Agricultural Economics Research Review
Vol. 26 (No.2) July-December 2013 pp 229-239

Agricultural Price Forecasting Using Neural Network Model: An Innovative Information Delivery System

Girish K. Jha[*a] and Kanchan Sinha[b]

[a]Indian Agricultural Research Institute, New Delhi – 110 012
[b]Indian Agricultural Statistics Research Institute, New Delhi – 110 012

Abstract

Forecasts of food prices are intended to be useful for farmers, policymakers and agribusiness industries. In the present era of globalization, management of food security in the agriculture-dominated developing countries like India needs efficient and reliable food price forecasting models more than ever. Sparse and time lag in the data availability in developing economies, however, generally necessitate reliance on time series forecasting models. The recent innovation in Artificial Neural Network (ANN) modelling methodology provides a potential price forecasting technique that is feasible given the availability of data in developing economies. In this study, the superiority of ANN over linear model methodology has been demonstrated using monthly wholesale price series of soybean and rapeseed-mustard. The empirical analysis has indicated that ANN models are able to capture a significant number of directions of monthly price change as compared to the linear models. It has also been observed that combining linear and nonlinear models leads to more accurate forecasts than the performances of these models independently, where the data show a nonlinear pattern. The present study has aimed at developing a user-friendly ANN based decision support system by integrating linear and nonlinear forecasting methodologies.

Key words: Hybrid model, neural networks, price forecasting, agriculture

JEL Classification: Q16, Q15

Introduction

Price forecasting is an integral part of commodity trading and price analysis. Quantitative accuracy with small errors, along with turning point forecasting power is important for evaluating forecasting models. Agricultural commodity production and prices are often random as they are largely influenced by eventualities and are highly unpredictable in case of natural calamities like droughts, floods, and attacks by pests and diseases. This leads to a considerable risk and uncertainty in the process of price modelling and forecasting. Agricultural commodity prices play an important role in consumers' access to food as they directly influence their real income, especially among the poor who spend a large proportion of their income on food. Since food price is an important component to fight hunger, policymakers need reliable forecasts of expected food prices in order to manage food security. Before the onset of liberalization and globalization, the government was controlling food prices, thus rendering food price forecasting a low value-added activity. Presently, the food prices are determined by the domestic and international market forces. This leads to increased price variability, and accords importance to reliable price forecasting techniques. The price forecasts are important for farmers also as they base their production and marketing decisions on the expected prices that may have financial repercussions many months later.

* Author for correspondence
Email: girish.stat@gmail.com

Agricultural Price and Time Series Modelling

Agricultural price modelling is different from modelling of non-farm goods and services due to certain special features of agricultural product markets. The characteristic features of agricultural crops include seasonality of production, derived nature of their demand, and price-inelastic demand and supply functions. The biological nature of crop production plays an important role in agricultural product price behaviour.

There are two basic approaches of forecasting, namely structural and time series models. The structural models proceed from the first principles of consumer and producer theory to identify the demand and supply schedules and the equilibrium prices resulting from their intersection. The structural modelling techniques provide valuable insights into the determinants of commodity price movements. The computational and data demands of structural price forecasting generally far exceed than what are routinely available in the developing countries. Consequently, researchers often rely on parsimonious representations of price processes for their forecasting needs. Contemporary parsimonious form of price forecasting relies heavily on time series modelling. The time series modelling requires less onerous data input for regular and up-to-date price forecasting.

In time series modelling, past observations of the same variable are collected and analyzed to develop a model describing the underlying relationship. During the past few decades, much effort has been devoted to the development and improvement of time series forecasting models. One of the most important and widely used time series models is the Auto Regressive Integrated Moving Average (ARIMA) model. The popularity of ARIMA model is due to its statistical properties as well as use of well-known Box-Jenkins methodology in the model building process.

Recently, Artificial Neural Network (ANN) modelling has attracted much attention as an alternative technique for estimation and forecasting in economics and finance (Zhang *et al*., 1998; Jha *et al*., 2009). ANN is a multivariate non-linear non-parametric data driven self-adaptive statistical method. The main advantage of neural network is its flexible functional form and universal functional approximator. With ANN, there is no need to specify a particular model form for a given data set. ANN has found applications in fields like biology, engineering, economics, etc. and its use in economics has been surveyed by Kuan and White (1994).

Rationale of Research Issue

Very few studies have been undertaken on agriculture price forecasting using ANN models. Moreover, the value of neural network models in forecasting economic time series, has been established for developed countries like USA, Canada, Germany, etc., but little work has been undertaken for developing countries in general and India in particular. Literature suggests that the performance of a non-linear model should be evaluated on the basis of percentage of forecasts that correctly predict the direction of change instead of measures based on error-terms. The prediction of turning point is more crucial for any commodity price forecasting. Lastly, as agricultural price data often contain both linear and nonlinear patterns, no single model is capable to identify all the characteristics of time series data on agricultural prices. Obviously, there is a need to examine the price forecasting performance of hybrid model which takes advantage of the unique strength of both linear ARIMA method and nonlinear ANN model.

