International Journal of Statistics and Applied Mathematics 2023; 8(4): 40-54

International Journal of Statistics and Applied Mathematics

ISSN: 2456-1452
Maths 2023; 8(4): 40-54

https://www.mathsjournal.com
Received: 24-04-2023
Accepted: 03-06-2023

Devra SJ
Department of Agricultural Statistics, Junagadh Agricultural University, Junagadh, Gujarat, India

Patel DV
Department of Agricultural Statistics, Junagadh Agricultural University, Junagadh, Gujarat, India

Shitap MS
Department of Agricultural Statistics, Junagadh Agricultural University, Junagadh, Gujarat, India

Raj SR
Department of Agricultural Statistics, Junagadh Agricultural University, Junagadh, Gujarat, India

Corresponding Author:
Devra SJ
Department of Agricultural Statistics, Junagadh Agricultural University, Junagadh, Gujarat, India

Time series forecasting of price for oilseed crops by combining ARIMA and ANN

Devra SJ, Patel DV, Shitap MS and Raj SR

DOI: https://doi.org/10.22271/maths.2023.v8.i4a.1098

Abstract
Time series modelling and forecasting is a vibrant research field that had attracted the interest of the scientific community in recent decades. Forecasts of agricultural prices are proposed to be useful for farmers, governments, policy makers and agribusiness industries. In this study, an effort is made to compare the forecasting capabilities of well-known linear Auto Regressive Integrated Moving Average (ARIMA) models, Time Delay Neural Network (TDNN) models and Hybrid (ARIMA-TDNN) models using data on monthly wholesale price of four major oilseed crops of India *viz*. groundnut, soybean, sesame and rapeseed and mustard from Jan-2001 to Dec-2021. Finally, the forecasting performance of these models are evaluated and compared by using common criteria's such as; Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE) and percentage of forecasts of correct sign. Results showed that the lowest RMSE and MAE were achieved for the hybrid model than the ARIMA and ANN for all the four crops prices with the exception of MAPE which gave higher value and the percentage of forecasts of correct sign were achieved highest for the hybrid model than others. Key findings revealed that the Hybrid (ARIMA-ANN) model outperformed each individual ARIMA and ANN model, for forecasting of four major oilseed crops price.

Keywords: Oilseed crops, time series forecasting, ARIMA, TDNN, hybrid methodology

1. Introduction
Oilseeds play an important role in Indian agriculture as second major crop (next to cereals) occupying 11 per cent of the total cropped area and 9 per cent of the total agricultural production. Oilseeds grown area in India was (288.18 lakh ha), production (365.65 lakh tonnes) and productivity (1269 kg/ha) during 2020-21. Status in Gujarat area was (33.56 lakh ha), production (61.88 lakh tonnes) and productivity (1844 kg/ha) during 2020-21 (Anon., 2021a) [2]. Agricultural price forecasts are meant to assist farmers, governments, and agribusiness industries in making decisions that influence producers, consumers, and policymakers. As a result, policymakers, businesspeople, traders and farmers are concerned about the ability to reliably estimate the price of oilseeds. There are also few issues, first is the domestic production of oilseeds in India is high but cannot satisfied domestic requirement so need to import the much more quantity from the rest of countries, and second one is the price volatility of oilseeds is also high due to various factors i.e. national market, international market and climatic factor. To overcome these issues, we were undertaken this study to develop an appropriate model for forecasting agricultural price series that exactly reveals the characteristic of series which was needed for the benefits of the traders, policymakers and farmers. Because of the importance of oilseeds and edible oils in the country's economy, the oilseed crops were purposefully chosen for the study.

1.1 Time Series Forecasting
Time series data is a data set that contains a succession of observations on a single phenomenon through time. Time series data includes daily maximum temperatures, weekly interest rates, monthly price statistics, annual crop production and so on. The majority of the time, these observations are made at evenly spaced, discrete time intervals.

Time series forecasting is a type of forecasting in which previous observations of the same variable are gathered and evaluated to create a model that describes the underlying relationship. The time series is then extrapolated into the future using the model. When there is limited information about the underlying data generation process or when there is no appropriate explanatory model that ties the prediction variable to other explanatory factors, this modelling method is highly beneficial.

Time series models are often divided into two groups in the literature: linear and nonlinear models. Linear models are simple, traditional, easy and predict only linear patterns of time series. Exponential Smoothing, Moving Average and ARIMA are some of the linear time series models. Neural network models come under the nonlinear category. Nonlinear models are complex than linear models but flexible and hence used to capture nonlinearity of time series.

1.2 ARIMA Model

Early attempts to study time series, particularly in the 19th century, were generally characterized by the idea of a deterministic world. Yule (1927) [14] was the first to introduce the concept of stochasticity in time series, proposing that every time series can be thought of as the realisation of a stochastic process. Since then, a number of time series methods have been developed based on this simple concept. Slutsky, Walker, Yaglom and Yule were among the first to propose the concept of autoregressive (AR) and moving average (MA) models. The linear forecasting problem of Kolmogorov was formulated and solved using Wold's decomposition theorem (1941). Since then, a large body of research dealing with parameter estimation, identification, model verification and forecasting has arisen in the field of time series; see, for example, Newbold (1983) [12] for an early survey.

The publication of Time Series Analysis: Forecasting and Control by Box and Jenkins (1970) [4] integrated the existing knowledge. Furthermore, the researchers established a three-stage iterative cycle for time series identification, estimate and verification (dubbed the Box–Jenkins technique). The book had a huge impact on modern time series analysis and forecasting theory and practice. The computer revolutionized the way ARIMA models and their extensions were used in many fields of science. However, in the mid-1960s, choosing a model was mostly a matter of the researcher's opinion; there was no technique that could uniquely identify a model. Since numerous strategies and procedures, such as Akaike's information criterion (AIC), Akaike's final prediction error (FPE) and Bayesian information criterion (BIC), have been proposed to add mathematical rigour to the search process of an ARIMA model. The maximum likelihood method is commonly used to estimate the number of parameters in an ARIMA model. If a time series is known to follow a univariate ARIMA model, forecasts using disaggregated observations are at least as good as forecasts using aggregated observations in terms of mean square error (MSE). The technique emphasises the recent past rather than the distant past, hence it is better for short-term prediction than long-term.

The fundamental disadvantage of ARIMA, as previously stated, is that it assumes a linear pattern in time series and hence does not capture nonlinear patterns. Machine learning techniques, particularly Artificial Neural Networks (ANN) models, are an appealing alternative to classic statistical models as it captures an uncertain functional relationship, which is difficult to fit by other models (Darbellay and Slama, 2000) [6].

1.3 Artificial Neural Networks

Artificial neural networks (ANNs) can be beneficial for nonlinear processes with uncertain functional relationships that are difficult to incorporate into a model (Darbellay and Slama, 2000) [6]. The major benefit of ANNs is that they give a very flexible framework for approximating any sort of data nonlinearity. The basic feature of ANNs is that before reaching the output variable the inputs and dependent variables are filtered via one or more hidden layers, each of which contains hidden units or nodes. Hence the ultimate output is linked to the intermediate output.

Both forecasting scholars and practitioners were considered ANNs to be an appealing alternative tool. ANNs are valuable for forecasting tasks due to a number of distinctive characteristics. First, unlike traditional model-based methods, ANNs are data-driven self-adaptive methods with a little priori assumption about the models for the problem at hand. Secondly, ANNs have the ability to generalize. And thirdly, Universal functional approximators are ANNs.

There are two methods that neural networks can simulate time series data. The first method is to establish recurrent connections between output nodes and the layer above (Elman, 1990) [8]. The second option is to add buffers to the nodes outputs (Haykin, 1999) [9]. A well-known illustration for the later models is the time-delay neural network (TDNN), which will be used for this study. Each layer in a TDNN is connected to the buffered output of the layer before it, allowing it to relate current input to previous values. TDNN is commonly used in time series analysis due to its ease of implementation (Zhang and Qi, 2005) [16].

1.4 Combining Forecasts

For the past three decades, researchers have looked into combining, mixing, or pooling quantitative forecasts obtained from a variety of time series approaches and data sources. There have been numerous suggestions for integrating different models. For time series forecasting, a hybrid technique combining ARIMA and ANN has been developed in the literature.

