Original Research Article

Time Series Analysis and Forecasting of Oilseeds Production in India – An Application of ARIMA and GMDH-Neural Network

ABSTRACT

Oilseeds have been the backbone of India's agricultural economy since long. Oilseed crops play the second most important role in Indian agricultural economy, next to food grains, in terms of area and production. Oilseeds production in India has increased with time, however, the increasing demand for edible oils necessitated the imports in large quantities, leading to a substantial drain of foreign exchange. The need for addressing this deficit motivated a systematic study of the oilseeds economy to formulate appropriate strategies to bridge the demand-supply gap. In this study, an effort is made to forecast oilseeds production by using Autoregressive Integrated Moving Average (ARIMA) model, which is the most widely used model for forecasting time series. One of the main drawbacks of this model is the presumption of linearity. The Group Method of Data Handling (GMDH) model has also been applied for forecasting the oilseeds production because it contains nonlinear patterns. Both ARIMA and GMDH are mathematical models well-known for time series forecasting. The results obtained by the GMDH are compared with the results of ARIMA model. The comparison of modeling results shows that the GMDH model perform better than the ARIMA model in terms of mean absolute error (MAE), mean absolute percentage error (MAPE), and root mean square error (RMSE). The experimental results of both models indicate that the GMDH model is a powerful tool to handle the time series data and it provides a promising technique in time series forecasting methods.

Keywords: Oilseeds, Forecasting, Autoregressive Integrated Moving Average, Group Method of Data Handling, Mean Absolute Percentage Error, Mean Absolute Error, Root Mean Square Error.

JEL Classification: ***Q10, C45, C53,***

1. INTRODUCTION:

India is one among world's largest producers and consumers of vegetable oils. Oilseeds have been the backbone of India's agricultural economy since long. Indian vegetable oil economy is the fourth largest in the world, next to USA, China, and Brazil. The country's contribution is 7 percent of the global vegetable oils production with 14 per cent share in the area. Oilseed crops play the second most important role in the

Indian agricultural economy next to food grains in terms of area and production. The Indian climate is suitable for the cultivation of oilseed crops; therefore, large varieties of oilseeds are cultivated here. The major oilseeds cultivated in our country are Groundnut, Rapeseed and Mustard, Castor seed, Sesame, Niger seed, Linseed, Safflower, Sunflower and Soybean. However, Groundnut, Rapeseed and Mustard, Sesame, Soybean and Sunflower account for a major chunk of the output. At present, more than 27 million hectares of land is under oilseeds cultivation. The area under oilseeds has been increasing over time and the production has registered many fold increase; however, the productivity is still low as compared to the other oilseed producing countries in the world. The low and fluctuating productivity is primarily because cultivation of oilseed crops is mostly done on marginal lands, which are lacking in irrigation and low levels of input is used here. To improve the situation of oilseeds in the country, Government of India has been pursuing several development programs, such as Oilseed Growers Cooperative Project, National Oilseed and Development Project, Technology Mission Oilseeds (TMO) and Integrated Scheme on Oilseeds, Pulses, Oil Palm and Maize (ISOPOM) etc. The concerted efforts of these development programs register significant improvement in annual growth of productivity and area under oilseed crops [2]. The combined efforts have been reflected in oilseeds production. But the growth in the domestic production of oilseeds has not been able to keep pace with the increase in demand in the country. As a result of which, India still imports a significant proportion of its requirement of edible oil. Edible oil is the largest imported (30 percent) commodity in India next only to petroleum products even though India had the world's second largest area under oilseeds [23].

In this paper, an effort has been made to forecast oilseeds production for the next five years (2016-17 to 2020-21). The model used for forecasting is an Autoregressive Integrated Moving Average (ARIMA) model. As the model was introduced by Box and Jenkins in 1960, this model is also known as Box-Jenkins model. The model is used for forecasting a single variable. Although it is used across various functional areas, its application is very limited in agriculture, mainly because of unavailability of required data and because agricultural output depends typically on monsoon and other factors [7]. The primary reason behind choosing ARIMA model for forecasting is that it assumes non-zero autocorrelation between the successive values of the time series data [12]. But ARIMA model can only capture linear feature of time series data [18] to deal with non-linearity of time series data, Group Method of Data Handling (GMDH) has also been used in our analysis for forecasting oilseeds production. This model was first used in 1968 by Prof. Alexey G. Ivakhnenko [11].

2. REVIEW OF LITERATURE:

Padhan Purna Chandra (2012) [17], has applied ARIMA model on a 60years' time series data (from 1950 to 2010) to forecast annual productivity of selected agricultural product (34 different products). The validity of the model is verified with various model selection criteria such as minimum of AIC (Akaike Information Criteria) and lowest MAPE (Mean Absolute Percentage Error) values. Among the selected crops, tea provides the lowest MAPE values, whereas cardamom provides lowest AIC values.

Kumar Manoj and Anand Madhu (2014) [12] forecasted sugarcane production in India by using ARIMA model. The order of the best ARIMA model was found to be (2, 1, 0). They suggested that the forecast results have shown the annual sugarcane production will grow in 2013, then there will be a sharp dip in 2014 and in subsequent years 2015 through 2017, it will continuously grow with an average growth rate of approximately 3 percent year-on-year.

