The Prediction of Taiwan 10-Year Government Bond Yield

KUENTAI CHEN[1], HONG –YU LIN[2]*, TZ–CHEN HUANG

[1]Department of Industrial Engineering and Management,
Mingchi University of Technology
84 Gungjuan Road, Taishan, Taipei Hsien 243,
TAIWAN
kuentai@mail.mcut.edu.tw

[2*]Department of Business and Management
Mingchi University of Technology
84 Gungjuan Road, Taishan, Taipei Hsien 243,
TAIWAN
hylin@mail.mcut.edu.tw

Abstract: Neural Networks are nowadays a promising technique in various financial applications. Numerous studies have demonstrated that the Neural Networks are accurate and efficient. Yet research for the field of forecasting government bond yield is short. Among these limited number of studies, Backpropagation network (BPN) seems to be the most used method. However, suffering form the potential problems, such as slow training speed, long processing time, and possible local minimum, BPN may not be the best solution for all applications in practice.

The purpose of this research is to provide an in-depth study of effects of on the performance of different neural networks in Taiwan's 10-year government bond yields forecasting. Five selected models with different structures, namely Backpropagation network (BPN), Resilient Propagation (RPROP), Radial Basis Function Neural Network (RBFN), Adaptive Neuro-Fuzzy Inference Systems (ANFIS), and Support Vector Regression (SVR), are investigated and the results are analyzed and compared. The results indicate that (1) the number of nodes in the hidden layer is insensitive to the prediction. (2) The recommended number of input nodes is five. (3) More training samples do enhance forecasting performance in our study. (4) The performance of RBFN is the best, followed by ANFIS and RPROP, SVR, and then BPN. (5) BPN is efficient but not the best approach. (6) Our result reveals that RBFN is a useful predicting approach in government bond yield, it performs better than other four models. The recommended structure for RBFN in this application is five input nodes, six center nodes in the hidden layer, and one output node.

Keywords: government bond, yield, forecasting, artificial neural network, neural networks, Radial Basis Function Neural Network

1. Introduction

Bonds and Shares both play important roles in financial markets, but in the beginning most investors lay eyes on Shares rather than Bonds since the later one has problems such as complex pricing, lack of information, and so on. Bonds did not draw enough attention until 1992, when the deal value of the bond market surmounted the stock market for the first time in Taiwan.

The bond valuation is sensitive to interest rates because of its size, liquidity, and lack of credit risk. Therefore, the bond market is often used to pick out changes in interest rates or the shape of the yield curve. As the result, how to predict bond yield precisely becomes a key issue for the investors to avoid risks and to profit.

The traditional research in predicting interest rates apply regression analysis or time series with certain proper model most of the time. Yet the regression model was challenged by Lucas (1976) because of its random model coefficients caused by policy change. Since then the time series models become the major approaches in prediction interest rates, including Autoregressive Integrated Moving Average Model (ARIMA), Autoregressive Conditional Heteroscedasticity Model (ARCH), Generalized Autoregressive Conditional Heteroscedasticity Model (GARCH), and so on. However, it was pointed out by Mandelbrot (1963) and Fama (1965) that the time series models are indeed good in linear problems, but the financial data with Volatility Clustering would cause nonlinearities. Michael et al. (1997) also indicated that by using linear models for nonlinear data would cause the cointegration errors in time series.

Neural Networks (NN) are capable of dealing with nonlinear problems, which has been proven in many researches, such as Tsoukalas and Uhrig (1997), Kuan and White (1994), and Lin and Lee (1996). As a well-developed prediction tool, NN can efficiently identify data structure through learning, if supplied with enough training data. It was a supplement to statistical analysis especially when the normality is not satisfied, and thus it is often applied in financial applications.

Although various NNs are applied to financial predictions, it is the center of attention on stock market, futures, and exchange rate, only a few papers focus on bond yields. For examples, Lapedes and Farber (1987) employed moving average and Back Propagation Neural Network (BPN) to predict Standard and Poor 500 (S&P500) and found the BPN outperformed moving average with 61% accuracy. Kimoto and Asakawa (1990) proposed a buying and selling timing prediction system for stocks on the Tokyo Stock Exchange based on modular neural networks with supplementary learning, which is also a variation of BPN. Their results showed a 67% profit if apply the proposed system. Kosaka et al. (1991) successfully combined BPN and fuzzy logic to construct a bond-trading decision support system, which had great performances in bond selection and price prediction. Bergerson and Wunsch (1991) used BPN and Expert systems to simulate the buying and selling points in futures trading. These literatures are summarized in Table 1, which demonstrates the successful use of NNs and suggests that BPN is the most commonly used neural network.