The above facts motivated us to assess the forecasting accuracy of neural network model and traditional statistical models for agricultural price forecasting using real price data by taking into account the major limitations of previous studies. This paper has summarized the experience of forecasting price and direction of change using ANN model with two monthly wholesale oilseeds price series compared to other approaches, where one series was linear and the other was nonlinear in nature. An attempt has also been made to discuss opportunities and advantages of soft computing based decision support system in agricultural price forecasting.

Methodology

Neural Network Model

The time series data can be modelled using ANN by providing the implicit functional representation of time, whereby a static neural network like multilayer perceptron is bestowed with dynamic properties

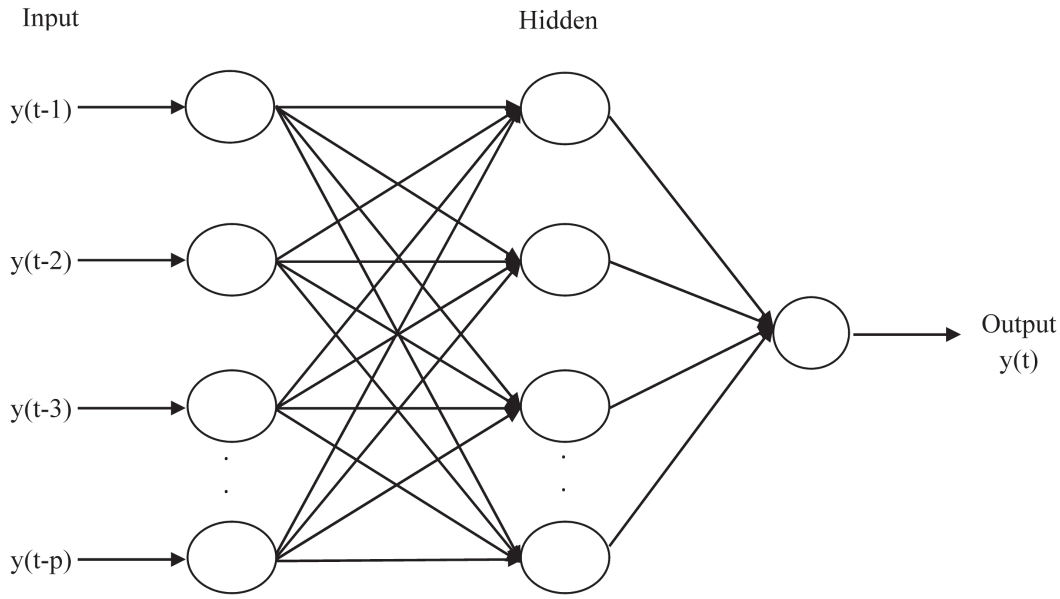

Figure 1. Time-Delay Neural Network (TDNN) with one hidden layer

(Haykin, 1999). A neural network can be made dynamic by embedding either long-term or short-term memory, depending on the retention time, into the structure of a static network. One simple way of building short–term memory into the structure of a neural network is through the use of time delay, which can be implemented at the input layer of the neural network. An example of such an architecture is a Time-Delay Neural Network (TDNN) (Figure 1), which has been employed in the present study.

The ANN structure for a particular problem in time series prediction includes the determination of number of layers and total number of nodes in each layer. It is usually determined through experimentation as there is no theoretical basis for determining these parameters. It has been proved that neural networks with one hidden layer can approximate any non-linear function given a sufficient number of nodes at the hidden layer and adequate data points for training. In this study, we have used neural network with one hidden layer. In time series analysis, the determination of number of input nodes which are lagged observations of the same variable plays a crucial role as it helps in modelling the autocorrelation structure of the data. The determination of number of output nodes is relatively easy. In this study, one output node has been used. Multi-step ahead forecasting is performed using iterative procedure following Box-Jenkins ARIMA Time Series modelling methodology. This involves use of forecast value as an input for forecasting the future value. It is always better to select the model with a smaller number of nodes in the hidden layer as it improves the out-of-sample forecasting performance and also avoids the problem of over-fitting. The general expression for the final output value y_{t+1} in a multi-layer feed forward time delay neural network is given by Equation (1):

$$y_{t+1} = g\left[\sum_{j=0}^{q} \alpha_j f\left(\sum_{i=0}^{p} \beta_{ij} y_{t-i}\right)\right] \quad ...(1)$$

