

Stock Price Forecasting: Autoregressive Modelling and Fuzzy Neural Network

D. Marcek

Fac. Management Science and Informatics, Univ. of Zilina,
010 26 Žilina Slovak Republic, Fac. Philosophy and Science,
Salesian University, 74601 Opava, Czech Republic

Abstract

Most models for the time series of stock prices have centered on autoregressive (AR) processes. Traditionally, fundamental Box-Jenkins analysis [3] have been the mainstream methodology used to develop time series models. Next, we briefly describe the develop a classical AR model for stock price forecasting. Then a fuzzy regression model is then introduced Following this description, an artificial fuzzy neural network based on B-spline membership function is presented as an alternative to the stock prediction method based on AR models. Finally, we present our preliminary results and some further experiments that we performed.

Keywords: B-spline function, Artificial Neural Network (ANN), Fuzzy Autoregressive Model (FAR)

1 Introduction

The paper compares the forecasts from autoregressive (AR) model of stock prices and fuzzy neural network (FNN) specification. Our motivation for this comparison lies in the recent increasing interest in the use of neural networks for forecasting purposes of economic variables. Several studies [2], [8] have compared nonfuzzy neural network and traditional Box-Jenkins model. There are only a few comparative analyses between FNN and standard statistical approaches [12].

In [6] the stock price autoregressive (AR) models based on the Box-Jenkins methodology [3] were described. Although an AR model can reflect well the reality, these models are not suitable for situations where the quantities are not functionally related. In economics, finance and so on, there are however many situations where we must deal with uncertainties in a manner like humans, one may incorporate the concept of fuzzy sets into the statistical models. The fuzzy regression is another efficient approach for computing the parameter of the structure for an uncertain situation and for predicting of uncertain events following the decision.

The fuzzy regression models have been in use in analyses for many years. Lots of issues of journal *Fuzzy Sets and Systems* as well as many others have been articles

whose analyses are based on the fuzzy regression models. From reviewing of these papers, it became clear that in economic applications the use of method is not on the same level as analyses using classical linear regression. Computers play an important role in fuzzy regression analyses and forecasting systems. The widespread use of the method is influenced by inclusion of fuzzy regression routines in major computer software packages and selection of appropriate forecasting procedure.

The primary objective of this paper is a focused introduction to the fuzzy regression model and its application to the analyses and forecasting from classical regression model of view. In Section 2, we briefly describe some basic notions of fuzzy linear regression. Following this description, we present evaluation of fuzzy linear regression model in the context of a practical application in comparison to the AR model presented in [6]. In this paper we primarily concern with short-term forecasting, say one period in the future. The data set used in this paper (163 observations, stock price) is available, and is depicted in [6]. To build a forecast model we have defined the sample period for analysis y_1, \dots, y_{128} , i.e. the period over which we will be building or estimating the forecasting model and the ex post forecast period (validation data set), y_{129}, \dots, y_{163} as the time period from the first observation after the end of the sample period to the most recent observation. In order to study the regularities of this process, the observed time series is viewed as a realization of a stochastic process. To simplify our notes, we suppose that the times at which the observations are recorded are regularly spaced, which allows us to consider an index by taking only integer values. The process is denoted by $y_t = (y_t, t \in T)$, where the index set T is A for analysis data set or E for ex post forecast data set.

2 AR modelling

We give an example that illustrates one kind of possible results. We will regard these results as the referential values for the approach of fuzzy autoregressive and ANN modelling.

To illustrate the Box-Jenkins methodology, consider the stock price time readings of a typical company (say VAHOSTAV company). We would like to develop a time series model for this process so that a predictor for the process output can be developed. The data was collected for the period January 2, 1997 to December 31, 1997 which provided a total of 163 observations (see Fig. 1). To build a forecast model the sample period for analysis y_1, \dots, y_{128} was defined, i.e. the period over which the forecasting model was developed and the ex post forecast period (validation data set), y_{129}, \dots, y_{163} as the time period from the first observation after the end of the sample period to the most recent observation. By using only the actual and forecast values within the ex post forecasting period only, the accuracy of the model can be calculated.