Nevertheless, the BPNs suffer from some disadvantages: slow training speed, requirement of mass data, and local optimum. Hence a plenty of efforts were made to overcome these problems, such as different structures, various learning schemes, faster searching mechanisms, and combining with other techniques. Resilient Propagation network is one of the attempt to speed up the training, an application can be found in Klinger and Rudolph (2006) as a data mining technique; Radial Basis Function Neural Network has a similar but different network in structure, an example can refer to Awad et al. (2006) as a function approximator; Adaptive Neuro-Fuzzy Inference Systems combined fuzzy logic and neural networks, an example in control and be found in Vasudevan et al. (2003); and Support Vector Regression provides the global optimal solution if applicable, which can be seen in Xiao et al. (2008).

In this research, other than BPN we select several famous approaches for comparison. It is one of the research purposes to argue that the most popular BPN in financial applications may not be the best neural networks to implement.

2 Research Target

The government bonds are usually considered as risk-free bonds and are issued at different time lengths. Taiwan 10-year index bond yield is selected in this research since it is the most traded bond and is sensible to the interest rate change. The sample data are daily 10-year index bond yields from CMoney database, dated from 2002/01/02 to 2006/02/09. As the data are closely related to time, several neural networks are selected to study how much data is sufficient for NN and how many days should be used for inputs. Hence three lengths of training data are used: short-term, medium, long-term. Short-term consists of 239 data points, dated 2005/01/03 to 2005/12/27; medium consists of 489 data points, dated 2004/01/02 to 2005/12/27; long-term consists of 985 data points, dated 2002/01/02 to 2005/12/27. Data ranged from 2006/01/02 to 2006/02/09 are used as testing data.

3 Research methods

Five networks are selected in this research: Backpropagation Neural Networks (BPN), Resilient Backpropagation (RPROP), Radial Basis Function Network (RBFN), Adaptive Neuro-Fuzzy Inference Systems (ANFIS), and Support Vector Regression (SVM). These methods are well-known and thus will only be briefly reviewed below.

The BPN was proposed by Rumelhart et al. (1986) and thereafter raised a series of discussions and successful applications. BPN feed messages forward and propagate errors backward, it consists of three layers: input, hidden, and output layer. Gradient decent methods are usually used for training, which are proven successful but relatively slow.

Riedmiller and Braun (1993) proposed RPROP to improve the training speed. It is a local adaptive learning scheme, performing supervised batch learning in feed-forward NNs. Its structure is similar to BPN but it updates weights according to the sign of gradient. Its advantages are fast convergence and that for many problems the choice of at most one parameter is needed to obtain optimal or nearly optimal convergence times.

Table 1 Some financial applications of Neural networks

Research	Methods	Targets	Results
Lapedes and Farber (1987)	BPN*; Moving average	Standard and Poor 500	61% accuracy
Kimoto and Asakawa (1990)	BPN	stocks on the Tokyo Stock Exchange	67% profit
Kosaka et al. (1991)	BPN+ fuzzy logic	bond	660% profit
Grudnitski and Osburn (1993)	BPN	S&P500, gold future	75% accuracy and 17.04% profit in S&P500; 61% accuracy and 16.36% profit in gold future
Lai et al. (1998)	BPN; Reasoning Neural Network*; Perceptron networks	S&P500	
Refenes and Zaidi (1995)	BPN* Moving average Average value	US dollar/German Mark exchange rate	
Wu (1995)	BPN*; ARIMA	exchange rate	
Zhang and Hu (1998)	BPN*; Random Walk Model		
Cheng et al. (1996)	BPN	U.S. Treasury Bond	
Vanstone and Finnie (2009)	BPN	Stockmarket trading	

Note: Method* represents the method with best performance

Another attempt to speed up the training is Radial Basis Function Network (RBFN), proposed by Moody and Darken (1988). The structure of RBFN is also similar to BPN, but the hidden layer has a variable number of neurons. Each neuron consists of a radial basis function centered on a point with as many dimensions as there are predictor variables. The spread of the RBF function may be different for each dimension. The hidden neuron computes the Euclidean distance of the test case from the neuron's center point and then applies the RBF kernel function to this distance using the spread values. The resulting value is passed to the next layer. RBFN is conceptually similar to K-Nearest Neighbor models and is usually used for function approximation, time series, and pattern classification.

