Comparison Between Artificial Neural Network and Neuro-Fuzzy for Gold Price Prediction

Conference Paper · August 2013

DOI: 10.1109/IFSC.2013.6675635

2 authors:

Mahsa Kangarani Farahani
Isfahan University of Technology

Soheil Mehralian
Khaje Nasir Toosi University of Technology

Comparison Between Artificial Neural Network and Neuro-Fuzzy for Gold Price Prediction

Mahsa KangaraniFarahani[1] , Soheil Mehralian[2]
Department of Electrical and Computer Engineering
Isfahan University of Technology
Isfahan, Iran
E-mail: m.kangaranifarahani[1], s.mehralian[2] @ec.iut.ac.ir

***Abstract*— This article presents a comparison of Artificial Neural Network (ANN) and Adaptive Neural Fuzzy Inference System (ANFIS) for predicting a real system, gold price. Also, we compared a new hybrid model which is a weighted average of the ANN and ANFIS model. The main objective is to predict the gold price in the Forex market. We used two prediction machine models in ANN, a model which feeds back the network output as input and another model that does not do it. Our results show that the performance error of the former model is more than the latter, and also the performance of ANFIS is better than both models of ANN. To evaluate the methods three performance measurements are used: Root Mean Squared Error (RMSE), percentage error and Mean Tendency Error (MTE) which is proposed in this study. The strength point of our method is the prediction machine model that is one of the most powerful prediction machine models of ANN. At last, a Wavelet denoising algorithm is applied to the data, but due to the chaotic structure of the gold price, it impairs data and causes to reduce the performance of prediction result.**

Keywords*— *Artificial Neural Network; Neuro-Fuzzy; time series; prediction; Forex market.

I. INTRODUCTION

The purpose of this paper is to compare two tools for gold price prediction that become very popular in the last decades: Artificial Neural Network and Neuro-Fuzzy System.

The main objective of the tools is to estimate the gold price using price history that depends on various economic factors. Price prediction plays important role in economic decision making and may be used in numerous application. For example, to earn income from speculative activities, to determine optimal government policies, or to make business decisions [1].

Up to now, several studies have been performed to predict the stock market. There are three main categories of prediction methods: (1) Classical methods. (2) Artificial Intelligence methods. (3) Hybrid approaches.

Before emersion of ANN, classical methods were in fashion and have been used in many applications. These methods were linear such as AutoRegressive Integrated Moving Average (ARIMA) and Multiple Regression.

ANN is a nonlinear method that is widely used in a variety of industries, businesses and sciences. One of the most important applications of ANN is time series prediction [2]. ANN is a powerful tool for stock market prediction [3]. The characteristics of ANN that make it appropriate for prediction are its nonlinear structure, flexibility, data-driven learning process and its ability to estimate universal functions.

It has been proven that ANN can estimate any continuous nonlinear functions with desired accuracy [4]. Recently, researchers have shown that ANNs can be combined with other methods, to improve the accuracy, which eventuate the hybrid systems Several models of hybrid systems have been used for gold price prediction such as the models that is a combination of ANN and ARIMA, and combination of two ANNs (one for prediction and the other for modification) [3]. Another hybrid system is called Neural Fuzzy system which combines Fuzzy and ANN. Nowadays, ANNs and Fuzzy are widely used for time series prediction.

In this paper, ANN and ANFIS approaches are compared to predict gold price. In addition to the comparison, a new hybrid model also is proposed that have comparable performance with ANN and ANFIS. To collect gold prices we used Meta Trader software which is a platform for gathering Forex market data. The used structures of ANN and ANFIS models are described in the next section.

The paper is organized as follows. In section 2, describe two compared method, ANN and Neural Fuzzy, and their structure for time series prediction. In section 3, present the ANN and Neural-Fuzzy models that implemented in this paper. In section 4, present the performance measurement. Finally, conclusions are discussed in section 5.