Figure 1. The data for VAHOSTAV stock prices (January 1997 - August 1997) and the values of the AR(7) model for VAHOSTAV stock prices estimated by GL algorithm

After some experimentation, we have identified two models for this series (see [3]): the first one (1) based on Box-Jenkins methodology and the second one (2) based on signal processing.

$$y_t = \xi + a_1 y_{t-1} + a_2 y_{t-2} + \varepsilon_t \quad (1)$$

$$t = 1, 2, \dots, N - 2$$

$$y_t = - \sum_{k=1}^7 a_k y_{t-k} + \varepsilon_t \quad (2)$$

$$t = 1, 2, \dots, N - 7$$

The final estimates of model parameters (1), (2) are obtained using OLS (Ordinary Last Square) and two adaptive filtering algorithms in signal processing [1]. The Gradient Lattice (GL) adaptive algorithm and Last Squares Lattice (LSL) algorithm representing the parameter estimates of the predictors (1), (2) were used. In Tab. 1 the parameter estimates for model (2) and corresponding RMSE's are given. The Fig. 1 shows the GL prediction results and actual values for stock price time series in both analysis and ex post forecast period.

3 Fuzzy autoregressive (FAR) modelling

Next, we examine the application of fuzzy linear regression model [11] to the stock price time readings used in (1) and (2). Recall that the models in (1) and (2) fit to the stock prices were the AR(2) and AR(7) processes. In the fuzzy regression model proposed by Tanaka et al. [11], the parameters are the fuzzy numbers.

Tab. 1

OLS, GL and LSL estimates of AR models

Model	Order	Est. proc	\hat{a}_1	\hat{a}_2	\hat{a}_3	\hat{a}_4	\hat{a}_5	\hat{a}_6	\hat{a}_7	RMSE*
(1)	2	OLS	1.113	-0.127			$\xi = 26.639$			67.758
(2)	7	GL	-0.7513	-0.1701	-0.0230	-0.0128	-0.0028	-0.0472	0.0084	68.540
(2)	7	LSL	-0.8941	-0.6672	0.7346	-0.2383	0.1805	-0.5692	0.4470	94.570

*ex post forecast period

The regression function of such a fuzzy parameters can be modeled by the following equation

$$Y_t = A_0 * \varphi_0(x_{0t}) \oplus A_1 * \varphi_1(x_{1t}) \oplus \dots \oplus A_k * \varphi_k(x_{kt}) = A'x_t \quad (3)$$

where A_0, A_1, \dots, A_k are fuzzy numbers, \oplus and $*$ are fuzzy addition and fuzzy multiplication operators respectively, Y_t is fuzzy subset of y_t . This kind of fuzzy modelling is know as fuzzy parameter extension.

The problem to find out fuzzy parameters gives the following linear programming solution [11]

$$\begin{aligned} \min s &= c_0 + c_1 + \dots + c_k \\ \text{subject to } &c_j \geq 0 \end{aligned}$$

and

$$(h - 1)c' |x| - (y_t - x'\alpha) \geq 0 \quad (4)$$

$$(1 - h)c' |x| - (y_t - x'\alpha) \geq 0$$

for $t = 1, 2, \dots, N$

where $c_j, j = 0, 1, \dots, k$ is the width or spread around the center of the fuzzy number, $\alpha = (\alpha_0, \alpha_1, \dots, \alpha_k)$ denotes vector of center of the fuzzy numbers for model parameters, $x+ = (x_0, x_1, \dots, x_k)^+$ denotes vector of regressor variables in (3), h is an imposed threshold $h \in [0, 1]$ (see [9]). A choice of the h value influences the widths c_j of the fuzzy parameters. The h value expresses a measure of the fitting of the estimated fruzzy model (3) to the given data. The fuzzines of $c' = (c_0, c_1, \dots, c_k)$ of the parameters $\tilde{A}_0, \tilde{A}_1, \dots, \tilde{A}_k$ for the models (1) and (2) are given in Tab. 2.

The forecast for future observation is generated sucessively throught the Eq. (3) by replacing the functions of the independent variables $(\varphi_j(x_{jt}), j = 0, 1, \dots, k$ by observations y_{t-j} . Then the forecasting function of the fuzzy AR process is

$$Y_{T+1}(T) = A_0 \oplus A_1 * y_T \oplus A_2 * y_{T-1} \oplus \dots \oplus A_k * y_{T-k+1} \quad (5)$$

where $Y_{T+1}(T)$ is the forecast for period $T+1$ made at origin T . We observe that the forecasting procedure (5) produces forecast for one period ahead. As a new observation becomes available, we may set the new current period $T+1$ equal to T and compute the next forecast again according to (5).