Adaptive Neuro-Fuzzy Inference Systems (ANFIS) was proposed by Jang (1993), it combines structure of neural network and fuzzy inference system in order to acquire strengths form both methods: learning ability and linguistic interpretation. ANFIS maps inputs through input membership functions and associated parameters, and then through output membership functions and associated parameters to outputs. Hybrid Learning Rule and Least Squares Estimate are used in this five-layer feed-forward network. Numerous successful ANFIS applications are reported since it was integrated in MATLAB as a toolbox.

Based on statistical learning theory, Vapnik (1995) developed Support Vector Machines (SVM) to solve the classification problem and fast gained popularity due to many striking features and promising performance in real-world applications. SVM performs classification by constructing an N-dimensional hyperplane that separates the data optimally into two categories. SVM is also close to neural networks when using a sigmoid kernel function, which makes SVM equivalent to a two-layer, perceptron neural network. By introducing an alternative loss function including a distance measure, SVM was then extended to the domain of regression problems (Vapnik et al., 1997), namely Support Vector Regression (SVR). The SVR model has only one optimal solution, which overcomes the best parameter selection problem that many neural networks suffering from.

4 Predictions of the Taiwan 10-Year Government Bond Yield

In this paper, MATLAB Neural Network Toolbox 7.0 was used and Mean Absolute Percentage Error (MAPE) as in EQ (1) is utilized as the error measure.

$$MAPE = \frac{\sum_{k=1}^{T} \left| \frac{Output_k - Target_k}{Output_k} \right|}{T} \quad (1)$$

In the following, all five method are tested in both number of input variables and number of nodes in hidden layer for all short-term, medium-term, and long-term data.

4.1 Backpropagation Neural Networks

The effects of node number in hidden layer, number of input nodes, and length of data were tested in this study. The results for node number in hidden layer are shown in Figure 1, which suggests that there is no trend for all lengths of data but at least 5 nodes in hidden layer will be required. As shown in Table 2, the best MAPE is 0.732721% with 6 nodes in hidden layer and one input variable, using long-term data. It can be also found that the predictions using long-term data are always the best.

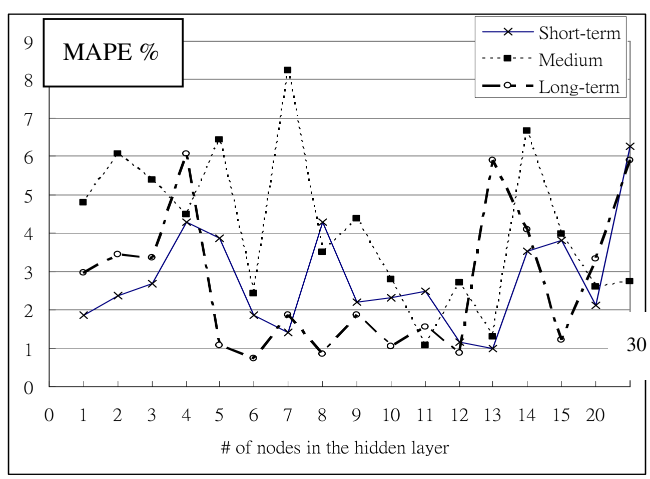

Fig. 1 The effect of node number in hidden layer

Table 2 MAPEs by the BPN with different number of input variables

Number of input nodes	Short-term	Medium	Long-term
1	0.994522	1.069324	0.732721
2	1.211741	1.298624	0.752394
3	1.592723	2.069717	0.868749
4	1.216647	1.049198	0.826493
5	1.46241	1.097041	0.951471