where, f and g denote the activation function at the hidden and output layers, respectively; p is the number of input nodes (tapped delay); q is the number of hidden nodes; β_{ij} is the weight attached to the connection between i^{th} input node to the j^{th} node of hidden layer; α_j is the weight attached to the connection from the j^{th} hidden node to the output node; and y_{t-i} is the i^{th} input (lag) of the model. Each node of the hidden layer receives the weighted sum of all the inputs, including a bias term for which the value of input variable will always be one. This weighted sum of input variables is then transformed by each hidden node using the activation function f which is usually a non-linear sigmoid function. In a similar manner, the output node also receives the weighted sum of the output of all the hidden nodes and produces an output by transforming the weighted sum using its activation function g. In the time series analysis, f is often chosen as the Logistic Sigmoid function and g, as an identity function. The logistic function is expressed as Equation (2):

$$f(y) = \frac{1}{1+e^{-y}} \quad ...(2)$$

For p tapped delay nodes, q hidden nodes, one output node and biases at both hidden and output layers, the total number of parameters (weights) in a three layer feed forward neural network is $q(p + 2) + 1$.

For a univariate time series forecasting problem, the past observations of a given variable serve as input variables. The TDNN model attempts to map the following function:

$$y_{t+1} = f(y_t, y_{t-1}, \dots \dots y_{t-p+1}, w) + \varepsilon_{t+1} \quad ...(3)$$

where, y_{t+1} pertains to the observation at time t+1, p is the number of lagged observation, w is the vector of network weights, and ε_{t+1} is the error-term at time t+1. Hence, TDNN acts like a non-linear autoregressive model. The neural network toolbox of MATLAB 7.10 software was used to carry out computation relating to TDNN model.

The ARIMA Model

In an Auto-Regressive Integrated Moving Average (ARIMA) model, time series variable is assumed to be a linear function of the previous actual values and random shocks. In general, an ARIMA model is characterized by the notation ARIMA (p, d, q), where p, d and q denote orders of Auto-Regression (AR), Integration (differencing) and Moving Average (MA), respectively. ARIMA is a parsimonious approach which can represent both stationary and non-stationary processes.

An ARMA (p, q) process is defined by Equation (4):

$$y_t = c + \emptyset_1 y_{t-1} + \emptyset_2 y_{t-2} + \cdots + \emptyset_p y_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \cdots - \theta_q \varepsilon_{t-q} \quad ...(4)$$

where, y_t and ε_t are the actual value and random error at time period t, respectively, Φ_i (i=1, 2,……,p) and ϕ_i (j=1, 2,……,q) are the model parameters. The random errors, ε_t are assumed to be independently and identically distributed with a mean of zero and a constant variance of σ^2.

The first step in the process of ARIMA modelling is to check for the stationarity of the series as the estimation procedure is available only for a stationary series. A series is regarded stationary if its statistical characteristics such as the mean and the autocorrelation structures are constant over time. The stochastic trend of the series is removed by differencing, while logarithmic transformation is employed to stabilize the variance. After appropriate transformation and differencing, multiple ARMA models are chosen on the basis of Auto-Correlation Function (ACF) and Partial Auto- Correlation Function (PACF) that closely fit the data. Then, the parameters of the tentative models are estimated through any non-linear optimization procedure such that the overall measure of errors is minimized or the likelihood function is maximized. Lastly, diagnostic checking for model adequacy is performed for all the estimated models through the plot of residual ACF and using Portmonteau test. The most suitable ARIMA model is selected using the smallest Akaike Information Criterion (AIC) or Schwarz-Bayesian Criterion (SBC) value and the lowest root mean square error (RMSE). In this study, all estimations and forecasting of ARIMA model have been done using SAS/ETS 9.2.

The Hybrid ARIMA - TDNN Methodology

In this section, the time series decomposition is proposed in which ARIMA and TDNN models are combined in order to obtain a robust and efficient methodology for time series forecasting. Accordingly, we postulate that our time series data can be decomposed into a linear and a nonlinear component (Rojas *et al*., 2008), viz.

$$y_t = L_t + N_t \quad ...(5)$$

where, y_t is the observed time series data, L_t is the linear auto-regressive component, and N_t is the non-linear component. In this approach, we apply an ARIMA model to the data series to fit the linear part and the residuals are modelled using neural network model only if there is an evidence of non-linearity for the series. Figure 2 shows a schematic diagram of this method. Let r_t be the residual at time t of the linear component, then

$$r_t = y_t - \hat{L}_t \quad ...(6)$$

where, $\hat{L}_t$ is the estimate of the linear auto-regressive component. For non-linear components, we apply neural network model, i.e.

$$\hat{r}_t = f(r_{t-1}, r_{t-2}, \ . \ . \ . \ , r_{t-p}) \qquad \ldots(7)$$

where, p is the number of input delays and f is the nonlinear function. So the combined forecast is given by Equation (8):

$$y_t = \hat{L}_t + \hat{r}_t + \varepsilon_t \qquad \ldots(8)$$

where, ε_t is the error-term of the combined model at time t. Here, it is assumed that since ARIMA model cannot capture the nonlinear structure of the data, the residual of linear model will contain information about nonlinearity. Hence, the hybrid architecture is expected to exploit the feature and strength of both the models in order to improve the overall forecasting performance.

