software, PCNeuron.

References

Chen, S. M. (1996). Forecasting enrollments based on fuzzy time series. *Fuzzy Sets and Systems, 81*, 311–319.

Chen, S. M., & Hwang, J. R. (2000). Temperature prediction using fuzzy time series. *IEEE Transactions on Systems, Man and Cybernetics, Part B, 30*(2), 263–275.

Cybenko, G. (1989). Approximation by superpositions of sigmoidal function, Math.. *Control Signals Systems, 2*, 303–314.

Funahashi, K. (1989). On the approximate realization of continuous mappings by neural networks. *Neural Networks, 2*, 183–192.

Hornik, K. (1991). Approximation capabilities of multiplier feed-forward networks. *Neural Networks, 4*, 251–257.

Hornik, K. (1993). Some new results on neural network approximation. *Neural Networks, 6*, 1069–1072.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks, 2*, 359–366.

Hsu, Y. Y., Tse, S. M., & Wu, B. (2003). A new approach of bivariate fuzzy time series analysis to the forecasting of a stock index. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems, 11*(6), 671–690.

Huang, C. S., Shih, D. H., & Liu, C. L. (1996). Neural network prediction of risk premiums on Taiwan life insurer stocks. *Journal of Information Management, 3*(1), 63–80.

Huarng, K. (2001). Heuristic models of fuzzy time series for forecasting. *Fuzzy Sets and Systems, 123*(3), 369–386.

Huarng, K., & Yu, H. K. (2005). A Type 2 fuzzy time series model for stock index forecasting. *Physica A, 353*, 445–462.

Huarng, K., & Yu, H. K. (2006). The application of neural networks to forecast fuzzy time series. *Physica A, 363*, 481–491.

Hwang, J. R., Chen, S. M., & Lee, C. H. (1998). Handling forecasting problems using fuzzy time series. *Fuzzy Sets and Systems, 100*, 217–228.

Indro, D. C., Jiang, C. X., Patuwo, B. E., & Zhang, G. P. (1999). Predicting mutual fund performance using artificial neural networks. *Omega International Journal of Management Science, 27*, 373–380.

Jang, J.S.R. (1993). ANFIS: Adaptive-network-based fuzzy inference system. *IEEE Transactions on Systems, Man and Cybernetics*.

Kim, J., & Kasabov, N. (1999). HyFIS: adaptive neuro-fuzzy inference systems and their application to nonlinear dynamical systems. *Neural Networks*.

Kwon, C. S., & Shin, T. S. (1999). Cointegration and causality between macroeconomic variables and stock market returns. *Global Finance Journal, 10*(1), 71–81.

Lee, L. W., Wang, L. H., & Chen, S. M. (2007). Temperature prediction and TAIFEX forecasting based on fuzzy logical relationships and genetic algorithms. *Expert Systems with Applications, 33*(3), 539–550.

Lee, L. W., Wang, L. H., & Chen, S. M. (2008). Temperature prediction and TAIFEX forecasting based on high-order fuzzy logical relationships and genetic simulated annealing techniques. *Expert Systems with Applications, 34*(1), 328–336.

Mavrides, M. (2003). Predictability and volatility of stock returns. *Managerial Finance, 29*(8), 46–57.

Nauck, D., & Kruse, R. (1999). Neuro-fuzzy systems for function approximation. *Fuzzy Sets and Systems, 101*(2), 261–271.

Peláez, R. F. (2003). Ten-year forecasts of real stock price changes. *Empirical Economics*, 28417–28429.

Smith, K. A., & Gupta, J. N. D. (2002). *Neural Networks in Business: Techniques and Applications*. Hershey, PA: Idea Group Publishing.

Song, Q., & Chissom, B. S. (1993a). Forecasting enrollments with fuzzy time series – Part 1. *Fuzzy Sets and Systems, 54*, 1–9.

Song, Q., & Chissom, B. S. (1993b). Fuzzy time series and its models. *Fuzzy Sets and Systems, 54*, 269–277.

Song, Q., & Chissom, B. S. (1994). Forecasting enrollments with fuzzy time series – Part 2. *Fuzzy Sets and Systems, 62*, 1–8.

Sullivan, J., & Woodall, W. H. (1994). A comparison of fuzzy forecasting and Markov modeling. *Fuzzy Sets and Systems, 64*, 279–293.

Wasserman, P. D. (1989). *Neural computing: theory and practice*. New York: Van Nostrand Reinhold.

Widrow, B., Rumelhart, D., & Lehr, M. A. (1994). Neural networks: Applications in industry, business and science. *Communications of the ACM, 37*(3), 93–105.

Yu, H. K. (2005a). A refined fuzzy time-series model for forecasting. *Physica A, 346*(3–4), 657–681.

Yu, H. K. (2005b). Weighted fuzzy time series models for TAIEX forecasting. *Physica A*, 349609–349624.

Zhang, G. P. (2004). In G. P. Zhang & Pa. Hershey (Eds.), *Business forecasting with artificial neural networks: an overview, neural networks in business forecasting* (pp. 1–22). Idea Group Publishing.

Zhang, G., Patuwo, B. E., & Hu, M. Y. (1998). Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting, 14*, 35–62.