The hybrid model's motivation was stemed from the following perspectives. To begin with, determining whether a time series under study is created from a linear or nonlinear underlying process, or whether one method is more effective than the other in out-of-sample forecasting performance, is typically challenging in reality. As a result, forecasters were found it challenging to select the best strategy for their particular circumstances. Typically, several different models are tested and the one that produces the most accurate results is chosen. Due to various potential impacting factors such as sampling variance, model uncertainty and structural change, the final selected model is not necessarily the best for future usage. The difficulty of model selection can be made easier with a little more effort by integrating multiple strategies. Second, time series in the actual world are rarely pure linear or nonlinear. There are often both linear and nonlinear patterns in them. If this is the case, neither ARIMA nor ANNs can be used to model and forecast time series because the ARIMA model can't handle nonlinear relationships and the neural network model can't handle both linear and nonlinear patterns equally well. As a result, complicated autocorrelation structures in data can be more correctly described by merging ARIMA and ANN models. Finally, the forecasting literature almost uniformly agreed that no single strategy was best in every case. This is owing to the fact that real-world problems are frequently complicated and no single model will be able to represent all patterns equally well. Many empirical investigations demonstrated that mixing multiple different models can typically enhance predicting accuracy over a single model without the necessity to determine the "real" or "best" model. As a result, integrating several models may improve forecasting performance by increasing the possibility of capturing distinct patterns in the data. Furthermore, the integrated model may be more robust in the face of possible data structural changes. The analysis of data for this study was carried out by using R 4.2.0 software.

2. Methodology

2.1 Time Series Forecasting

Visualizing the graph of time series were a key step in data exploration. The curve of a time series with a level, trend and seasonal component can be visualized to anticipate the time series. The time plot is a graph that depicts the variation of data over time.

2.2 Stationarity Test

The time series was said to be stationary if the mean and variance of the stochastic time series data remain constant across time. It denotes that the data has neither increased nor decreased. Along the time axis, the data must be roughly horizontal. The Augmented Dickey Fuller (ADF) Test and the Phillips-Perron Test (PP test) are two statistical tests for determining the stationarity of time series. We were used ADF test because it had been utilized for a larger and more complicated set of time series models. The ADF statistic used in the ADF test is a negative number. The more negative it is, the stronger the rejection of the hypothesis that there is a unit root.

2.2.1 Augmented Dickey Fuller (ADF) Test

The ADF test given by Dickey and fuller (1979) [7] consists of estimating the following regression equation:

$$\Delta y_t = a_1 + a_2 t + \delta y_{t-1} + \sum_{i=1}^{h} b_i y_{t-i} + e_t \qquad (1)$$

Where, $\Delta y_t = y_t - y_{t-1}$, a_1, a_2 and b_i are the parameters of regression model, h is length of the lag, $\delta = \rho - 1$ and $-1 \leq \rho \leq 1$ and ρ is correlation between consecutive variable.

In ADF test, whether the time series having a unit root or not, which means the time series under consideration will be stationary or not is tested.

H_0: $\delta = 0$, Time series is non-stationary
H_1: $\delta \neq 0$, Time series is stationary

Based on p-value of the test, it was decided that whether the data are stationary or not.

2.3 ARIMA Model

The future value of a variable is supposed to be a linear function of numerous prior observations and random errors in an autoregressive integrated moving average model. That is, the underlying process that generate the time series has the form

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \ldots + \phi_p y_{t-p} + \varepsilon_t - \theta_1 \varepsilon_{t-1} - \theta_2 \varepsilon_{t-2} - \ldots - \theta_q \varepsilon_{t-q} \qquad (2)$$

Where, c is constant term related to mean, y_t and ε_t are the actual observation and random error at time period t, respectively; ϕ_i (i = 1, 2, …, p) and θ_j (j = 1, 2, …, q) are model parameters. *p* and *q* are integers and often referred to as order of the model. Random errors ε_t are assumed to be NID $(0, \sigma^2)$.

Several essential special examples of the ARIMA family of models are included in Equation (2). Eq. (2) becomes an AR model of order p if q= 0. When p=0, the model is reduced to a q-order MA model. One of the most important aspects of developing an ARIMA (p, d, q) model was determining the right model order (p, d, q), where d is the order of differencing. We use the procedure to analyze and forecast agricultural prices which had five steps, which were described below:

Step 1: Determine whether the time series is stationary

The series under study must be stationary. The statistical features of a stationary time series, such as the mean and variance, remain constant over time. As indicated earlier the presence of stationarity in the data can be obtained by simply plotting the raw data or by plotting the autocorrelation and partial autocorrelation function.

To test stationarity, statistical tests the Dickey-Fuller test was used. Some kinds of transformation, such as logarithms or square root transformation, could attain stationarity in variance. If there is a time trend or seasonality in the series, or if there is any nonstationary pattern, the series was differenced repeatedly until it becomes stationary. In this study, we were applying differencing for making the series stationary.

Step 2: Identify the model

Once the time series has been stabilised, candidate ARIMA models were identified. Multiple ARIMA models that closely suit the data could be identified after getting the Auto Correlation Function (ACF) and Partial Auto Correlation Function (PACF). The order of tentative models could be determined during the identification process by looking for significant autocorrelation and partial autocorrelation functions.

Step 3: Estimate the model parameters
Step 4: Perform diagnostic checking
Step 5: Select the most suitable ARIMA model

The most suitable ARIMA model was selected using the smallest Akaike Information Criterion (AIC) (Akaike, 1977) [1] or Schwarz-Bayesian Criterion (SBC) (Schwarz, 1978) [13]. AIC is given as:

$$\text{AIC} = (-2 \log L + 2m) \qquad (4)$$

Where, m= p + q and L is the likelihood function for the model. SBC was also used as an alternative to AIC which is given as:

$$\text{SBC} = \log\sigma^2 + (m \log n) / n \qquad (5)$$

Here, σ^2 is error variance and n is the number of observations. If the model was not being adequate, a new tentative model was identified and the above steps were repeated. Diagnostic information may help to suggest alternative model(s). These steps of model building process are typically repeated several times until a satisfactory model was finally selected.

The final model can then be used for prediction purposes. Schematic representation of Box-Jenkins methodology showed in Figure 1.

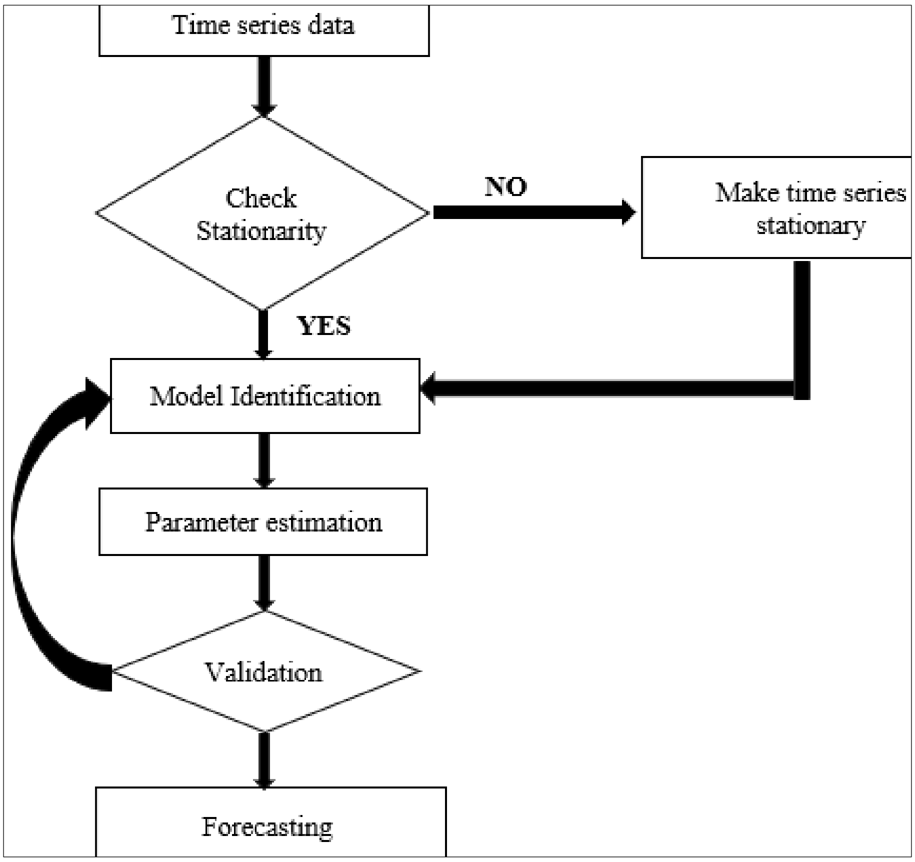

Fig 1: Schematic representation of Box-Jenkins methodology (Box and Jenkins, 1970) [4]

2.4 Non-Linearity Test

Before applying neural network model to the residual of the time series, the linearity was checked. Here, BDS test (Broock *et al*., 1996) [5] was used for checking non-linearity. BDS test statistics, which was calculated as follows:

First, the m-histories' of the data x_t^m=(x_t,x_(t+1),…,x_(T-m+1)) are calculated for t = 1, 2, …, T-m for some integer embedding dimension m ≥ 2. To converts the series of scalars into a series of vectors of overlapping series. Select m value in such a way that the embed time series into m-dimensional vectors, by taking each m successive points in the series.

$$x_1^m = (x_1, x_2, \dots, x_m)$$
$$x_2^m = (x_2, x_3, \dots, x_{m+1})$$
$$x_{T-m}^m = (x_{T-m}, x_{T-m+1}, \dots, x_T) \quad (6)$$