II. COMPARED METHODS

A. Artificial Neural Network Structure

It is no longer than few decades than ANN proposed, but because of some characteristics such as parallel processing, intelligence and flexibility it becomes a very important tool for complex problems in pattern recognition, clustering and time series prediction [5].

ANNs act like a brain; it means that ANN first spend some times for learning then it can be used in practice. The more

978-1-4799-1228-5/13/$31.00 ©2013 IEEE

complete and detailed observations, the more accuracy and output. Although, some observations may be misleading. Therefore, what is given to the ANN must be as right as possible. Observation may have labels or not, but in time series prediction they must be used in labeled form.

One of the most common types of ANN is Multilayer Perceptron (MLP) with backpropagation algorithm. Theoretically it is proved that by selecting an appropriate internal structure for ANN, it would be able to model and simulate any nonlinear system.

The most widely used models for time series modeling is Single hidden layer Feed Forward Network [5]. In Feed Forward Networks the output of each neuron or layer is forwarded to input of next neuron or layer. A Multilayer Feed Forward Network is shown in Fig. 1.

The perceptron is an algorithm for supervised classification that was invented in 1957 by Frank Rosenblatt [6]. The perceptron is a binary classifier which maps its input x, to an output value $f(x)$.

$$f(x) = \begin{cases} 1 & if \quad w.x + b > 0 \\ 0 & otherwise \end{cases} \tag{1}$$

Where w is a vector of real-valued weights and b is the bias, a constant term that does not depend on any input value. Rosenblatt presented learning algorithm steps as below.

1. Initialization: Initialise weights and threshold. Note that weights may be initialized by setting each weight node to 0 or to a small random value.

2. For each sample i in our training set, perform the following steps over the input x_i and desired output t_i :

2a. Calculate the actual output:

$$O_i(l) = f[w(l).x_i] \tag{2}$$

2b. Adapt weights:

$$w_i(l+1) = w_i(l) + \Delta w_i(l) \tag{3}$$

Weights are modified according the delta rule, where the delta rule is as follows:

$$e(l) = t_i - O_i(l) \tag{4}$$

$$\Delta w_i(l) = \alpha . x_i(l).e(l) \tag{5}$$

Step 2 is repeated until the iteration error becomes less than a user-specified error threshold.

In this paper we have used backpropagation algorithm in layers of the Feed Forward Network. Backpopagation is a supervised learning algorithm; hence we must create a network with inputs and desired outputs. The idea behind of backpropagation algorithm is to minimize the difference between network output and the desired output.

The activation function of backpropagation algorithm is used to activate neurons.

$$A_j(\overline{p}, \overline{w}) = \sum_{i=0}^{n} p_i w_{ji} \tag{6}$$

Activation function only depends to the inputs and weights of neurons. If output function and activation function be same, the network is called linear network that has a lot of limitations. Most common output functions are Sigmoid function:

$$O_j(\overline{p}, \overline{w}) = \frac{1}{1 + e^{A_j(\overline{p}, \overline{w})}} \tag{7}$$

The purpose of the learning process is to generate desired output from the input. Because the impact of weights of neurons on the network error, we must adapt weights to minimize the error. The error for each neuron is calculated in the form of Equation (8).

$$E_j(\overline{p}, \overline{w}, t) = (O_j(\overline{p}, \overline{w}) - t_i)^2 \tag{8}$$

The network error is equal to the summation of all errors of output layer neurons that is shown in Equation (9).

$$E_j(\overline{p}, \overline{w}, \overline{t}) = \sum_j (O_j(\overline{p}, \overline{w}) - t_i)^2 \tag{9}$$

The parameter w_{ji} of neural network, are changed by a gradient descent method that is shown in Equation (10).

$$\Delta w_{ji} = -\eta \frac{\partial E}{\partial W_{ji}} \tag{10}$$

The Backpropagation algorithm has high time complexity, for this reason it is not common in networks with many layers [7].