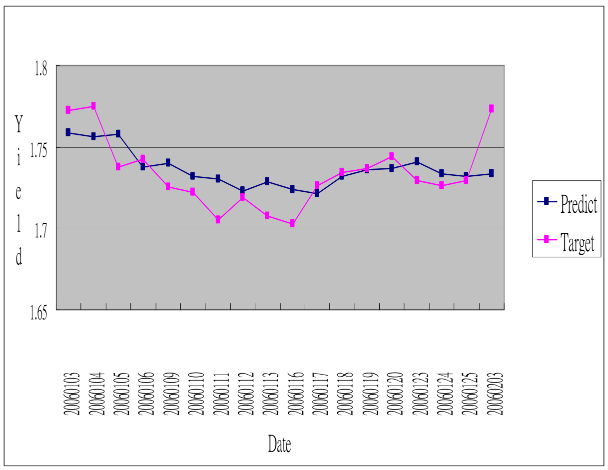

Fig. 2 The predictions of BPN

The BPN predictions of testing data followed the targets closely, as shown in Figure 2.

4.2 Resilient Backpropagation

The results form RPROP are shown in Table 3 and Table 4. As shown in Table 3, the predictions using long-term data are also the best; and four or five input nodes will be better. From Table 4, the best prediction has MAPE of 0.614742%, trained by 4 input variables with 8 nodes in hidden layer using long-term data. Figure 3 is the predictions of RPROP with the best setting, which follow the targets well.

4.3 Radial Basis Function Neural Network

Table 5 shows that the best result for RBFN is to use 5 input variables in long-term data, with the smallest MAPE of 0.573541%. Both Table 5 and Table 6 show that using long-term data for training outperformed short-term and medium-term greatly. Table 6 shows MAPEs from different number of center points in hidden layer. The RBFN with 5 input nodes and 4 center points has the best performance. Figure 4 shown the RBFN predictions, which also followed the targets well.

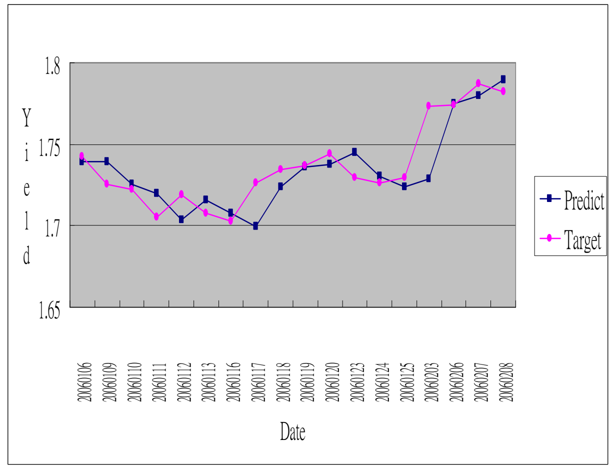

Fig. 3 The predictions of RPROP

Table 3 MAPEs by the RPROP with different number of input variables

# of input node	Short-term	Medium	Long-term
1	1.051342	0.77333	0.838759
2	0.943173	0.918644	0.766941
3	1.13362	0.897183	0.707054
4	**0.745159**	0.883122	**0.617739**
5	0.83512	**0.766729**	0.648289

Table 4 MAPEs by the RPROP with different node number in hidden layer

Node #	Short-term	Medium	Long-term
2	1.445996	0.939775	1.684356
5	1.288035	0.961748	0.749682
6	**0.745159**	1.100684	0.810669
8	0.764468	0.917146	**0.614742**
9	0.991137	**0.827659**	0.649393
10	1.139156	0.883122	**0.617739**