The correlation integral was then computed, which counts the proportion of points in m-dimensional hyperspace that are within a distance (∈) of each other.

$$C_m T(\in) = \frac{2}{(T-m+1)} \sum_{t\langle m} I_e\,(x_t^m - x_s^m) \quad (7)$$

Where, I_e is an indicator function that equals one if $\|x_t^m - x_s^m\| \langle \in$ and zero otherwise, and $\|.\|$ denotes the sup. norm. BDS show that under the null hypothesis that the observed x_t are *iid*, then $C_{m,I}(\in) - C_{I,T}(\in)^m$ with probability one as the sample size tends to infinity and ∈ tends to zero. The BDS test statistic, which has a limiting standard normal distribution, then, follows as:

$$W_{m,I}(\in) = \frac{T^{1/2} C_{m,I}(\in) - C_{I,T}(\in)^m}{\sigma_{m,I}(\in)} \quad (8)$$

Where, $\sigma_{m,I}(\in)$ will be estimated by

$$\sigma_{m,I}(\in) = \frac{6 \sum_{t\langle s\langle r} h_e(\in)\, x_t^m, x_r^m, x_s^m}{[(T-m+1)(T-m)(T-m-1)]} \quad (9)$$

And,

$$h\in(i,j,k) = \frac{[\,I\in(i,j)\, I\in(j,k)\, I\in(i,k)\, I\in(k,j) + I\in(j,i)\, I\in(i,k)]}{3} \quad (10)$$

Two parameters are to be chosen: the value of ∈ (the radius of the hyper-sphere which determines whether two points are 'close' or not) and m (the value of the embedding dimension). Broock *et al.* (1996) [5] recommend that ∈ is set to between half and three halves the standard deviation of the actual data and m is set in line with the number of observations available. BDS test is a two-tailed test, we would reject the null hypothesis if the BDS test statistic was greater than or less than the critical values. Usually, 5% level of significance was considered for hypothesis testing and the hypothesis are as follows:

H_0: Residual is linear
H_1: Residual is nonlinear

2.5 Artificial Neural Network (ANN) Model

The main disadvantage of the ARIMA model is that it ignores the nonlinear component of time series data, which is bounded by residuals in nonlinear time series. Machine learning techniques were increasingly appropriate in dealing with such situations. It is obvious from the literature that the hybrid model will perform better than the individual model in many circumstances. In hybrid model, a combination of ARIMA & ANN was used to predict linear & non-linear components of a time series, simultaneously.

The foremost important thing in model fitting was dividing the dataset into training and testing. Most of the researchers followed particular procedures for splitting dataset. It was used 80 percent or 90 percent for training and 20 percent or 10 percent for testing. In this study, last 24 months data (2 years) were used as testing. From the literature, the fact confirmed was that more the number of training observations resulted into the better fit of model.

2.5.1 Architecture of ANN

The ANN model was made up of multiple interconnected units called neurons. One input layer, one output layer and one or more hidden layers made up a standard ANN model. Each layer has nodes, which include input nodes, output nodes and hidden nodes. Each layer's output is the weighted sum of its inputs, which is a typical feature.

$$u_i = \sum_j w_{ij} * output_j + v_i \quad (11)$$

Where, u_i is the value of net input of i^{th} node, w_{ij} is the weights connecting j^{th} to i^{th} neuron, v_i denotes bias for i^{th} node. Many nonlinear functions are available in literature to be used as an activation function by researchers. The two most widely used activation functions are identity function and sigmoid function.

$$\emptyset(netinput) = \begin{cases} 1, netinput \geq 0 \\ 0, otherwise \end{cases} \quad (12)$$

$$g(netinput) = \frac{1}{1+e^{-netinput}} \quad (13)$$

Where $\emptyset$ and g represents identity and logistic activation function respectively.

2.6 Time-Delay Neural Network

The TDNN model with one hidden layer was written as *I:Hs:Ol*, where *I* is the number of input nodes, *H* is the number of hidden nodes, *O* is the number of output nodes, *l* indicates the linear transfer function and *s* denotes the logistic sigmoid transfer function.

For modelling the residual component, a feed-forward Time-Delay Neural Network (TDNN) with a single hidden layer was used as a multi-scale learning tool in this study. The general expression for a TDNN with single hidden layer is given by (Jha and Sinha, 2012) [10]:

$$y_{t+1} = g\left(\sum_{j=0}^{q} \alpha_j f\left(\sum_{i=0}^{p} \beta_{ij} y_{t-i}\right)\right) \quad (14)$$

Where $y_{t+1} = \ln(y_{t+1}/y_t)$ is the predicted value for y_t at time t, α_j (j=0, 1, 2,., q) and β_{ij} (i=0, 1, 2,., p; j=1, 2,., q) are the model parameters, p is number of input layer nodes, q is the number of hidden layer nodes, f and g denote the activation function at hidden and output layer respectively and y_{t-i} is the i^{th} input (lag) of the model. There is no formula for calculating the number of layers and nodes. These parameters are discovered using the trial-and-error method, which entails running tests using given data.

Many studies have concluded that a neural network model with a single hidden layer was sufficient to accurately mimic any complex nonlinear function. The selection of an activation function is also a challenging undertaking because it is one of the most important factor to consider when employing a neural network model.

A single hidden layer and a single input layer were utilized in this study to forecast future values based on previously recorded values. The logistic sigmoid transfer function is one of the most common activation functions for adding nonlinearity to a model. Schematic presentation of TDNN with one hidden layer is shown in Figure 2

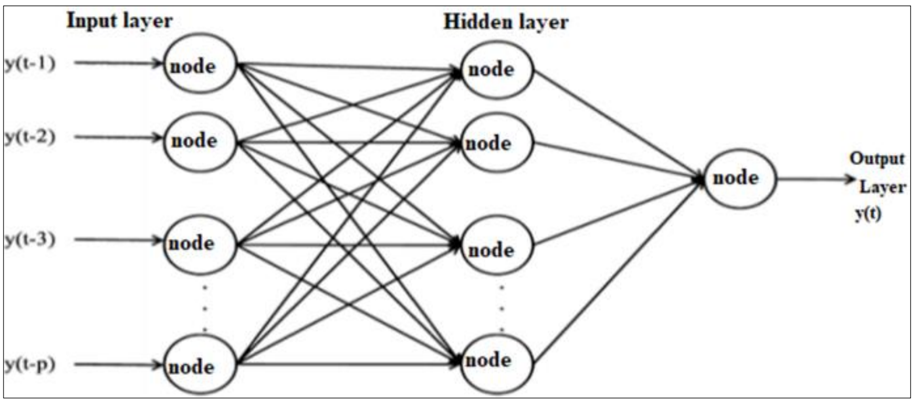

Fig 2: Time-Delay Neural Network (TDNN) with one hidden layer

2.7 Hybrid Model (Zhang, 2003) [15]

Both ARIMA and ANN models have achieved successes in their own linear or nonlinear domains. However, none of them is a universal model that is suitable for all circumstances. The approximation of ARIMA models to complex nonlinear problems may not be adequate. On the other hand, using ANNs to model linear problems have yielded mixed results. For example, when there are outliers or multicollinearity in the data, neural networks can significantly perform better to attain the linearity of time series. It was also established that the performance of neural networks for linear regression problems depend on the sample size and noise level. So, it is not appropriate to apply ANNs blindly to any type of data. The real-world time series data may contain both linear and nonlinear components so a hybrid methodology that has both linear and nonlinear modelling capabilities can be a good strategy for practical application. By combining different models, different aspects of the underlying patterns may be captured.

A hybrid model comprising a linear and nonlinear component will be employed in this experiment.

$$y_t = L_t + N_t \quad (15)$$

Where, y_t is the time series data, L_t is linear AR component and N_t is the nonlinear component. The ARIMA model is applied to the data series to fit the linear part. Then the residuals are modeled using neural networks. Let r be the residual of the linear component, then

$$r_t = y_t - \hat{L}_t \quad (16)$$

Where, $\hat{L}_t$ is the estimate of the linear AR component. For nonlinear components, we apply neural networks:

$$\hat{r}_t = f(r_{t-1}, r_{t-2}, \ldots, r_{t-p}) \quad (17)$$

Where, p is the number of input delays and f is the nonlinear function.
So the combined forecast will be,

$$y_t = \hat{L}_t + \hat{r}_t + \varepsilon_t \quad (18)$$

Where, ε_t is the error of the combined model. Since ARIMA models cannot model nonlinearity, it is assumed that the residuals of the linear component will contain nonlinear component which would be modelled using neural networks. In this way hybrid model is exploiting the strength of both components. To improve overall modelling and forecasting performance, it may be useful to model linear and nonlinear patterns independently using various models and then combine the forecasts. Schematic representation of proposed hybrid model is shown in Figure 3.