In this study, the McLeod and Li test (1983) has been applied to detect non-linearity in the data. This test is based on the autocorrelations of the squared residuals. In this test, the residuals which are obtained from fitted ARIMA model are utilized to test non-linearity. The test statistic is given by Equation (9):

$$Q = n(n+2)\sum_{i=1}^{h}\frac{r^2(i)}{n-i} \qquad \ldots(9)$$

where, $r(i)$ is the autocorrelation of the squared residuals, and h is the number of autocorrelations.

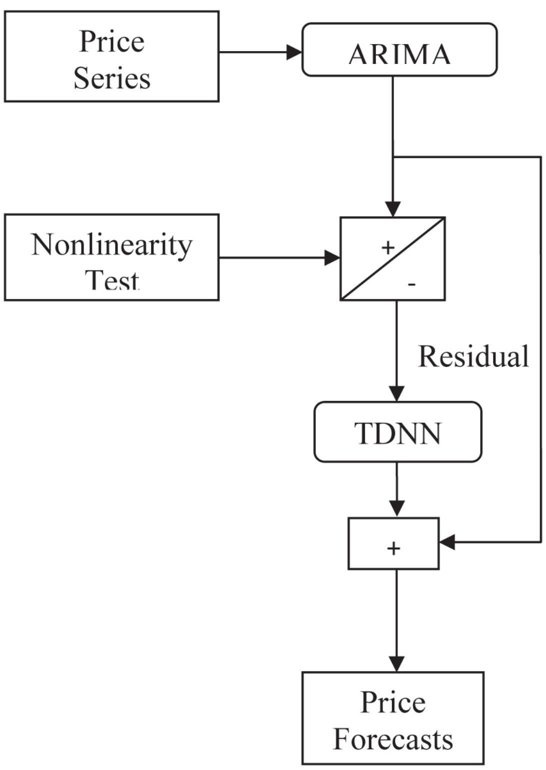

Figure 2. Hybrid method that combines both ARIMA and TDNN models

Forecast Evaluation Methods

The forecasting ability of different models is assessed with respect to two common performance measures, viz. the root mean squared error (RMSE) and the mean absolute deviation (MAD). The RMSE measures the overall performance of a model and is given by Equation (10):

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2} \qquad \ldots(10)$$

where, y_t is the actual value for time t, $\hat{y}_t$ is the predicted value for time t, and n is the number of predictions. The second criterion, the mean absolute deviation is a measure of average error for each point forecast and is given by Equation (11):

$$MAD = \frac{1}{n}\sum_{t=1}^{n}|y_t - \hat{y}_t| \qquad \ldots(11)$$

where the symbols have the same meaning as above.

Data

This paper has used the monthly average wholesale (nominal) price (rupees per quintal) of two major crops of oilseeds in India, viz. soybean and rapeseed-mustard, traded in the Indore (Madhya Pradesh) and Delhi markets, respectively, to evaluate the prediction ability of different models. The data on soybean were obtained from the website of the Soybean Processors Association of India (SOPA), Indore, and on rapeseed-mustard were collected from various issues of *Agricultural Prices in India*, published by the Directorate of Economics and Statistics, Government of India, New Delhi. The price series on soybean covered a period of 228 months (October, 1991 to September, 2010) and on rapeseed-mustard covered a period of 372 months (January, 1980 to December, 2010). These series illustrate the complexity and variation of typical agricultural price data (Figure 3). These prices were deflated using the wholesale price index data (2004-05=100) of oilseeds estimated by the Office of Economic Advisor, Ministry of Commerce & Industry, Government of India. The basic characteristics of the price series used in the study are presented in Table 1.

Empirical Results and Discussion

Data Preprocessing

The data preprocessing refers to analyzing and transforming the input and output variables to minimize noise, highlight important relationships, detect trends, and flatten the distribution of variables to assist both traditional and neural network models in the relevant pattern. The first step in time series analysis is to plot the data. Figure 3 shows the time series plot of average monthly price of rapeseed-mustard from January 1980 to December 2010. A perusal of Figure 3 reveals a positive trend over time which indicates the non-stationary nature of series. A similar trend was observed in the case of soybean also.