Tab. 2

h=0.5 \hat{A}_k k:	0	1	2	3	4	5	6	7
Model AR(2)								
Modal values(α)	26.639	1.113	-0.127					
Spread (c)	0	0	0.229008					
Model AR(7)								
Modal values(α)	45.930	1.085	0.0861	-0.2531	0.0836	-0.0057	0.2081	-0.2281
Spread (c)	0	0	0	0	0.209587	0	0	0

4 B-spline neural network approach

The concept of fuzzy neural network (FNN) can be approached from several different avenues. The one that we have used for stock price forecasts is shown in Fig. 2. This figure shows the FNN with $p \times n$ input neurons (input layer), a single hidden layer with p processing units (fuzzy neurons) and one output unit.

Figure 2. The neuro fuzzy system architecture

Input selection is of crucial importance to the successful development of FNN models. In models (1) and (2) potential inputs were chosen based on traditional statistical analysis: these included the raw stock price series and lags thereof. The relevant lag structure of potential inputs was analysed using traditional statistical tools: ACF, PACF and the MSE criterion. All the above techniques are in reality imprecise (we developed parsimonious models, that is, models which adequately describe the time series yet contain relatively few parameters, the theoretical ACF was estimated by the sample ACF, etc.). In fact we obtain a certain number of input values, but we are sure that these values are one of many other possible

values. Thus, we will further suppose that the potential inputs, which were chosen based on statistical analysis, are fuzzy numbers characterized by a membership functions (the uncertainty is modeled as a possibility distribution) belonging to a class of bell shaped functions.

Inputs to the fuzzy neuron in hidden layer are fuzzy numbers denoted $B_{j,k,t}$, $j = 1, 2, \dots, p$, k identifies the order of the B-spline basis functions. They express the neural input signals in terms of their membership functions based on B-spline basis functions of the data. This concept is often called as B-spline FNN [12].

Now, let us suppose that the system has $B'_{j,k,t} = [B_{1,k,t}, B_{2,k,t}, \dots, B_{p,k,t}]$ as inputs and $y' = [y_1, y_2, \dots]$ as outputs. Then the information set ψ describing process behaviour may be written in the form

$$\psi = \{(B_{1,k,t}, B_{2,k,t}, \dots, B_{p,k,t}, y_t) : t \in T\} \quad (6)$$

Each the j -th input neuron distribute the inputs to the j -th neuron in the hidden layer. Neural input signals are then weighted by weights denoted $\omega_{j,t}$. In general, the weights are in the range of $(0, 1)$ Each processing unit performs internal operations on these neural inputs and computes the neural output signal \hat{a}_j . The internal operations are based on aggregation, i.e., the sum of the products of weights and inputs, and its transformation into the neural output \hat{a}_j . These two internal operations for j -th neuron in the hidden layer are defined as

$$U_j = \sum_{t \in A} \omega_{j,t} B_{j,k,t} \quad (7)$$

for aggregation, where U_j is a measure similarity between the inputs and weights, and

$$\hat{a}_j = f(U_j) \quad (8)$$

for transformation operation, where f is the type of transfer function. We set this function to the identity, that is, $f(U_j) = U_j$.

The neuron in the output layer provides simply the computation of Eqs. (1), (2) and produces output signal \hat{y}_t .

The learning algorithm is based on error signal. The neural network modifies the weights $\omega_{j,t}$ in synaptic connections with respect to the desired fuzzy system output y_t . The error of the fuzzy system, i.e., the difference between the fuzzy system forecast \hat{y}_t and the actual value y_t is analysed through the RMSE. Let \hat{y}_t be a linear function

$$\hat{y}_t = \hat{a}_1 y_{t-1} + \hat{a}_2 y_{t-2} + \dots + \hat{a}_p y_{t-p} = \sum_{j=1}^p \sum_{t \in A} \omega_{j,t} B_{j,k,t} y_{t-j} \quad (9)$$

such that

$$\sum_{t \in T} (y_t - \hat{y}_t)^2 \leq \sum_{t \in T} [y(z'_t) - \hat{y}_t]^2 \quad (10)$$

for any other linear function $y(z'_t)$. It is clear that such function can be derived by using the Normalized Back-Propagation (NBP) algorithm which minimizes the term of the left-hand side of the relation (10) by adjusting the weights $\omega_{j,t}$ in Eq. (9). The proof of the convergence of the NBP algorithm can be found in [8].