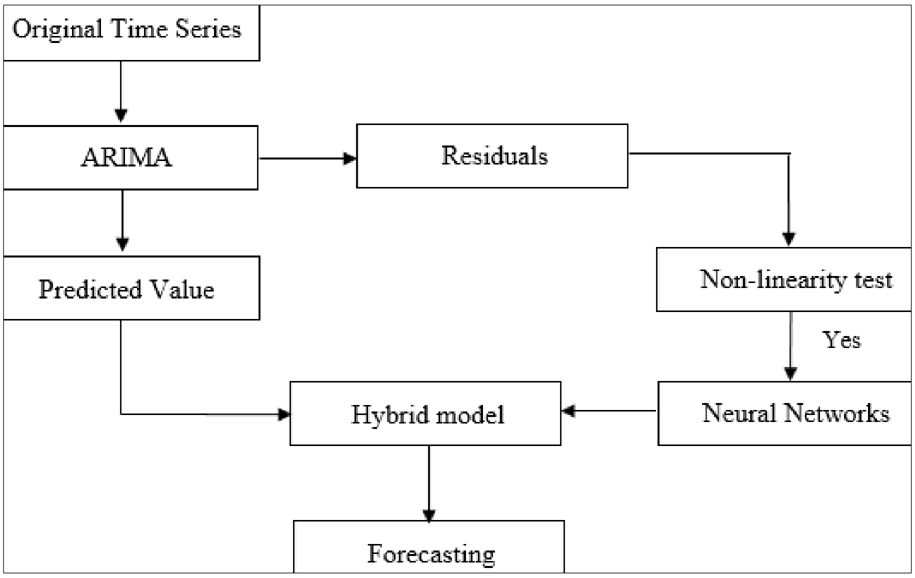

Fig 3: Schematic representation of proposed hybrid methodology

2.8 Forecast Evaluation Methods

Comparisons of the ARIMA and neural network models predicting abilities were made by using three criteria based on error terms. The first one is root mean square error (RMSE), which measures the overall performance of a model. The formula for RMSE is given by,

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(y_t - \hat{y}_t)^2} \quad (19)$$

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 observations used in predictions.
The second criterion is the mean absolute error (MAE). It is a measure of average error for each point forecast made by the two methods. MAE is given by,

$$\text{MAE} = \frac{1}{n}\sum_{t=1}^{n}|y_t - \hat{y}_t| \quad (20)$$

And the third one is mean absolute percentage error (MAPE), MAPE is given by,

$$\text{MAPE} = \frac{1}{h}\sum_{t=1}^{h}\frac{|e_t|}{y_t} \times 100 \quad (21)$$

Where, y_t is the time series, h is the forecast horizon and e_t is the residuals of the time series $e_t = y_t - \hat{y}_t$.

2.9 Turning Point Evaluation

Several researchers have suggested that RMSE type measures, such as RMSE, MAE, and MAPE, may not be appropriate for nonlinear models because they can imply that a nonlinear model is less accurate than a linear one even when the former is the true data generating process. In effect, a nonlinear model may generate more variation in forecast values than a linear model, and thus may be overly penalized for large magnitude errors. As part of the forecast accuracy, in this study we calculated the percentage of forecasts that correctly predicted the direction of monthly price change.

3. Empirical results

3.1 Data

The data of monthly wholesale price (Rs./q) for four major oilseed crops of India *viz.* groundnut, soybean, sesame and rapeseed and mustard were collected from Centre for Monitoring Indian Economy Pvt. Ltd. (CMIE) for period Jan. 2001 to Dec. 2021 (Anon., 2021b) [3].

3.2 Discussion

Visualisation of data was the foremost process of time series analysis. So the time plot that shows the changes in data over time was used. The presence of components such as level (Cyclic + Irregular variation), trend and seasonality in dataset can also be identified from the time plot. The characteristics of price series like stationarity and linearity can also be visualized by time plot. The Figure 4(a), (b), (c) & (d) presented the time plot of groundnut, soybean, sesame & rapeseed and mustard prices. Descriptive statistics of the price series used in the experiments are presented in Table 1.

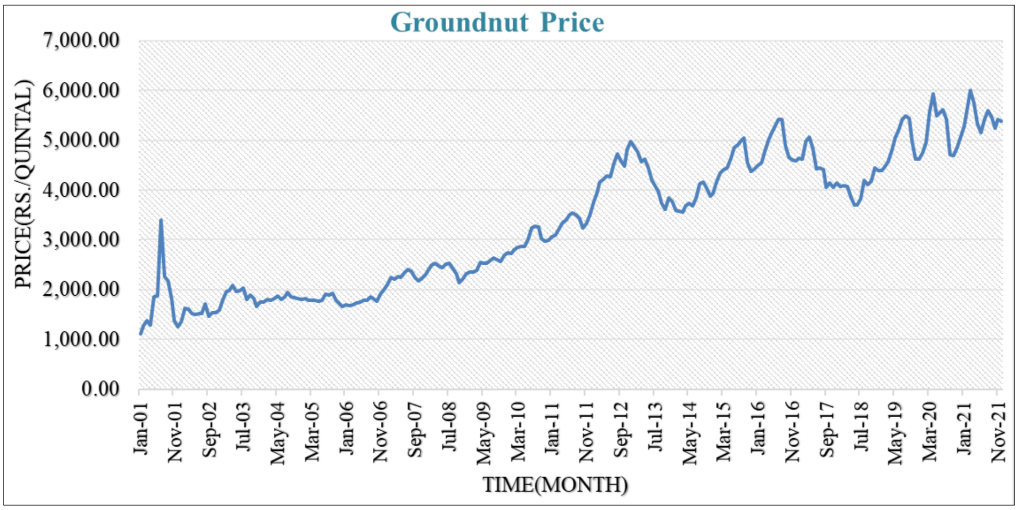

Fig 4 a): Time plot for groundnut price

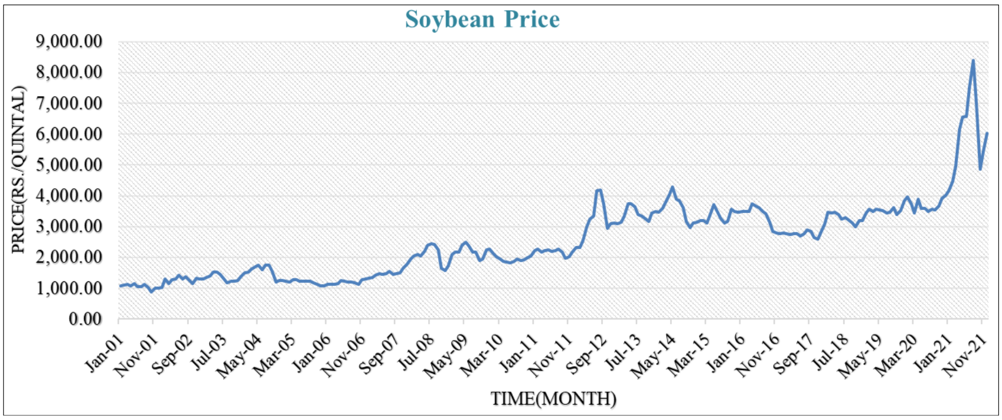

Fig 4 b): Time plot for soybean price

Fig 4 c): Time plot for sesame price

Fig 4 d): Time plot for rapeseed and mustard price

Table 1: Descriptive statistics of the price series used in the experiments

Statistics	Groundnut	Soybean	Sesame	Rapeseed and mustard
Observations	252	252	252	252
Mean (Rs./q)	3344.41	2550.62	5633.92	2878.94
Median (Rs./q)	3396.95	2298.15	5279.70	2633.70
Maximum (Rs./q)	5996.90	8389.20	12119.90	7348.40
Minimum (Rs./q)	1105.00	875.00	1150.00	899.50
Standard Deviation (Rs./q)	1342.96	1252.02	2697.76	1249.35
Skewness	0.11	1.22	0.35	0.99
Kurtosis	-1.38	2.79	-0.90	1.44

Augmented Dickey Fuller (ADF) test was used to test stationarity of time series. Results presented in Table 2 revealed the test result of groundnut, soybean, sesame and rapeseed and mustard price. Which showed that original groundnut price was stationary while for soybean, sesame and rapeseed and mustard original price was non-stationary, so we apply first differencing to it after differencing it became stationary.