In this study, we have applied the natural choice of logarithmic transformation to the data to stabilize the variance. The logarithmic transformation is used for the data which can take both small and large values and is characterized by an extended right hand tail distribution. The logarithmic transformation is one of the data processing techniques which also converts multiplicative or ratio relationship to additive which is believed to simplify and improve neural network training. We have applied the Augmented Dickey Fuller (ADF) test for each level and transformed series to test for the unit root and the results have been presented in Table 2. The values in Table 2 clearly show the non-stationarity of level and transformed series. Therefore, we have used first differencing for both the price series. The first differenced series were found to be stationary in both cases as indicated in Table 2 and hence further differencing was not required. The ACF and PACF of different series have not shown a strong and consistent seasonal pattern.

Table 1. Descriptive statistics of price series used in the study

Crop	Minimum (₹/q)	Maximum (₹/q)	Mean (₹/q)	Standard deviation (₹/q)	Skewness	Kurtosis
Soybean	646	2680	1256	472	1.21	3.74
Rapeseed-mustard	370	3175	1288	741	0.85	3.02

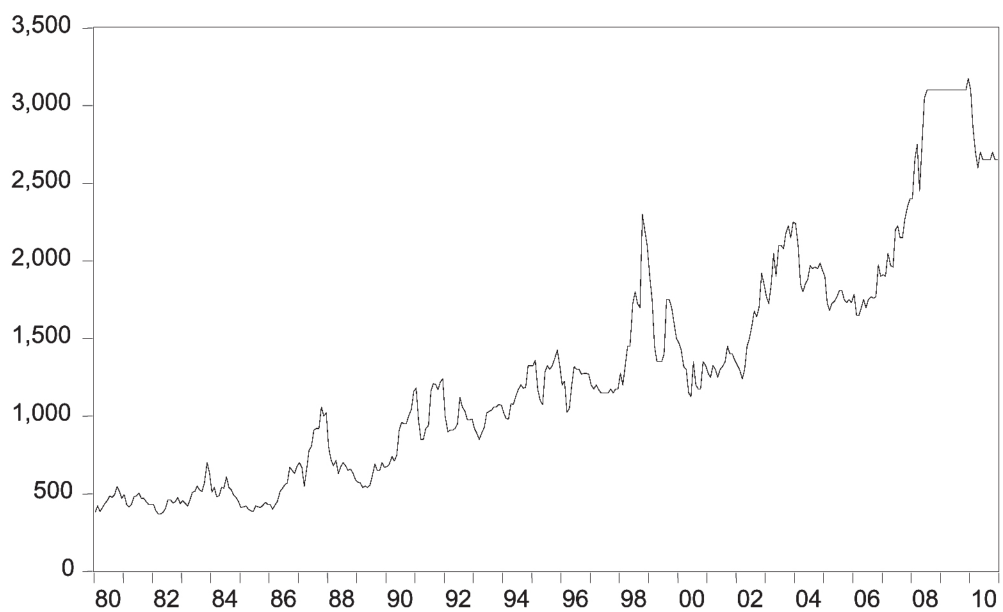

Figure 3. Rapeseed-mustard monthly price data from January 1980 to December 2010 (₹/q)

Table 2. Augmented Dickey-Fuller stationarity test for different series

Null hypothesis	Level series		Logarithmic transformed series		1st difference of transformed series	
	t-statistic	Prob.	t-statistic	Prob.	t-statistic	Prob.
Soybean series has a unit root	-1.951	0.308	-1.557	0.502	-11.666	< 0.0001
Rapeseed-mustard series has a unit root	-0.737	0.830	-1.321	0.621	-17.428	< 0.0001

Nonlinearity Test

For choosing the technique for modelling and prediction of data, it is important to find whether a given time series is non-linear or not. If there is an evidence of nonlinearity in the dynamics underlying the data generating process, then nonlinear models should be tried in addition to linear models for forecasting the data. This also enables us to examine whether nonlinearity tests provide any reliable guide for post sample forecast accuracy of neural network model. In this study, we have applied McLeod and Li nonlinearity test to the data set. It tests the null hypothesis of linearity against different types of possible nonlinearity and is based on the autocorrelations of squared residuals. In this study, autocorrelations up to 24 lags have been used for computing the test. The results of McLeod and Li non-linearity test presented in Table 3, reveal strong rejection of linearity in the case of rapeseed-mustard only. In other words, the analysis has indicated the existence of some hidden structure left unaccounted in the residuals of linear model in the case of rapeseed-mustard. Based on this evidence, we have suggested suitability of nonlinear model for price forecasting of rapeseed-mustard.