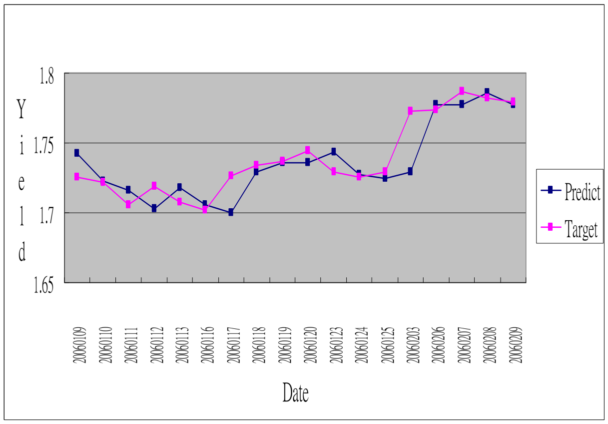

Fig. 4 The predictions of RBFN

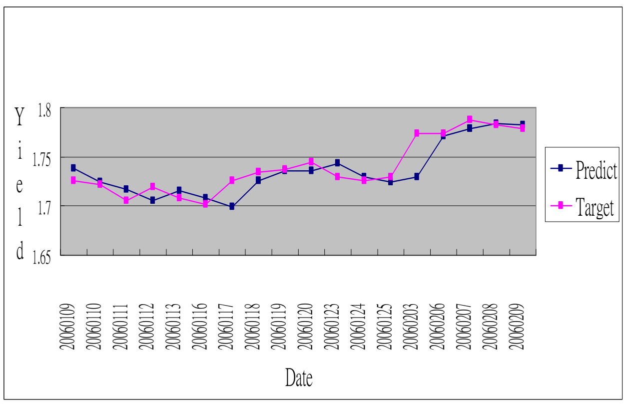

Fig.5 The predictions of ANFIS

Table 5 MAPEs by the RBFN with different number of input variables

# of input node	Short-term	Medium	Long-term
1	**0.72751**	**0.673619**	0.709762
2	0.768373	0.768381	0.705031
3	0.856192	0.814516	0.723565
4	0.795819	0.722295	0.607644
5	0.863237	0.846556	0.573541

4.4 Adaptive Neuro-Fuzzy Inference Systems

In ANFIS, number of input variables and different membership functions are tested. Table 7 shows the results of different number of input variables, and ANFIS with 5 input variables performed the best with MAPE of 0.587416%. Training with long-term data is still the best. The effects of membership functions are displayed in Table 8, which shows to use long-term data is always the best and triangular membership function is the best membership function to use. Figure 5 shows a pretty good prediction of ANFIS versus the targets.

Table 6 MAPEs by RBFN with different number of center points

Center points	Short-term	Medium	Long-term
1	7.053385	6.739566	1.050759
2	2.151029	6.739566	0.890843
3	0.880337	1.768969	0.743576
4	1.157439	0.907255	**0.573541**
5	1.186712	**0.846556**	0.58282
6	1.030881	0.916627	0.576433
7	1.00767	1.005779	0.622744
8	1.01975	1.160595	0.62931
9	0.948954	1.160595	0.618882
10	**0.863237**	1.368612	0.634793

Table 7 MAPEs by the ANFIS with different number of input variables

# of input nodes	Short-term	Medium	Long-term
1	**0.68486**	**0.711657**	0.711514
2	0.688041	0.715003	0.704385
3	0.712211	0.87406	0.736343
4	0.709507	0.802542	0.633319
5	0.859628	0.99472	0.587416

Table 8 MAPEs by the ANFIS, different membership functions

Membership function	Short-term	Medium	Long-term
Triangular	0.888282	1.199581	**0.587416**
Trapezoid	1.564587	1.234724	0.716474
Bell shape	**0.753024**	**0.99472**	0.678991
Gaussian	0.859628	1.080076	0.624453
Two- Gaussian	0.895636	0.996379	0.689922
π	0.996252	1.122318	0.689974
dsigmf	1.05753	1.446943	0.818627
psigmf	1.05753	1.44663	0.818627

4.5 Support Vector Regression (SVR)

In SVR, we used Grid Algorithm with 5-fold cross-validation to search the best combination of parameters. It can be concluded that 4 or 5 input variables with long-term data is the best for SVR, as shown in Table 9. In Table 9, it can be clearly seen that long-term data still performed better. Figure 6 is the predictions of SVR, which also followed the targets well.