Table 2: Stationarity test for price series used in the experiments

Data	Series	ADF	
		t-statistic	p-value
Groundnut Prices	Original	-3.49	0.04
Soybean Prices	Original	-2.31	0.45
	1st Difference	-7.46	<0.01
Sesame Prices	Original	-2.86	0.21
	1st Difference	-6.86	<0.01
Rapeseed and Mustard prices	Original	-1.49	0.79
	1st Difference	-6.37	<0.01

The order of Moving Average (MA) model can be determined using ACF plot by visualizing spikes of the plot and the order of Auto Regressive (AR) model can be determined by using PACF plot. Figure 5(a), (b), (c), (d) showed ACF-PACF plot for price series used in the experiments

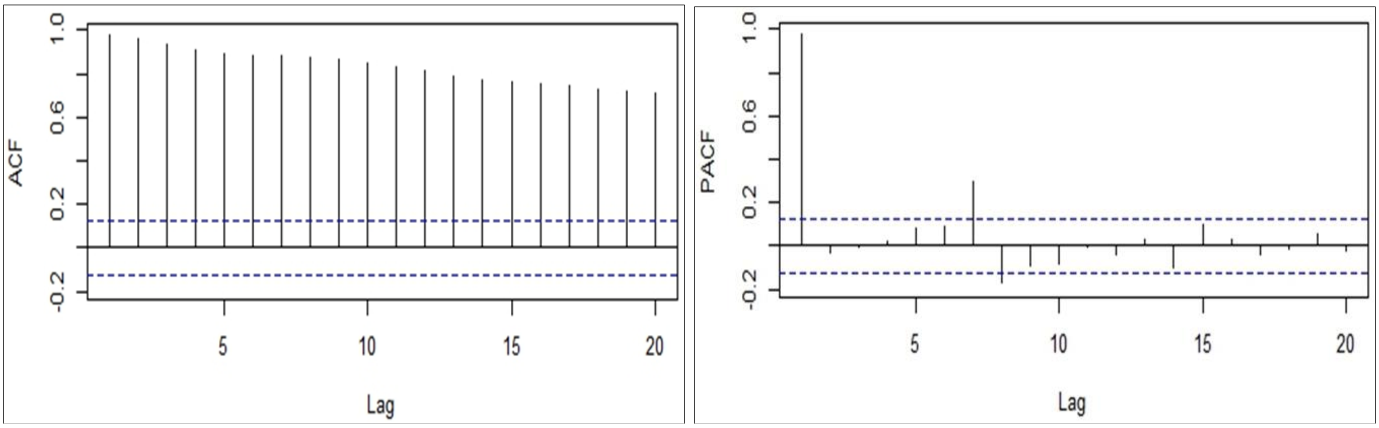

Fig 5 a): Groundnut price

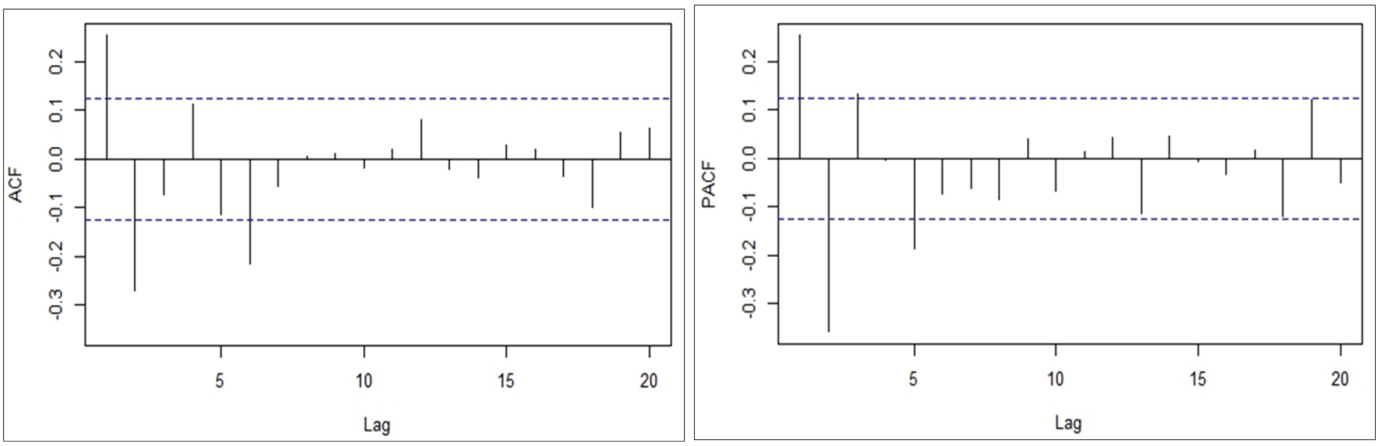

Fig 5 b): Soybean price

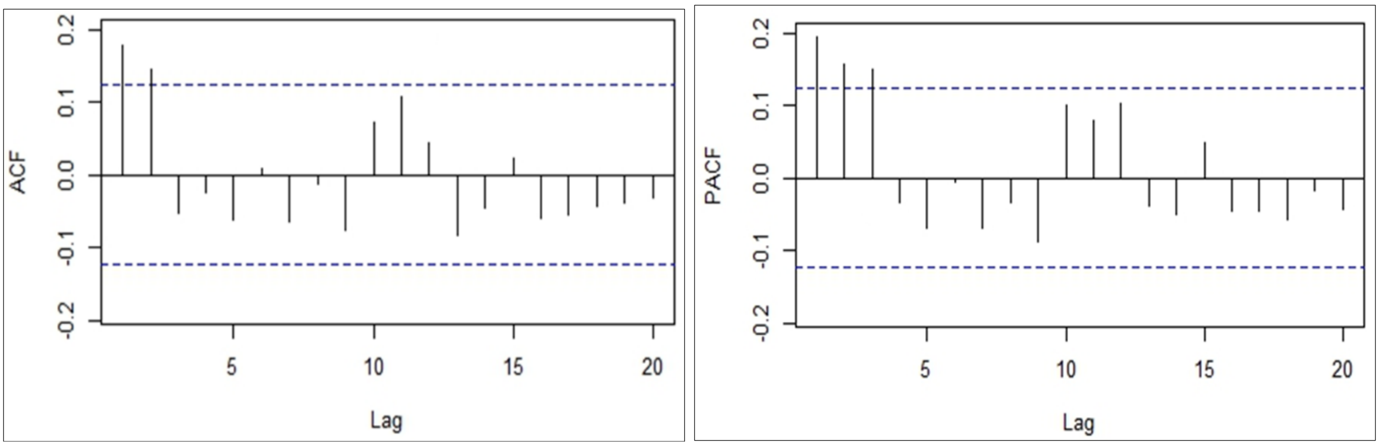

Fig 5 c): Sesame price

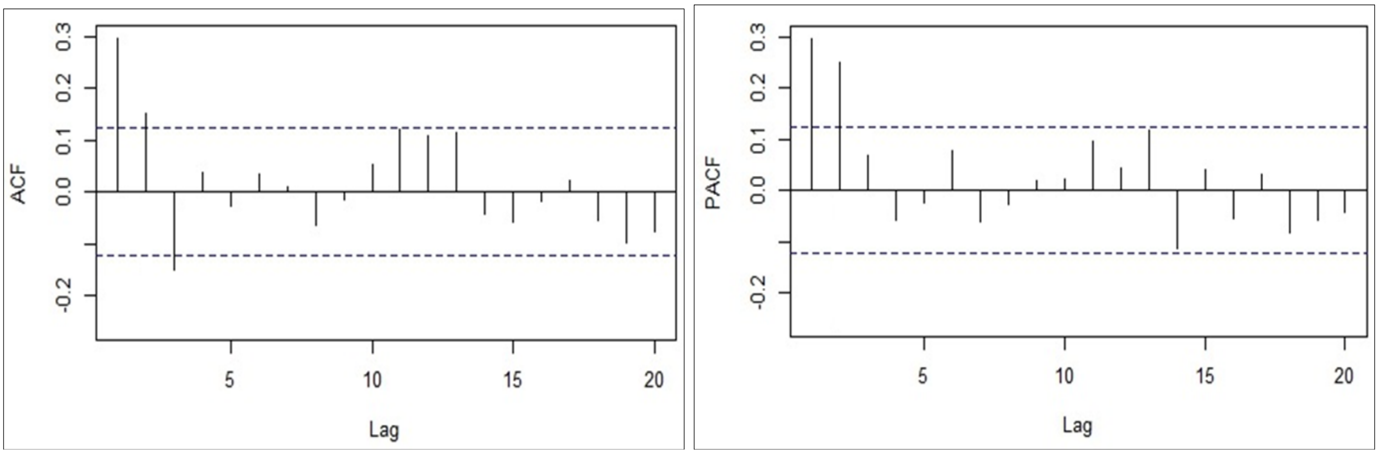

Fig 5 d): Rapeseed and mustard price

After visualizing ACF-PACF plot and considering stationarity test of different price series the possible ARIMA structures for all prices presented in Table 3. Based on the lowest AIC and BIC values we obtain the best ARIMA model for each series. We select ARIMA (1,0,1), ARIMA (3,1,1), ARIMA (1,1,1) and ARIMA (1,1,1) model for groundnut, soybean, sesame and rapeseed and mustard price, respectively.