The measure of similarity (7) may be defined as the inner product of vectors $B_{j,k}(y_{t-j})$ and $\omega_{t-j}(j)$, that is

$$U_j = \omega_{t-j}^T(j) B_{j,k}(y_{t-j}) \quad (11)$$

where : $\omega_{t-j}^T(j) = [\omega_{t-j}(j), \omega_{t-j+1}(j), \dots, \omega_{t-j+n}(j)]$ is a $1 \times n$ row vector of the weights and

$B_{j,k}^T(y_{t-j}) = [B_{j,k}(y_{t-j}), B_{j,k}(y_{t-j+1}), \dots, B_{j,k}(y_{t-j+n})]$ is a $1 \times n$ row vector of the B-spline functions.

Next we show that the B-spline neural network may be considered as a fuzzy linear controller. We now define the vectors as follows:

Let z_j^T be an $1 \times n$ row vector of the regressor variables

$$z_j^T = [y_{t-j}, y_{t-j+1}, \dots, y_{t-j+n}] \quad j = 1, 2, \dots, p$$

y^T be an $1 \times n$ row vector of the observations

$$y^T = [y_1, y_2, \dots, y_n]$$

and \hat{a}^T be an $1 \times p$ row vector of the parameters

$$\hat{a}^T = [\hat{a}_1, \hat{a}_1, \dots, \hat{a}_p]$$

Then the concept of B-spline FNN may be also considered as a well known Sugeno and Takagi [10] linear fuzzy controller which (in our notation) has the following form

$$\begin{aligned} \mathbf{R} = \mathbf{if} \quad U_1 = \hat{a}_1 \quad \text{and} \quad U_2 = \hat{a}_2 \quad \text{and} \quad \dots \quad \text{and} \quad U_p = \hat{a}_p \\ \mathbf{then} \quad \hat{y}_t = \hat{a}^T z_j \quad t = 1, 2, \dots, n \end{aligned} \quad (12)$$

where the fuzzy linear control rules R has been derived by neural network purely from the database describing previous or next behaviour of the system.

5 Empirical results

The network described in Section 3 was trained in software at the Faculty of Management Science and Informatics Zilina. The statistical forecast accuracy of the FNN according to Fig. 2 depends on the type of transfer function in Eq. (8) and the formulation of the B-spline curve in Eq. (6). The approximation is better the higher the value of k . All B-spline basis functions are cubic ones. Assume that the mesh points of B-spline basis function is x_i , $i = 1, 2, 3, 4$. Then the cubic B-spline basis functions ($k=3$) for $i = 1$ have the form as follows

$$\begin{aligned}
B_{i,3}(y_{t-j}) &= \frac{y_{t-j}-x_i}{x_{i+3}-x_i} \frac{(y_{t-j}-x_i)^2}{(x_{i+2}-x_i)(x_{i+1}-x_i)} && \text{for } y_{t-j} \in (x_i, x_{i+1}) \\
B_{i,3}(y_{t-j}) &= \frac{(y_{t-j}-x_i)}{x_{i+3}-x_i} \left[\frac{(y_{t-j}-x_i)(x_{i+2}-y_{t-j})}{(x_{i+2}-x_i)(x_{i+2}-x_{i+1})} + \frac{(y_{t-j}-x_{i+1})(x_{i+3}-y_{t-j})}{(x_{i+3}-x_{i+1})(x_{i+2}-x_{i+1})} \right] + \\
&\quad + \frac{(x_{i+4}-y_{t-j})}{(x_{i+4}-x_{i+1})} \left[\frac{(y_{t-j}-x_{i+1})^2}{(x_{i+3}-x_{i+1})(x_{i+2}-x_{i+1})} \right] && \text{for } y_{t-j} \in (x_{i+1}, x_{i+2}) \\
B_{i,3}(y_{t-j}) &= \frac{(y_{t-j}-x_i)}{x_{i+3}-x_i} \left[\frac{(x_{i+3}-x_i)^2}{(x_{i+3}-x_{i+1})(x_{i+3}-x_{i+2})} \right] + \\
&\quad + \frac{(x_{i+4}-x_i)}{x_{i+4}-x_{i+1}} \left[\frac{(y_{t-j}-x_{i+1})(x_{i+3}-y_{t-j})}{(x_{i+3}-x_{i+1})(x_{i+3}-x_{i+2})} + \frac{(x_{i+4}-y_{t-j})(y_{t-j}-x_{i+2})}{(x_{i+4}-x_{i+2})(x_{i+3}-x_{i+2})} \right] && \text{for } y_{t-j} \in (x_{i+2}, x_{i+3}) \\
B_{i,3}(y_{t-j}) &= \frac{(x_{i+4}-y_{t-j})}{x_{i+4}-x_{i+1}} + \left[\frac{(x_{i+4}-y_{t-j})^2}{(x_{i+4}-x_{i+2})(x_{i+4}-x_{i+3})} \right] && \text{for } y_{t-j} \in (x_{i+3}, x_{i+4}) \\
B_{i,3}(y_{t-j}) &= 0 && \text{otherwise}
\end{aligned}$$