Neural Network and ARIMA Model

For developing a model, we have divided the data into two sets, viz. training set and testing set. The last twelve months price data were retained for testing. The training set was used for modelling procedure and in-sample prediction and testing set was kept for post-sample forecasting. The training set for the soybean and rapeseed-mustard series contained 216 and 360 observations, respectively. After logarithmic transformation, each series was differenced to make it stationary as price data are trended and nonstationary in nature. Then, we modelled the relative change in the price series which also had a meaningful economic interpretation.

Table 3. McLeod and Li non-linearity test for different series

Series	Value	Prob. value
Soybean	9.73	0.99
Rapeseed-mustard	87.82	less than 0.001

We have found the ARMA structure of differenced series, based on the autocorrelation function (ACF), partial autocorrelation function (PACF) and AIC information criterion. We obtained the best ARIMA model for each series based on the lowest AIC and BIC information criteria as well as the lowest RMSE and MAD values. We selected the ARIMA (1, 1, 0) for soybean and ARIMA (2, 1, 0) for rapeseed-mustard series. Due importance was given to the well-behaved residuals while selecting the best model.

We have found the best time delay neural network with single hidden layer for this study. Following the previous studies, the *logistic* and *identity* functions were used as activation function for the hidden nodes and output node, respectively. We have focused primarily on the one-step-ahead forecasting and the multi-step-ahead forecasting was done using the iterative procedure; so only one output node was employed. Hence, the model uncertainty was associated only with the number of tapped delays (p) which was the number of lagged observations in this case and the number of hidden layer nodes (q). The number of tapped delay and hidden nodes were determined through experimentation. We have used multiple starts, with different random starting points, in order to avoid local minima and find the global minimum. In particular, based on the training sample, we have trained each neural network model twenty times using twenty different sets of initial random weights. The overall performance of each configuration of TDNN model was evaluated on the basis of mean performance of 20 randomly initialized TDNN. We varied the number of input nodes from 1 to 6 and the number of hidden nodes from 2 to 10 with an increment of 2 with basic cross validation method. Thus, different numbers of neural network models were tried for each series before arriving at the final structure of the model.

There are many variations of the backpropagation algorithm used for training feed-forward networks. In this study, the Levenberg-Marquardt algorithm (Hagan and Menhaj, 1994), which has been designed to approach second-order training speed without computing the Hessian matrix, has been employed. It has been shown (Demuth and Beale, 2002) that this algorithm provides the fastest convergence for moderately sized feed-forward neural network used on function approximation problems. A typical TDNN structure with one hidden layer is denoted by I:Hs:O*l*, where I is the number of nodes in the input layer, H is the number of nodes in the hidden layer, O is the number of nodes in the output layer, s denotes the *logistic* sigmoid transfer function, and *l* indicates the *linear* transfer function. The forecasting ability of both models is assessed with respect to two common performance measures, viz. root mean squared error (RMSE) and mean absolute deviation (MAD). In this study, our interest was centred on short-term forecasting and hence we have considered forecast horizon up to one year. In terms of the forecast horizon, we have included the results for one month, three months, six months and twelve months ahead forecast.

The best time lagged neural network with single hidden layer was found for each series by conducting experiments with the basic cross validation method. Table 4 summarizes the forecasting performance of various TDNN models for rapeseed-mustard in terms of training and testing root mean square error (RMSE), respectively. A similar exercise was carried out for soybean also and the results have not been presented in the manuscript. Out of a total of 24 neural network structures, a neural network model with two input nodes and three hidden nodes (2:3s:1*l*) performed better than other competing models in respect of out-of sample forecasting for soybean series. Similarly, a TDNN with two lagged observations as input node and eight hidden nodes (2:8s:1*l*) showed the minimum training and testing RMSE for a forecasting horizon of 12 months in Table 4. This means that most accurate price forecast for the given series is obtained when the price of two preceding months is used as inputs.

Table 4. Forecasting performance of TDNN models for rapeseed-mustard price series