Table 3: Forecasting performance of ARIMA (p,d,q) model for different price series

ARIMA Model	AIC	BIC
Groundnut		
1,0,1	3413.17	3427.28
1,0,2	3413.58	3431.22
1,0,3	3411.31	3432.48
7,0,1	3408.68	3435.98
7,0,2	3406.47	3438.29
7,0,3	3405.99	3439.35
	Soybean	
1,1,1	3478.21	3488.78
3,1,1	3465.01	3482.64
	Sesame	
1,1,1	3911.48	3922.06
1,1,2	3914.32	3928.42
2,1,1	3914.33	3928.43
2,1,2	3913.49	3931.12
3,1,1	3914.67	3932.29
3,1,2	3915.38	3936.53
	Rapeseed and Mustard	
1,1,1	3218.22	3228.79
1,1,2	3219.17	3233.27
2,1,1	3217.87	3231.97
2,1,2	3215.87	3233.49

Table 5, 6, 7 & 8 summarized forecasting performance of various TDNN models for groundnut, soybean, sesame and rapeseed and mustard, respectively in terms of training and testing RMSE, MAE, MAPE. Table 5 and 7 showed that out of 24 neural network structures, a neural network model with three input nodes and three hidden nodes (*3:3s:1l*) performs better than other competing models as it had least RMSE, MAE and MAPE values with reasonable number of parameters for groundnut and sesame prices, respectively. In case of soybean prices, a neural network with two input nodes and three output nodes (*2:3s:1l*) performs better and for rapeseed and mustard prices, a neural network with four input nodes and two hidden nodes (*4:2s:1l*) performs better due to lower RMSE, MAE and MAPE value with fair numbers of parameter as presented in Table 6 and 8, respectively.

Table 5: Forecasting performance of TDNN models for groundnut prices

Model	No. of Parameters	RMSE		MAE		MAPE	
		Training	Testing	Training	Testing	Training	Testing
1:1s:1l	4	191.07	254.41	118.69	179.67	4.06	3.36
1:2s:1l	7	190.53	224.62	118.12	173.04	4.03	3.23
1:3s:1l	10	190.06	200.09	117.96	156.29	4.03	2.92
1:4s:1l	13	188.67	199.15	117.71	156.01	4.02	2.90
1:5s:1l	16	187.67	190.76	117.57	145.99	4.01	2.73
1:6s:1l	19	187.15	180.73	117.37	139.26	4.00	2.60
2:1s:1l	5	191.48	216.11	119.53	173.99	4.09	3.26
2:2s:1l	9	170.10	172.65	109.45	121.46	3.83	2.28
2:3s:1l	13	161.34	127.89	103.47	98.30	3.60	1.83
2:4s:1l	17	158.83	80.43	101.36	68.20	3.54	1.26
2:5s:1l	21	157.71	69.43	100.12	51.73	3.52	0.95
2:6s:1l	25	156.12	54.11	98.01	38.23	3.42	0.69
3:1s:1l	6	189.48	212.34	119.85	162.92	4.09	3.05
3:2s:1l	11	160.14	160.13	106.87	110.43	3.69	2.08
3:3s:1l	16	136.42	74.17	99.56	47.82	3.45	0.90
3:4s:1l	21	129.66	43.69	93.59	30.95	3.29	0.58
3:5s:1l	26	122.90	25.16	90.25	14.52	3.16	0.28
3:6s:1l	31	117.28	9.07	86.90	5.53	3.05	0.10
4:1s:1l	7	188.97	201.46	119.12	149.98	4.04	2.83
4:2s:1l	13	160.38	109.84	107.95	81.50	3.69	1.54
4:3s:1l	19	137.25	47.68	99.92	37.19	3.44	0.69
4:4s:1l	25	121.68	25.47	90.46	15.98	3.13	0.29
4:5s:1l	31	115.46	4.60	86.01	2.49	2.96	0.04
4:6s:1l	37	110.50	1.18	82.92	0.61	2.83	0.01

Table 6: Forecasting performance of TDNN models for soybean prices

Model	No. of Parameters	RMSE		MAE		MAPE	
		Training	Testing	Training	Testing	Training	Testing
1:1s:1l	4	160.24	580.44	110.01	341.14	4.98	6.49
1:2s:1l	7	158.93	494.45	107.87	314.78	4.87	5.98
1:3s:1l	10	157.42	373.66	107.74	249.15	4.86	4.82
1:4s:1l	13	156.38	230.48	107.55	184.47	4.85	3.80
1:5s:1l	16	154.87	201.15	106.95	154.10	4.84	3.20
1:6s:1l	19	153.42	195.12	105.94	149.52	4.81	3.11
2:1s:1l	5	151.79	573.74	109.35	333.78	5.07	6.28
2:2s:1l	9	146.56	339.93	106.18	261.02	4.87	4.94
2:3s:1l	13	142.08	157.92	103.38	134.68	4.71	2.74
2:4s:1l	17	139.71	114.96	101.27	90.85	4.59	2.15
2:5s:1l	21	137.34	106.29	99.57	76.54	4.51	1.91
2:6s:1l	25	136.63	85.54	98.86	53.48	4.48	1.39
3:1s:1l	6	151.20	587.19	109.89	301.63	5.11	5.53
3:2s:1l	11	143.78	212.60	104.47	154.62	4.82	3.09
3:3s:1l	16	138.34	99.36	101.23	78.47	4.59	1.69
3:4s:1l	21	135.28	55.68	100.09	32.82	4.54	0.79
3:5s:1l	26	132.06	46.31	97.43	23.78	4.43	0.59
3:6s:1l	31	128.92	41.62	94.95	20.08	4.32	0.52
4:1s:1l	7	151.37	488.11	109.79	276.77	5.08	5.16
4:2s:1l	13	142.93	212.75	103.82	140.74	4.78	2.75
4:3s:1l	19	131.45	66.78	98.49	43.24	4.58	1.00
4:4s:1l	25	124.34	46.23	94.01	23.72	4.33	0.61
4:5s:1l	31	113.55	46.15	87.38	27.48	4.08	0.62
4:6s:1l	37	110.93	32.00	85.16	14.58	3.98	0.38

Table 7: Forecasting performance of TDNN models for sesame prices

Model	No. of Parameters	RMSE		MAE		MAPE	
		Training	Testing	Training	Testing	Training	Testing
1:1s:1l	4	576.88	412.46	329.91	336.37	7.60	3.97
1:2s:1l	7	570.46	391.93	326.26	320.77	7.48	3.78
1:3s:1l	10	565.40	381.60	322.75	312.94	7.45	3.70
1:4s:1l	13	563.16	321.48	322.45	264.38	7.45	3.17
1:5s:1l	16	560.56	316.07	322.27	255.62	7.43	3.06
1:6s:1l	19	559.64	284.60	322.04	222.94	7.42	2.69
2:1s:1l	5	575.56	406.13	336.98	350.38	7.67	4.15
2:2s:1l	9	510.35	321.04	318.31	256.86	7.18	3.05
2:3s:1l	13	481.89	245.02	305.66	183.44	6.74	2.19
2:4s:1l	17	474.74	179.17	299.84	132.40	6.54	1.58
2:5s:1l	21	469.04	103.78	294.41	73.66	6.42	0.89
2:6s:1l	25	461.53	87.72	288.96	62.17	6.37	0.75
3:1s:1l	6	576.37	361.85	339.03	283.93	7.72	3.39
3:2s:1l	11	535.49	253.34	328.36	189.08	7.52	2.27
3:3s:1l	16	470.82	153.94	307.34	106.57	6.75	1.29
3:4s:1l	21	461.43	82.11	294.47	56.14	6.58	0.67
3:5s:1l	26	450.00	28.75	285.16	20.43	6.36	0.24
3:6s:1l	31	442.33	14.01	279.06	8.28	6.22	0.10
4:1s:1l	7	577.29	341.21	339.07	266.23	7.69	3.16
4:2s:1l	13	514.46	236.82	318.38	184.70	7.24	2.19
4:3s:1l	19	467.16	110.33	294.81	79.95	6.54	0.94
4:4s:1l	25	449.13	32.84	281.76	21.66	6.28	0.25
4:5s:1l	31	432.21	15.49	271.93	10.16	6.06	0.13
4:6s:1l	37	414.66	6.98	260.75	3.74	5.87	0.04