In our prediction machine model (Fig. 2) we have used a prediction machine (ANN) with input i and output o. This machine after learning process is ready to observe i samples of time series and predict immediately o next samples.

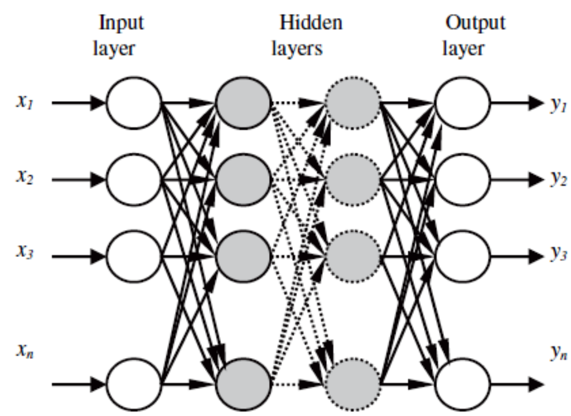

Fig. 1. A multilayer feed forward network

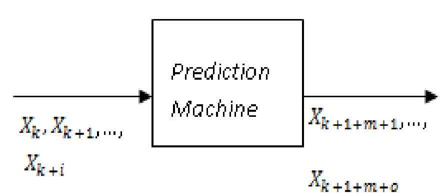

Fig. 2. Prediction machine model

B. Neural Fuzzy Structure

A Neural Fuzzy system consists of Artificial Neural Network and Fuzzy system. The ANN and Fuzzy system have their advantages and disadvantages that the aim of combining these two approaches is using the advantages of two approaches. For example The ANN models are very efficient in adapting and learning, but on the other hand, they have some negative attributes, they are black box [8]. Fuzzy logic has the ability to express the ambiguity of human thinking and translate expert knowledge into computable numerical data. The derivation of if-then rules and corresponding membership functions depends, a lot, on the a priori knowledge about the system. However there is no systematic way to transform experiences and knowledge of human experts to the knowledge base of a fuzzy inference system. There is also a need for adaptability or learning algorithms to produce outputs Whereas a combination of ANN and fuzzy system that is called Neural-Fuzzy system is able to eliminate the basic problem in fuzzy system design (generating a set of fuzzy if-then rules) by using the learning capability of an ANN for automatic fuzzy if-then rule generation and parameter optimization.

There are the different types of fused Neural-Fuzzy systems such as Adaptive Neural Fuzzy Inference System (ANFIS), Fuzzy Inference and Neural Network in Fuzzy Inference Software (FINEST).

ANFIS is a Multilayer Feed Forward Network with a supervised learning scheme, which makes the model of given dataset based on Takagi-Sugeno inference system. Suppose that the rule base contains the following two Sugeno-type fuzzy if-then rules:

Rule1: if x is A_1 and y is B_1 then $f_1=p_1x+q_1y+r_1$

Rule2: if x is A_2 and y is B_2 then $f_2=p_2x+q_2y+r_2$

Where x and y are the inputs, A_i and B_i are the fuzzy sets, f_i is the output, and $\{p_i, q_i, r_i\}$ are the parameters that are determined during the training process.

The five-layer ANFIS structure consists of fuzzification, inference, normalization, consequent, and output, as is presented in Fig. 3.

The ANFIS structure is described as follows [9]:

Layer1: Each node in first layer generates fuzzy membership grades that set by fuzzy membership function (MF). Let O^1_i be the output of node i, and layer 1.

$$O_i^1 = \mu A_i(x) \qquad (11)$$

Where x is the input of node i, and A_i is the linguistic labels of MF. In fact, O^1_i is the MF for A_i In general, μA_i is given by Bell functions Equation (12), where maximum value is 1, and minimum value is 0.

$$\mu A_i(x) = \frac{1}{1+[(\frac{x-c_i}{a_i})^2]^{b_i}} \qquad (12)$$

If changed the parameters values $\{a_i, b_i, c_i\}$ the value of MF would be changed too. These must be adopted in learning steps.