Table 9 MAPEs for SVR with different number of input nodes

# of input nodes	Short-term	Medium	Long-term
1	**0.7502**	0.8099	0.713459
2	0.8580	**0.7696**	0.708588
3	0.9293	0.8620	0.738637
4	0.761001	0.772844	**0.633817**
5	0.840263	0.779042	0.635723

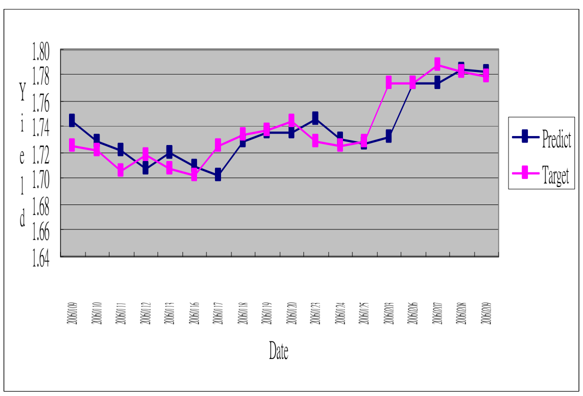

Fig. 6 The predictions of SVR

4.6 Comparison

These five networks are all valid in predicting Taiwan 10-year government bond yield. To compare these five networks, the results MAPEs by the best settings for each network are shown in Table 10. It is obvious that long-term data should be used for prediction at all circumstances, as their MAPEs are always the smallest. Among these five networks, RBFN has the best performance using medium and long-term data and is the second best for short-term data. BPN is the worse one but can still predict efficiently. The overall predictions of all five methods and the targets are displayed in Figure 6.

Table 10 Comparison of five networks with best settings of each network

Network	Short-term	Medium	Long-term
BPN	0.994522	1.049198	0.732721
RPROP	0.745159	0.766729	0.617739
RBFN	0.72751	**0.673619**	**0.573541**
ANFIS	**0.68486**	0.711657	0.587416
SVR	0.75020	0.76960	0.633817

5 Conclusion and Discussion

In this research, five networks are employed to predict Taiwan 10-year Government bond yield. Number of input variables, which in this case representing how many days should be included in the model, and number of hidden layers are tested. The lengths of training data are also tested by short-term, medium-term, and long-term data. The results indicate that (1) the number of nodes in the hidden layer is insensitive to the prediction. (2) The recommended number of input nodes is five. (3) Obviously, more training samples do enhance forecasting performance in our research. To further confirm this conclusion, we extended the data and examined it again. The result is showed in Table 11. In Table 11, it can be seen that (a) RBFN is still the best, and (b) result from long-term data did not significantly differ from results from more data. That is, 985 days should be enough for training. (4) The performance of RBFN is the best, followed by ANFIS and RPROP, SVR, and then BPN. (5) BPN is efficient but not the best approach. (6) Our result reveals that RBFN is a useful predicting approach in government bond yield, it performs better than other four neural network models. The recommended combination of parameters for RBFN is five input nodes, six center nodes in the hidden layer, and one output node.