Model	No. of parameters	RMSE training	RMSE testing	MAD testing
1:2s:1l	7	0.0301	0.0163	0.0082
1:4s:1l	13	0.0301	0.0177	0.0092
1:6s:1l	19	0.0298	0.0172	0.0090
1:8s:1l	25	0.0298	0.0171	0.0088
1:10s:1l	31	0.0285	0.0172	0.0098
2:2s:1l	9	0.0293	0.0156	0.0105
2:4s:1l	17	0.0288	0.0160	0.0106
2:6s:1l	25	0.0280	0.0158	0.0092
2:8s:1l	33	0.0278	0.0124	0.0087
2:10s:1l	41	0.0266	0.0138	0.0085
3:2s:1l	11	0.0293	0.0159	0.0106
3:4s:1l	21	0.0279	0.0159	0.0098
3:6s:1l	31	0.0269	0.0186	0.0126
3:8s:1l	41	0.0266	0.0128	0.0091
3:10s:1l	51	0.0258	0.0149	0.0089
4:2s:1l	13	0.0294	0.0162	0.0106
4:4s:1l	25	0.0275	0.0165	0.0107
4:6s:1l	37	0.0269	0.0214	0.0138
4:8s:1l	49	0.0243	0.0163	0.0125
4:10s:1l	61	0.0244	0.0204	0.0145
5:2s:1l	15	0.0292	0.0160	0.0106
5:4s:1l	29	0.0275	0.0169	0.0118
5:6s:1l	43	0.0250	0.0113	0.0096
5:8s:1l	57	0.0237	0.0135	0.0098
5:10s:1l	71	0.0213	0.0168	0.0116
6:2s:1l	17	0.0278	0.0161	0.0105
6:4s:1l	33	0.0255	0.0178	0.0091
6:6s:1l	49	0.0242	0.0137	0.0096
6:8s:1l	65	0.0213	0.0192	0.0141
6:10s:1l	81	0.0206	0.01214	0.0158

The comparative results for the best ARIMA and TDNN models with respect to RMSE and MAD for various horizons are given in Table 5. We can see that for both the price series, RMSE and MAD values are

Table 5. Forecasting performance of different models for various horizons

MODEL	1 month ahead		3 months ahead		6 months ahead		12 months ahead	
	RMSE	MAD	RMSE	MAD	RMSE	MAD	RMSE	MAD
				Soybean				
ARIMA	5.43	1.56	29.13	13.22	37.70	35.00	35.35	27.94
TDNN	33.90	22.90	30.00	22.90	32.07	18.36	19.70	17.80
Hybrid	43.00	23.30	52.80	22.70	40.60	23.90	31.50	25.60
				Rapeseed-mustard				
ARIMA	3.35	0.97	25.01	10.89	47.08	30.76	72.59	69.81
TDNN	4.79	1.00	9.20	1.30	9.40	4.60	12.40	8.70
Hybrid	3.46	1.01	7.68	3.53	7.80	2.46	10.38	5.60

Note: All RMSE and MAD values should be multiplied by 10^{-3}.

in general less in neural network model than in ARIMA model, suggesting a better performance of TDNN model. At this juncture, it is worth mentioning that a specific neural network model is selected for each forecast horizon which implies that p and q may vary over forecast horizon. However, we have observed that ARIMA model performs better than TDNN model for a forecast horizon of one month. In general, TDNN model performs better in 6 and 12 months ahead forecasting, while ARIMA models dominate in one month and 3 months forecast horizons. Moreover, for rapeseed-mustard series, the RMSE value pertaining to neural network model is smaller as compared to ARIMA model for all horizons, except one month, suggesting better performance of TDNN which is truly a nonlinear time series data set. Hence, nonlinearity test provides a fairly good indication to post-sample forecast accuracy for neural network models.

Turning Point Evaluation

Several researchers have suggested that RMSE type measures may not be appropriate for nonlinear models as these measures can imply that a nonlinear model is less accurate than a linear one even when former is the true data generating process. In effect, a nonlinear model may generate more variation in forecast values than a linear model, and hence could be unduly penalized for errors that are large in magnitude. Clements and Smith (1997) have argued that the value of nonlinear model forecast may be better reflected by the direction of change. Hence in this study, we have also computed the percentage of forecasts that could correctly predict the direction of monthly price change as part of post-sample forecast accuracy. The direction of change or turning point evaluation is a measure of accuracy related to price forecasts interpreted only in terms of whether agricultural commodity prices will increase or decrease.

With one year of post-sample data, we have 12 one-step ahead forecast errors. The number of forecast errors decreases as the forecast horizon increases, so we have calculated the direction of change only for the forecast horizon of 1 month, 3 months and 6 months with 12, 10 and 7 forecast errors, respectively, as given in Table 6. The implications of the direction of change results of Table 6 are, however, very different from the results based on RMSE. At horizon of 1 month, 3 months and 6 months, the neural network model always had a larger percentage of correct sign than the linear model for all series. The results of Table 6 imply that the relative forecasting performance of both models crucially depends on the manner performance is measured.