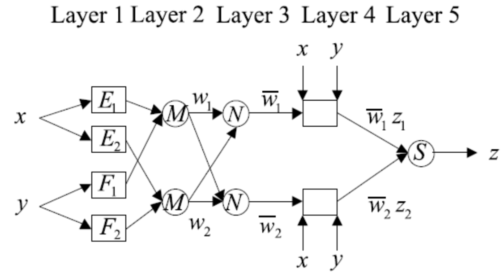

Fig. 3. ANFIS structure

Layer 2: Each node in the second layer calculates the firing strength of each rule via multiplication. In fact, the relationships between linguistic labels are determined.

$$O_i^2 = w_i = \mu A_i(x) \times \mu B_i(y), \qquad i = 1,2. \qquad (13)$$

Layer 3: The firing strength for each rule is normalized.

$$O_i^3 = \overline{w_1} = \frac{w_i}{w_1 + w_2}, \qquad i = 1,2. \qquad (14)$$

Layer 4: In this layer, parameters are called consequent parameters and every node computes the contribution of ith rule towards the overall output.

$$O_i^4 = \overline{w_i} f_i = \overline{w_i}(p_i x + q_i y + r_i) \qquad (15)$$

Layer 5: Finally, the single node calculates the overall output that is the summation of all rules.

$$O_i^5 = overalloutput = \sum_i \overline{w_i} f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \qquad (16)$$

Consequently, an Adaptive Network Fuzzy Inference System is developed, so we need to an adaptive learning algorithm for training the network. ANFIS uses the hybrid-learning algorithm, which is the combination of Gradient Descent and Least-Squares methods.

One of the most important applications of ANFIS is in time series prediction. We want to use time series until time t and predict the next time at $t+P$; P parameter is predicting interval. In time series prediction with ANFIS we must predict the next value by using of D past data. The appropriate parameters D, P achieved by test and error.

An ANFIS structure that used for time series prediction is shown in Fig. 4. In this network exist 5 inputs and each input has 6 membership functions. We want to estimate time $t+P$, so this network must have one output.

III. Implementation

In this study, the gold prices are from Forex market. This data set consists of the prices that closed at any hour per ounce from Sep. 17, 2010 to Jan. 21, 2011. One of the most important steps in developing a successful prediction model is the selection of the input variables, which determines the architecture of the model. So to do this, several network architectures should be tried.

We divide the essential parameters in the ANN structure into two classes: input parameters and output parameters. The

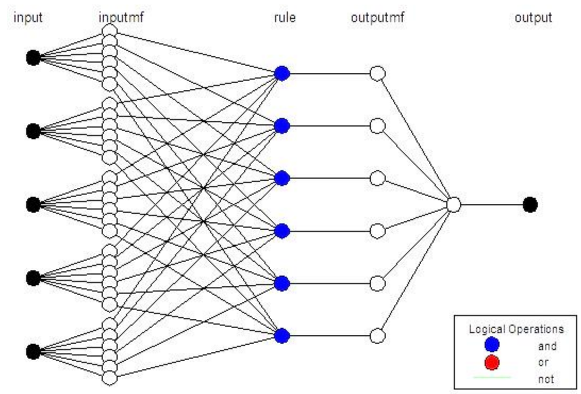

Fig. 4. An ANFIS structure for time series prediction

input parameters are such as the number of neurons, the numbers of hidden layers, the number of input data, type of learning algorithm, transfer function and network type. The output parameters are such as the performance errors, the number of learning epochs.

To have a nonlinear decision boundry, each neuron should have a nonlinear function. For this reason, in this study hyperbolic tangent sigmoid transfer function is used.