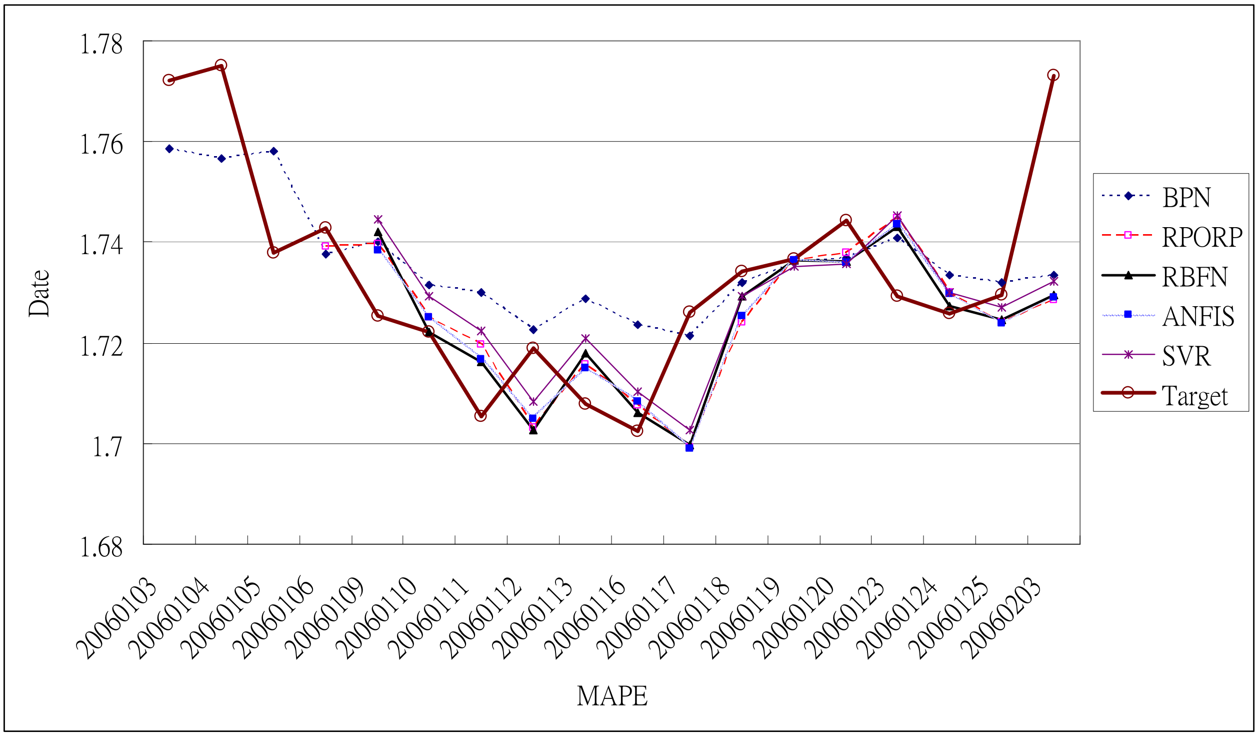

Fig. 6 The outputs of five networks and targets

Table 11 The MAPEs of networks with extended data

4 input nodes				
Training data	BPN	RPROP	RBFN	ANFIS
Short-term (239)	1.216647	0.745159	0.795819	0.709507
Medium (489)	1.049198	0.883122	0.722295	0.802542
Long-term (985)	0.826493	**0.617739**	**0.607644**	**0.633319**
1006	**0.695866**	0.62746	0.611814	0.635942
1028	0.701348	0.65449	0.613842	0.637247
5 input nodes				
Training data	BPN	RPROP	RBFN	ANFIS
Short-term (239)	1.46241	0.83512	0.863237	0.859628
Medium (489)	1.097041	0.766729	0.846556	0.99472
Long-term (985)	0.951471	0.648289	0.573541	0.587416
1006	1.148432	0.612411	**0.563666**	0.585474
1028	**0.903916**	**0.596415**	0.580989	**0.583324**

Reference:

[1] Awad, M., Pomares, H., Rojas, I., Herrera, L.J., Guillen, A., Valenzuela, O. (2006) "Hierarchical Structure for function approximation using Radial Basis Function" WSEAS Trans. on Mathematics, Issue 1 Vol 5, January 2006 pp75

[2] Bergerson, K. and D. C. Wunsch (1991). "A Commodity Trading Model Based on a Neural Network-Expert System Hybrid." IJCNN-91, Vol.I, pp.289-293.

[3] Cheng W, Wanger L, Lin CH. Forecasting the 30-year US treasury bond witha system of neural networks. Journal of Computational Intelligence in Finance 1996;4:10–6.

[4] Fama, E. F. (1965). "Mandelbrot and the Stable Paretian Hypothesis." Journal of Business, Vol.38, pp.34-105.

[5] Grudnitski, G. and L. Osburn (1993). "Forecasting S&P and Gold Future Prices : An Application of Neural Networks." The Journal of Futures Markets, Vol.13, No.6, pp.631-643.

[6] Jang, J. S. R. (1993). "ANFIS: Adaptive Network-based Fuzzy Inference System." IEEE Transactions on Systems, Man, and Cybernetics, Vol.23, No.3, pp.665-685.

[7] Kimoto, T. and K. Asakawa (1990). "Stock Market Predication System with Modular Networks." IJCNN-90-Wash, Vol.I, pp.1-6.

[8] Klinger, Roman and Rudolph, Gunter (2006), "Evolutionary Composition of Music with Learned Melody Evaluation", Proceedings of the 5th wseas int. conf. on Computational intelligence, Man-machine systems and Cybernetics, Venice, Italy, November 20-22, 2006 pp234

[9] Kosaka, M., Mizuno, H., T. Sasaki, R. Someya, and N. Hamada (1991). "Applications of Fuzzy logic/Neural Network to Securities Trading Decision Support System." IEEE, pp.1913-1919.