Hybrid Model

Turning to the issue of whether the combination of ARIMA and TDNN models performs better than a single model. As mentioned earlier, the combined models are constructed in a sequential manner, with the application of ARIMA model first to the original time series and then its residuals are modelled using neural networks. We have found the optimal structure of neural network for the residual series following the procedure employed for the original series. Table 5

Table 6. Post-sample percentage of forecasts of correct sign

Series	1 month-ahead		3 months-ahead		6 months-ahead	
	ARIMA	TDNN	ARIMA	TDNN	ARIMA	TDNN
Soybean	42	55	46	54	57	60
Rapeseed-mustard	50	67	44	68	49	71

provides the forecasting performance of possible hybrid models in terms of RMSE and MAD values for soybean and rapeseed-mustard for forecasting horizons of 1 month, 3 months, 6 months and 12 months.

The RMSE and MAD values of Table 5 reveal mixed results in post-sample forecast accuracy of hybrid model for the experimental data. We can see from Table 5 that for soybean series, hybrid model in general provides a poor forecast as compared to ARIMA and TDNN models in terms of RMSE and MAD values. The principle underlying the hybrid model is that at the first stage ARIMA will pick up the linear component in the data, while at the second stage, the neural network will model the nonlinear component. In the case of soybean series, after ARIMA was fit at the first stage, the residual was close to random because of its linear nature. In the case of rapeseed-mustard, the hybrid model outperformed both ARIMA and TDNN models consistently across four different time horizons and with both error measures. In nutshell, the empirical results with two real price data sets suggest that the hybrid model performed better than each component model in the case of nonlinear pattern.

Concluding Remarks

The main advantage of univariate time-series forecasting is that it requires data only of the time series in question. First, this feature is advantageous if we are to forecast a large number of price series. Second, this avoids the problem that occurs sometimes with multivariate models; for example, consider a model including import, prices and domestic production. It is possible that a consistent data on import series is available only for a shorter period of time than the other two series, restricting the time period over which the model can be estimated. Third, timeliness of data can be a problem with multivariate models.

This paper has compared the ARIMA and TDNN models in terms of both modelling and forecasting using monthly wholesale price data of two oilseed crops, namely soybean and rapeseed-mustard traded in Indore and Delhi markets of India. The TDNN model in general has provided a better forecast accuracy in terms of conventional RMSE and MAD values as compared to the ARIMA model. It has been found that the evidence of nonlinearity in a series plays a fairly good role in providing a reliable guide to post-sample forecast accuracy of ARIMA and TDNN models in terms of RMSE for these price series. The study has suggested that before adopting any nonlinear model one needs to check whether the series is indeed nonlinear. Moreover, TDNN has performed substantially better than linear models in predicting the direction of change for these series, and hence may be preferred than linear models in the context of predicting turning point, which is more relevant in the case of price forecasting. Such direction of change forecasts are particularly important in economics for capturing the business cycle movements relating to the turning points. Finally, the empirical results with rapeseed-mustard data, which is a true nonlinear pattern, have indicated that the combined model can be an effective way to improve forecasting accuracy achieved by either of the models used independently.

Agricultural price information needs for decision-making at all levels are increasing due to globalization and market integration. This necessitates an effort towards designing a market intelligence system by integrating traditional statistical methods with soft computing techniques like neural network, fuzzy logic, etc. to provide accurate and timely price forecast by taking into account the local information to the farmers, traders and policymakers so that they may make production, marketing and policy decisions well in advance. The decision support system should provide customized advice to individual farmers in view of their local conditions.

Acknowledgements

The authors are grateful to the referee for valuable comments and suggestions in improving the paper.

References

Clements, M.P. and Smith, J. (1997) The performance of alternative methods for SETAR models. *International Journal of Forecasting*, **13:** 463– 475.

Demuth, H. and Beale, M. (2002) *Neural Network Toolbox: User's Guide*. Mathworks, Natic, MA.

Hagan, M. T. and Menhaj, M. (1994) Training feed-forward networks with the Marquardt algorithm. *IEEE Transactions on Neural Networks*, **5**: 989-993.

Haykin, S. (1999) *Neural Networks: A Comprehensive Foundation*, Prentice Hall, New Delhi.

Jha, G.K., Thulasiraman, P. and Thulasiram, R. K. (2009) PSO based neural network for time series forecasting. *Proceedings of the International Joint Conference on Neural Networks.* Atlanta, USA. pp. 1422-1427.

Kuan, C. M. and White, H. (1994) Artificial neural networks: An econometric perspective. *Econometric Reviews*, **13**: 1-91.

McLeod, A.I. and Li, W.K. (1983) Diagonostic checking ARMA time series models using squared residual autocorrelations. *Journal of Time Series Analysis*, **4**: 269-273.

Rojas, I., Valenzuela, O., Rojas, F., Guillen, A., Herrera, L. J., Pomares, H., Marquez, L. and Pasadas, M. (2008) Soft-computing techniques and ARMA model for time series prediction. *Neurocomputing*, **71**: 519-537.

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

Received: February, 2013; Accepted May, 2013