In the MLP network with backpropagation algorithm the signals are sent forward and errors feeds back. In this way, the mean squared error between network output and desired output will be minimized by gradient descent method. There are various learning algorithms such as Levenberg-Marquardt, Gradient Descent, Gradient Descent Momentum, etc. The Gradient Descent algorithm may trap in local minimum; to avoid this problem an increasing momentum which is also known as Gradient Descent Momentum algorithm can be used. In this study, we used Gradient Descent Momentum algorithm for this purpose.

In ANN model, we used a backpropagation feedforward network that has 15 input neurons in the first layer, 3 hidden layers and one output neuron, in abbreviated form we will show this network as (15-3-1). Performance error is reduced by increasing the input data set but the learning time is increased. So we should trade-off between time and accuracy.

The input and desired output structure are as follows:

Input:
$$\begin{bmatrix} I_1 & I_2 & \cdots & I_{11} \\ I_2 & I_3 & \cdots & I_{12} \\ \vdots & \vdots & \ddots & \vdots \\ I_{30} & I_{31} & \cdots & I_{40} \end{bmatrix}$$

Desired output:
$$\begin{bmatrix} I_{31} & I_{32} & \cdots & I_{41} \end{bmatrix}$$

The advantage of our approach is in the input structure used for prediction machine model. We tested another model that feedback current network output as input of the next prediction epoch. The data structure of this model is described as below.

Input:
$$\begin{bmatrix} I_1 & I_2 & I3 & \ldots & I_{10} \\ \vdots & \vdots & \vdots & & \vdots \\ & & P_{31} & \ddots & P_{39} \\ I_{30} & P_{31} & P_{32} & & P_{40} \end{bmatrix}$$

Network output:
$$\begin{bmatrix} P_{31} & P_{32} & P_{33} & \cdots & P_{41} \end{bmatrix}$$

Desired output:
$$\begin{bmatrix} I_{31} & I_{32} & I_{33} & \cdots & I_{41} \end{bmatrix}$$

In Neural-Fuzzy model, an ANFIS network with 5 inputs, 1 output and 6 fuzzy rules are used. The type of membership functions in the input is Gaussian function and it is linear in the output. Our ANN output, ANFIS output and desired output are plotted in Fig. 5.

In order to get a more reliable prediction, a hybrid model is proposed, which it weighted the output of both ANN and ANFIS prediction model based on their training performance , then generate the hybrid output by the weighted average of both models. This model can be an effective way to improve the prediction performance of each individual model. The results of hybrid model are compared with other metheds in Fig. 5.

We also applied a Wavelet Denoising algorithm on gold prices to have a robust prediction; but due to the chaotic structure of the gold market, it impairs data and causes to reduce the performance of prediction result. The results are described in the next section.

IV. PERFORMANCE

In order to select the best models for prediction, we evaluated each model's performance individually. In this paper, we used three performance measurements: root mean squared error (RMSE), percentage error, Mean Tendency Error (MTE). These measures are calculated by following equation.

Equation (17) defines the RMSE measure.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(t(i)-a(i))^2}, \tag{17}$$

Where *t(i)* is desired output and *a(i)* is network output.

Percentage error is defined in Equation (18).

$$Per = \frac{1}{n}\sum_{i=1}^{n}100 * \frac{|t(i)-a(i)|}{t(i)} \tag{18}$$

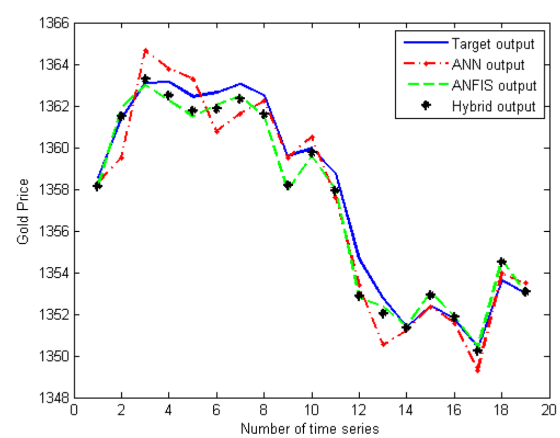

Fig. 5. Prediction Results of ANN, ANFIS and Hybrid model

Market analysis has proven that knowing whether the price is going up or down is enough to buy or sell. In this study, we presented a new performance measure called mean tendency error (MTE) that is based on market analysis and is very applicable in market prediction. MTE measure is shown in Equation (19).