[10] Kuan, C. and H. White (1994). "Artificial Neural Networks: An Econometric Perspective." Econometric Reviews, Vol.13, No.1, pp.1-91.

[11] Lai, C.C., Hsu, Y. and R. Tsaih (1998). "Forecasting S&P 500 stock index futures with a hybrid AI system." Decision Support Systems, Vol.23, pp.161-174.

[12] Lapedes, A. and R. Farber (1987). "Nonlinear Signal Processing Using Neural Networks: Prediction and System Modeling." Los Alamos National Laboratory Report.

[13] Lin, C. T. and C. S. G. Lee (1996). Neural Fuzzy System. New Jersey：Prentice-Hall Inc.

[14] Lucas, R. E. (1976). "Econometric Policy Evaluation：A Critique." Carnegie-Rochester Conference Series on Public Policy, Vol.1, pp.19-46.

[15] Mandelbrot, B. (1963). "The Variation of Certain Speculative Prices." Journal of Business, Vol.36, pp.394-419.

[16] Michael, P., Nobay, A. R. and D. A. Peel (1997). "Transactions Costs and Nonlinear Adjustment in Real Exchange Rates: An Empirical Investigation." Journal of Political Economy, Vol.105, No.4, pp.862-879.

[17] Moody, J. and C. Darken (1988). "Learning with Localized Receptive Fields." Proceedings of the 1988 Connectionist Models Summer School, Carnegie Mellon University, Pittsburgh, PA. San Mateo, CA Morgan Kaufmann Publishers.

[18] Refenes, Apostolos-Paul and A. Zaidi (1995). "Managing Exchange Rate Prediction Strategies with Neural Networks.", Neural Networks in the Capital Markets, Chichester, England, John Wiley and Sons, pp.213-219.

[19] Riedmiller, M. and H. Braun (1993). "A Direct Adaptive Method for Faster Backpropagation Learning: The RPROP Algorithm." In Proceedings of the IEEE International Conference on Neural Networks, pp.586-591.

[20] Rumelhart, D. E., Hinton, G. E. and R. J. Williams (1986). "Learning Representations by Back-Propagating Errors." Nature, Vol.323, pp.533-536.

[21] Tsoukalas, L. H. and R. E. Uhrig (1997). Fuzzy

and Neural Approaches in Engineering. New York：Wiley and Sons.

[22] Vapnik. (1995) The Nature of Statistical Learning Theory. Springer, N.Y., 1995.

[23] Vapnik, V. Golowich, S. and Smola. A., (1997) "Support vector method for function approximation, regression estimation, and signal processing". Advances in Neural Information Processing Systems 9, pages 281–287, Cambridge, MA, 1997. MIT Press.

[24] Vanstone, B. and Finnie, G. (2009) **"**An empirical methodology for developing stockmarket trading systems using artificial neural networks", Expert Systems with Applications, Vol. 36, Issue 3, Part 2, April 2009, Pages 6668-6680

[25] Vasudevan, M., Arumugam, R., Venkatesan, G. , Paramasivam, S. (2003) "Estimation of stator resistance of induction motor for direct torque control scheme using adaptive neuro-fuzzy inference systems" WSEAS Trans. on Circuits and Systems, Issue 2, Vol 2, April 2003 pp327

[26] Wu, B. (1995). "Model-free Forecasting for Nonlinear Time Series: with Application in Exchange Rates." Computational Statistics and Date Analysis, Vol.19, No.4, pp.433-459.

[27] Xiao, H., ZHANG, X., Du, Yumei (2008) "A Comparison of Neural Network, Rough Sets and Support Vector Machine on Remote Sensing Image Classification", 7th WSEAS Int. Conf. on Applied computer & Applied computational science (ACACOS '08), Hangzhou, China, April 6-8, 2008 pp597

[28] Zhang, G. and M. Y. Hu (1998). "Neural Network Forecasting of the British Pound/US Dollar Exchange Rate." Journal of Management Science, Vol.26, No.4, pp.495-506