where y_{t-j} , $j = 1, 2, \dots, p$, $t \in A$ are observations. These mesh points are given as

$$x_1 = \min \{y_{t-j}\} \quad x_5 = \max \{y_{t-j}\} \quad t \in A$$

$$x_2 = (x_5 - x_1) / 4 + x_1 \quad x_3 = 2[(x_5 - x_1) / 4] + x_1 \quad x_4 = 3[(x_5 - x_1) / 4] + x_1$$

Our FNN was trained on the training data set. Periodically, during the training period, the RMSE of the FNN were measured not only on the training set but also on the validation set. The final FNN chosen for the stock price prediction is the one with the lowest error on the validation set. Note also, the training phase was finished after $5 \cdot 10^3$ epochs, the best model being obtained after $2,3 \cdot 10^3$ epochs (see Fig. 3).

The RMSE's of our predictor models are shown in Tab. 3. From this table can be seen that the basic (non fuzzy) artificial neural network architecture described in [7] does not support its use for daily frequencies. The initial results of the FNN forecasting model are clearly better.

Tab. 3

Model	RMSE*
AR(2)	67.7
Basic (non fuzzy) neural network (see [7])	67.2
FNN	63.5

* Validation set

Figure 3. RMSE's - validation set

Acknowledgement

This work was supported by Slovak grant foundation under the grant No. 1/6257/99. I thank M. Tkacin for the data sets and R. OndrejssÜ for computational support.

References

- [1] Bayhan, G.M.: *Sales Forecasting Using Adaptive Signal Processing Algorithms*. Neural Network World 4-5/1997, Vol. 7, (579-589)
- [2] Boero G., Cavalli, E.: Exchange rate Forecasting: Neural Networks Versus Econometric Models. Neural Network World 1/1997 (29-42)
- [3] Box, G.E., Jenkins, G.M.: *Time Series Analysis, Forecasting and Control*. Holden-Day, San Francisco, CA 1976
- [4] Kosko, B.: Neural networks and fuzzy systems - a dynamical systems approach to machine intelligence. Prentice-Hall International, Inc. 1992
- [5] Kosko, B.: Fuzzy system as universal approximators. Proc. IEE Int. Conf. on Fuzzy Systems. San. Diego, 1992, (1153-1162)
- [6] Marček, D.: *Stock Price Prediction Using Autoregressive Models and Signal Processing Procedures*. Proceedings of the 16th Conference MME'98, Cheb 8.-10.9.1998, (114-121)
- [7] Marček, D.: Neural Network Model for Stock Prices Forecasting. Neural Network World 3/1997, (347-352)

- [8] Polycarpou M.M., Ioannou P.A.: Learning and convergence analysis of neural-type structured networks. *IEE Transactions on Neural Networks*, 3, 1992, (39-50)
- [9] Savic, D.A., Pedricz, W.: Evaluation of fuzzy regression models. *Fuzzy Sets and Systems*. 39 (1991) North-Holland (51-53)
- [10] Takagi T., Sugeno M.: Fuzzy identification of systems and its applications to modelling and control. *IEEE trans. System Man. Cybernet*, 16 yr 1985, (116-132)
- [11] Tanaka, H., Uejima, S. and Asai, K.: Linear Regression Analysis with Fuzzy Model. *IEEE Transaction on Systems, Man and Cybernetics*, Vol. SMC - 12. No. 6, November/December 1982, (903-906)
- [12] Wu Z.Q., Harris Ch.J.: Indirect Adaptive Neurofuzzy Estimation of Nonlinear Time Series. *Neural Network World* 3/96, (407-416)