Table 8: Forecasting performance of TDNN models for rapeseed and mustard prices

Model	No. of Parameters	RMSE		MAE		MAPE	
		Training	Testing	Training	Testing	Training	Testing
1:1s:1l	4	142.43	197.86	86.04	158.00	3.81	2.94
1:2s:1l	7	141.48	180.53	84.54	138.32	3.73	2.58
1:3s:1l	10	139.31	153.99	84.01	117.92	3.70	2.25
1:4s:1l	13	137.93	110.52	83.68	87.62	3.69	1.72
1:5s:1l	16	137.26	91.11	83.35	74.09	3.67	1.48
1:6s:1l	19	137.42	91.31	83.23	72.49	3.67	1.43
2:1s:1l	5	136.62	193.28	80.48	143.59	3.58	2.59
2:2s:1l	9	132.42	153.46	76.97	118.18	3.45	2.25
2:3s:1l	13	128.16	85.34	73.45	65.27	3.29	1.29
2:4s:1l	17	125.02	70.71	72.13	57.04	3.23	1.15
2:5s:1l	21	122.01	53.58	71.08	44.56	3.18	0.88
2:6s:1l	25	120.77	45.57	70.99	37.71	3.19	0.75
3:1s:1l	6	131.46	176.11	80.55	139.48	3.54	2.59
3:2s:1l	11	122.83	95.57	74.83	73.61	3.34	1.34
3:3s:1l	16	112.48	50.59	66.60	42.99	2.91	0.80
3:4s:1l	21	109.89	30.27	65.66	23.32	2.84	0.46
3:5s:1l	26	107.20	14.72	64.29	10.31	2.78	0.20
3:6s:1l	31	103.96	12.04	63.25	7.79	2.73	0.15
4:1s:1l	7	131.08	176.81	80.12	139.79	3.55	2.57
4:2s:1l	13	119.58	66.05	72.46	50.07	3.22	0.95
4:3s:1l	19	110.67	34.82	66.69	24.21	2.91	0.47
4:4s:1l	25	101.82	21.38	63.56	15.45	2.71	0.30
4:5s:1l	31	100.09	15.40	62.54	9.35	2.67	0.19
4:6s:1l	37	96.83	12.18	60.16	6.77	2.56	0.14

Table 9 shows the result of BDS (Brock, Dechert and Scheinkman) nonlinearity test used for best fitted ARIMA model residuals. It revealed that null hypothesis (H_0: Residuals are linear) was rejected as the test result was found to be highly significant for the residuals of best fitted ARIMA model after considering two and three embedding dimension for each price data.

Table 9: BDS nonlinearity test for different series

Series	Embedding dimension				Conclusion
	2		3		
	Statistics	Probability	Statistics	Probability	
Groundnut prices	5.69	<0.001	6.91	<0.001	Nonlinear
	4.23	<0.001	4.81	<0.001	
	3.88	<0.001	4.79	<0.001	
	2.96	<0.001	4.19	<0.001	
Soybean prices	7.00	<0.001	9.52	<0.001	Nonlinear
	6.45	<0.001	7.59	<0.001	
	6.19	<0.001	6.84	<0.001	
	4.59	<0.001	5.33	<0.001	
Sesame prices	6.50	<0.001	6.67	<0.001	Nonlinear
	7.21	<0.001	6.76	<0.001	
	6.27	<0.001	5.54	<0.001	
	6.09	<0.001	5.41	<0.001	
Rapeseed and Mustard prices	6.42	<0.001	8.61	<0.001	Nonlinear
	6.03	<0.001	6.79	<0.001	
	6.05	<0.001	5.39	<0.001	
	6.26	<0.001	5.52	<0.001	

After considering nonlinearity of the best fitted ARIMA model Residuals, fitted with TDNN. Table 10, 11, 12 & 13 provides the performance of hybrid models for groundnut, soybean, sesame and rapeseed and mustard prices, respectively in terms of RMSE, MAE and MAPE. Among them the best hybrid model with least RMSE, MAE and MAPE values and reasonable number of parameters was selected.

Table 10 stated that out of 24 neural network models, a neural network model with two input nodes and four hidden nodes (*2:4s:1l*) was best for forecasting of groundnut prices. While in case of soybean and rapeseed and mustard, a neural network with three input nodes and three hidden nodes (*3:3s:1l*) performs better for forecasting price as revealed from Table 11 and 13, respectively. Table 12 showed that a neural network model with four input nodes and three hidden nodes (*4:3s:1l*) performs better among all models for sesame price forecasting.

Comparison of forecasting performance of best fitted ARIMA, TDNN and Hybrid model in terms of RMSE, MAE and MAPE presented in Table 14. Results revealed that hybrid model had least RMSE and MAE value then others except MAPE, so it stated that hybrid model have better forecasting efficiency.

Several researchers suggested that RMSE, MAE and MAPE type measures may not be appropriate for nonlinear models, hence as part of forecast accuracy we calculate the percentage of forecast that correctly predicted the monthly price change which presented in Table 15.

Table 10: Forecasting performance of hybrid models for groundnut prices

Model	No. of Parameters	RMSE		MAE		MAPE	
		Training	Testing	Training	Testing	Training	Testing
1:1s:1l	4	181.09	275.79	112.41	223.23	182.07	127.26
1:2s:1l	7	171.18	257.21	106.58	201.94	207.65	107.31
1:3s:1l	10	170.31	231.07	105.23	190.62	185.40	107.84
1:4s:1l	13	169.55	206.14	104.69	168.15	159.37	103.75
1:5s:1l	16	168.08	178.13	104.12	144.78	148.85	111.32
1:6s:1l	19	168.39	178.41	103.58	135.71	191.07	117.68
2:1s:1l	5	178.58	187.85	111.29	142.00	221.68	152.36
2:2s:1l	9	145.06	145.96	104.35	115.03	224.94	112.07
2:3s:1l	13	135.31	99.06	97.44	81.99	135.70	73.43
2:4s:1l	17	132.38	73.70	94.30	47.30	123.94	29.71
2:5s:1l	21	129.78	60.09	91.81	34.78	254.62	25.23
2:6s:1l	25	129.19	44.59	90.81	23.97	250.28	13.73
3:1s:1l	6	175.39	170.52	109.83	120.80	169.27	140.98
3:2s:1l	11	147.37	120.81	102.15	95.83	138.86	109.05
3:3s:1l	16	133.85	70.74	95.91	59.18	149.85	57.20
3:4s:1l	21	127.57	31.29	91.18	26.45	214.21	27.76
3:5s:1l	26	122.07	10.81	86.29	8.37	153.87	4.28
3:6s:1l	31	119.01	5.39	85.33	3.73	119.18	3.69
4:1s:1l	7	172.05	169.98	107.92	114.02	268.48	139.57
4:2s:1l	13	151.47	102.74	101.16	78.74	144.95	95.98
4:3s:1l	19	129.46	38.25	93.71	28.69	189.99	35.14
4:4s:1l	25	123.26	26.77	90.08	18.28	156.71	16.17
4:5s:1l	31	110.63	5.08	82.11	3.35	217.89	5.11
4:6s:1l	37	105.66	0.55	77.89	0.29	310.77	0.16

Table 11: Forecasting performance of hybrid models for soybean prices

Model	No. of Parameters	RMSE		MAE		MAPE	
		Training	Testing	Training	Testing	Training	Testing
1:1s:1l	4	163.91	559.96	115.01	414.86	137.01	98.16
1:2s:1l	7	163.16	468.78	114.08	321.48	129.92	85.40
1:3s:1l	10	162.99	403.79	113.95	305.46	128.74	82.45
1:4s:1l	13	162.28	333.11	113.31	249.05	127.75	82.49
1:5s:1l	16	161.75	321.71	112.96	238.99	125.66	80.96
1:6s:1l	19	159.32	327.26	111.96	227.31	125.75	75.21
2:1s:1l	5	159.58	465.68	112.57	362.89	176.12	117.10
2:2s:1l	9	154.35	345.79	110.30	263.21	170.02	105.65
2:3s:1l	13	147.11	192.50	106.62	147.99	167.71	94.65
2:4s:1l	17	141.77	134.37	104.42	103.99	150.97	75.54
2:5s:1l	21	137.84	82.58	101.87	61.53	151.11	52.03
2:6s:1l	25	132.54	40.87	99.12	30.53	145.48	23.99
3:1s:1l	6	152.77	459.85	111.38	358.63	153.21	100.76
3:2s:1l	11	145.81	312.84	106.44	243.49	142.51	84.25
3:3s:1l	**16**	**140.82**	**126.07**	**101.83**	**103.55**	**137.45**	**39.13**
3:4s:1l	21	132.38	52.35	97.50	41.73	137.15	18.37
3:5s:1l	26	127.39	31.86	94.92	25.38	128.27	13.32
3:6s:1l	31	124.57	3.46	91.55	2.29	135.16	1.75
4:1s:1l	7	150.66	475.25	109.75	352.45	157.86	80.54
4:2s:1l	13	139.77	260.14	102.88	194.34	157.01	59.10
4:3s:1l	19	134.44	103.46	98.13	80.62	145.95	33.25
4:4s:1l	25	123.84	62.21	90.73	40.89	127.03	15.48
4:5s:1l	31	116.61	17.73	85.69	9.77	120.99	2.74
4:6s:1l	37	112.96	3.33	83.60	1.73	125.01	1.11