$$MTE = \frac{1}{n}\sum_{i=1}^{n} TE_i, \tag{19}$$

Where TE_i is equal to 0 if $\frac{t_i - t_{i-1}}{a_i - a_{i-1}} \geq 0$, otherwise TE_i is 1.

Finally, we selected the best models of ANN and ANFIS based on these performance measurements, average results are in TABLE I. The results of TABLE I show that the ANFIS and hybrid models are better than ANN. All three models are tested with the denoised data in a same condition and their performance are shown in TABLE I. The results show that noise reduction reduces the performance of gold price prediction, but MTE is better. According to the result, ANN model is more robust against to the denoising that leads to have impared data.

V. CONCLUSION

In this study, a comparison between ANN and ANFIS for gold price prediction is presented. The gold price data are from Forex market and we used 20 of them to test algorithms. Three performance measurements, including RMSE, MTE, and percentage error are used for analysis and evaluate performances of the models. The results show that the ANFIS and ANN methods are both powerful tools for modelling the gold price and ANFIS is a little better and robust than ANN. Our hybrid method in addition to comparable accuracy with ANN and ANFIS has more reliable results. Also, we used a wavelet denoising algorithm for data, and our results showed that ANN is more robust against impaired data and the other are not as compatible as ANN with the denoising.

TABLE I. PREDICTION PERFORMANCE OF ANN, ANFIS AND HYBRID MODELS

Model	**Prediction Performance Measurement**		
	RMSE	***Per***	***MTE***
ANN	2.62	0.14	0.49
ANFIS	2.52	0.13	0.46
Hybrid model	2.54	0.13	0.47
ANN (denoised)	2.47	0.13	0.18
ANFIS (denoised)	2.77	0.15	0.30
Hybrid (denoised)	2.72	0.15	0.30

REFERENCES

[1] Z. Ismail, A.Yahya, A. Shabri, "Forecasting Gold Prices Using Multiple Linear Regression Method". American Journal of Applied Sciences, 2009, vol. 6(8), pp. 1509-1514.

[2] R. Sharda, "Neural Networks for the MS/OR Analyst: An Application Bibliography". Interfaces, 1994, vol. 24(2), pp. 116-130.

[3] Matroushi, M.S, S. Samarasinghe," Building a Hybrid Neural Network Model for Gold Price Forecasting", Lincoln University: Canterbury, New Zealand.

[4] B. Irie, S. Miyake,"Capabilities of three-layered perceptrons". IEEE International Conference on Neural Networks, 1988, vol. 1, pp. 641-648.

[5] G. Zhang, B.E. Patuwo, M. Y. Hu. "Forecasting with artificial neural networks: The state of the art." International Journal of Forecasting, 1998, vol. 14, pp. 35-62.

[6] F.Rosenblatt, “The perceptron, a perceiving and recognizing automaton”, Project Para Report No. 85-460-1, Cornell Aeronautical Laboratory (CAL), jan. 1957 .

[7] C. Gershenson , “Artificial Neural Networks for Beginners” Networks,2003., vol. cs.NE/0308, pp. 8.

[8] M. Bilgehan, "Comparison of ANFIS and NN models—With a study in critical buckling load estimation", Applied Soft Computing, 2011, vol.11, pp. 3779-3791.

[9] J. R. Jong,C. T. Sun, E. Mizutani, “Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence”, Prentic Hall, 1997.