Table 12: Forecasting performance of hybrid models for sesame prices

Model	No. of Parameters	RMSE		MAE		MAPE	
		Training	Testing	Training	Testing	Training	Testing
1:1s:1l	4	540.94	516.37	321.36	406.99	155.82	97.34
1:2s:1l	7	492.98	392.46	303.46	336.52	144.53	94.29
1:3s:1l	10	493.29	332.56	299.12	270.60	148.28	82.78
1:4s:1l	13	470.72	295.29	293.56	242.46	157.21	77.98
1:5s:1l	16	475.37	278.23	290.64	236.12	159.01	71.35
1:6s:1l	19	464.01	284.15	286.31	233.91	159.79	74.70
2:1s:1l	5	547.97	457.78	324.96	339.26	156.43	88.93
2:2s:1l	9	494.33	353.25	306.93	267.66	144.97	66.30
2:3s:1l	13	484.44	262.87	294.74	205.28	153.56	57.29
2:4s:1l	17	459.76	205.85	289.68	162.86	160.12	49.86
2:5s:1l	21	463.39	134.71	287.92	98.28	166.09	30.08
2:6s:1l	25	457.96	89.19	281.84	69.21	177.51	22.38
3:1s:1l	6	548.69	477.53	323.56	378.08	171.82	87.58
3:2s:1l	11	491.61	293.48	303.29	220.32	156.11	53.94
3:3s:1l	16	468.81	161.89	289.66	127.31	144.45	38.23
3:4s:1l	21	454.48	75.02	277.39	57.42	157.93	21.24
3:5s:1l	26	440.33	29.34	271.34	20.54	173.22	8.07
3:6s:1l	31	417.85	9.39	259.64	5.76	164.91	1.96
4:1s:1l	7	563.13	355.53	332.80	296.44	183.59	83.99
4:2s:1l	13	487.66	275.64	300.82	206.59	165.42	56.44
4:3s:1l	19	441.88	106.45	274.87	77.59	139.55	26.00
4:4s:1l	25	407.02	30.03	258.56	22.37	152.28	8.85
4:5s:1l	31	417.14	3.98	258.43	2.19	148.60	0.94
4:6s:1l	37	377.02	0.40	238.64	0.29	148.33	0.15

Table 13: Forecasting performance of hybrid models for rapeseed and mustard prices

Model	No. of Parameters	RMSE		MAE		MAPE	
		Training	Testing	Training	Testing	Training	Testing
1:1s:1l	4	129.55	189.75	75.41	150.74	121.41	122.20
1:2s:1l	7	126.39	183.14	73.72	149.57	131.17	116.70
1:3s:1l	10	124.36	140.50	72.47	109.37	130.71	88.43
1:4s:1l	13	111.93	136.46	70.04	104.56	125.22	86.32
1:5s:1l	16	107.07	125.97	69.12	96.50	122.16	79.98
1:6s:1l	19	110.92	128.48	68.23	100.53	131.08	83.68
2:1s:1l	5	126.58	178.20	73.49	142.15	123.83	131.60
2:2s:1l	9	123.09	107.72	71.49	84.98	145.58	71.63
2:3s:1l	13	112.28	93.48	68.83	71.76	162.86	63.30

2:4s:1l	17	100.72	72.24	65.96	58.98	162.29	54.42
2:5s:1l	21	93.92	48.55	64.65	39.74	156.32	38.45
2:6s:1l	25	88.42	39.95	62.13	30.50	159.35	31.44
3:1s:1l	6	124.37	149.31	71.53	122.56	135.21	114.29
3:2s:1l	11	115.58	101.12	68.64	84.63	150.61	89.65
3:3s:1l	16	110.51	61.47	67.20	47.73	166.40	38.70
3:4s:1l	21	89.41	38.17	61.55	25.70	158.52	25.57
3:5s:1l	26	89.01	25.01	60.03	17.56	160.05	21.32
3:6s:1l	31	82.04	6.99	57.75	4.42	154.82	6.06
4:1s:1l	7	124.29	150.24	71.94	118.08	132.12	105.11
4:2s:1l	13	108.01	94.21	68.41	74.87	131.70	71.83
4:3s:1l	19	109.97	43.35	64.66	33.55	140.83	33.17
4:4s:1l	25	93.81	13.58	59.35	9.92	140.86	7.82
4:5s:1l	31	80.87	4.05	55.86	2.67	127.94	2.46
4:6s:1l	37	76.38	1.29	53.60	0.79	142.10	0.98

Table 14: Comparison of ARIMA, TDNN and Hybrid model for different series

Series	Model	RMSE		MAE		MAPE	
		Training	Testing	Training	Testing	Training	Testing
Groundnut prices	ARIMA (1,0,1)	206.26		130.01		4.17	
	TDNN (3:3s:1l)	136.42	74.17	99.56	47.82	3.45	0.90
	Hybrid (2:4s:1l)	132.38	73.70	94.30	47.30	123.94	29.71
Soybean prices	ARIMA (3,1,1)	235.23		146.01		5.64	
	TDNN (2:3s:1l)	142.08	157.92	103.38	134.68	4.71	2.74
	Hybrid (3:3s:1l)	140.82	126.07	101.83	103.55	137.45	39.13
Sesame prices	ARIMA (1,1,1)	577.59		337.50		6.89	
	TDNN (3:3s:1l)	470.82	153.94	307.34	106.57	6.75	1.29
	Hybrid (4:3s:1l)	441.88	106.45	274.87	77.59	139.55	26.00
Rapeseed and Mustard prices	ARIMA (1,1,1)	145.09		86.34		3.38	
	TDNN (4:2s:1l)	119.58	66.05	72.46	50.07	3.22	0.95
	Hybrid (3:3s:1l)	110.51	61.47	67.20	47.73	166.40	38.70

Table 15: Percentage of forecast of correct sign for different series

Model	Percentage
Groundnut prices	
ARIMA (1,0,1)	37.50%
TDNN (3:3s:1l)	38.09%
Hybrid (2:4s:1l)	50.00%
Soybean prices	
ARIMA (3,1,1)	33.33%
TDNN (2:3s:1l)	50.00%
Hybrid (3:3s:1l)	52.38%
Sesame prices	
ARIMA (1,1,1)	50.00%
TDNN (3:3s:1l)	52.38%
Hybrid (4:3s:1l)	55.00%
Rapeseed and Mustard prices	
ARIMA (1,1,1)	33.33%
TDNN (4:2s:1l)	65.00%
Hybrid (3:3s:1l)	66.67%

4. Conclusion

This study has compared ARIMA, TDNN and Hybrid model in terms of modeling and forecasting using monthly wholesale price data of four oilseed crops namely groundnut, soybean, sesame and rapeseed and mustard. Based on the computational experience with four oilseed crops prices, results revealed that the hybrid model was superior to the ARIMA model and the ANN model. The lowest RMSE and MAE were achieved for the hybrid model than the ARIMA and ANN for all the four crops prices with the exception of MAPE which gave higher value and the percentage of forecasts of correct sign were achieved highest for the hybrid model than others.

The combinatorial approach was devised as an effective way to improve forecasting performance because of the complexity in linear and nonlinear structures. The empirical results with four oilseed crops price data sets clearly showed that the hybrid model outperformed each individual model.

5. References

1. Akaike H. On entropy maximization principle. In Applications of Statistics, Krishnaiah PR (ed.). North Holland: Amsterdam. 1977;8:27-41.
2. Anonymous, Food and Agriculture Organization of the United State (FAOSTAT), Rome, Italy; c2021a. Available at http://www.fao.org/faostat Retrieved on 20 November, 2021.
3. Anonymous, Centre for Monitoring Indian Economy Pvt. Ltd. (CMIE); c2021b. Available at https://commodities.cmie.com Retrieved on 26 November, 2021.
4. Box GEP, Jenkins GM. Time series analysis: Forecasting and control. San Francisc, Holden-Day; c1970.
5. Broock WA, Scheinkman JA, Dechert WD, LeBaron B.A test for independence based on the correlation dimension. Econometric Reviews. 1996;15(3):197-235.
6. Darbellay GA, Slama M. Forecasting the short-term demand for electricity: Do neural networks stand a better chance? International Journal of Forecasting. 2000;16:71-83.
7. Dickey DA, Fuller WA. Distribution of the Estimators for Autoregressive Time Series with a Unit Root. Journal of the American Statistical Association. 1979;74(366):427-431.
8. Elman J. Finding structure in time. Cognitive Science. 1990;14:179-211.
9. Haykin S. Neural networks: A comprehensive foundation (2 edition). Prentice Hall; c1999.
10. Jha GK, Sinha K. Time-delay neural networks for time series prediction: an application to the monthly wholesale

price of oilseeds in India. Neural Computing and Applications. 2012;24(3-4):563-571.
11. Kolmogorov AN. Stationary sequences in Hilbert space (in Russian). Moscow University Mathematics Bulletin. 1941;2(6):1-40.
12. Newbold P. ARIMA model building and the time series analysis approach to forecasting. Journal of Forecasting. 1983;2:23-35.
13. Schwarz G. Estimating the dimension of a model. Annals of Statistics. 1978;6:1310-1317.
14. Yule GU. On the method of investigating periodicities in disturbed series, with special reference to Wolfer's sunspot numbers. Philosophical Transactions of the Royal Society London, Series A. 1927;226:267-298.
15. Zhang GP. Time series forecasting using a hybrid ARIMA and neural network model. Neurocomputing: An International Journal. 2003;50:159-175.
16. Zhang G, Qi M. Neural network forecasting for seasonal and trend time series. European Journal of Operational Research. 2005;160:501-514.