Arivarasi R and Ganesan Madhavi (2015) [6] have also used the ARIMA Model to forecast the area and production of vegetables in the in the feeder zones (zone 1-Kancheepuram district & zone 2 -Thiruvallur district) of Chennai city. The ARIMA (0, 1, 2) model is suitable for the cultivation area of the zone 2 and ARIMA (2, 0, 1) model is suitable for zone 1. ARIMA (2, 0, 1) model is highly suitable for the vegetable production in both the zones. The model performances are validated by comparing the regression co-efficient values. While the model was used for forecasting for the period 2011-12 to 2014-15, decreasing trend was found in cultivated area and production of vegetables in zone 1.However, in zone 2 increasing trend was found in cultivated areas but decreasing trend was found for the vegetable production. Hence, it can be concluded that if this situation remained the same for a long period, then the further cultivation of vegetable crops will no longer be possible in both the zones.

Borkar Prema & Bodade V.M, (2017) [7] have applied the ARIMA model to forecast annual productivity of selected pulse crops. Applying annual data from 1950-51 to 2014-15, forecasted values have been obtained for another 5 years since 2016. The evaluation of forecasting of pulses production has been carried out with Root Mean Squares Percentage Error (RMSPE), Mean Absolute Percentage Error (MAPE) and Relative Mean Absolute Percentage Error (RMAPE).

Amanifardet. al., (2008a) [4] presented two meta-models based on the evolved group method of data handling (GMDH) type neural networks for modeling of both pressure drop (ΔP) and Nusselt number (Nu). It was shown that some interesting and important relationships like useful optimal design principles involved in the performance of micro-channels can be discovered by Pareto based multi-objective optimization of the obtained polynomial meta-models representing their heat transfer and flow characteristics. They concluded that, such important optimal principles would not have been obtained without the use of both GMDH type neural network modeling and the Pareto optimization approach.

Amanifardet. al., (2008b) [5] presented a quadratic model based upon some experimental results, using evolved GMDH-type neural networks for modeling of the transient evolution of spiky stall cells in an axial compressor. They concluded that the methodology applied in this work could sufficiently derive such complex model of unstable flow of rotating stall based on experimental input–output data. The prediction ability of such polynomial model has also been presented for some unforeseen data.

Ahmadiet. al., (2015) [3] proposed an intelligent approach to determine the output power and torque of a Stirling heat engine. The approach employs the GMDH method to develop an accurate predictive tool for determining output power and torque of a Stirling heat engine in manner that is inexpensive, fast and precise. Consequently, based on the output results, the GMDH approach can help energy experts to design Stirling heat engines with high levels of performance, reliability and robustness and with a low degree of uncertainty.

Osman Dag and Ceylan Yozgatligil (2016) [16] in their study, the R package GMDH is presented to make short term forecasting through GMDH-type neural network algorithms. The GMDH package has options to use different transfer functions (sigmoid, radial basis, polynomial, and tangent functions) simultaneously or separately. Data on cancer death rate in Pennsylvania from 1930 to 2000 are used to illustrate the features of the GMDH package. The results based on ARIMA models and exponential smoothing methods are included for comparison. GMDH algorithms show the same or even better performance than the other methods.

3. MATERIAL AND METHOD:

The specific objective of the study is to attempt a short-term forecasting of the future oilseeds production by using Autoregressive Integrated Moving Average (ARIMA) forecasting model and also through Group Method of Data Handling (GMDH) - neural network which is an important model of time series data (one the sub-model of Artificial Neuron Networks).

3.1 Data

The study used data of oilseeds production in India for the last 50 years, i.e., from 1966-67 to 2015-16 which have been collected from "Latest APY State Data", uploaded by the Ministry of Agriculture and Farmers Welfare, Govt. of India.

3.2 Autoregressive Integrated Moving Average (ARIMA)

The model used in this study is the autoregressive integrated moving average (ARIMA).The ARIMA is an extrapolation[i] method, which requires historical time series data of underlying variable.

The model in specific and general forms may be expressed as follows.

Let Y_t is a discrete time series variable which takes different values over a period of time. The corresponding AR (p) model of Y_t series,

Which is the generalizations of autoregressive model, can be expressed as:

AR (p) Y_t

$$Y_t = \mu + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \ldots + \phi_p Y_{t-p} + \varepsilon_t \;\ldots\ldots\ldots(1)$$

Where, Y_t is the response variables at time t,

Y_{t-1}, $Y_{t-2}, \ldots\ldots Y_{t-p}$ is the respective variables at different time with lags;

μ is the constant mean of the series, ϕ_1, $\phi_2, \ldots, \phi_p$ are the coefficients; and ε_t is the error factor. ε_t is a white noise process, where E(ε_t) = 0, var (ε_t) **= σ^2**>0, cov($\varepsilon_t, \varepsilon_{t-h}$) **= 0, t, h ≠ 0**

Similarly, the MA (q) model which is again the generalization of moving average model may be specified as:

MA (q): $Y_t = \mu + \varepsilon_t - \delta_1 \varepsilon_{t-1} - \delta_2 \varepsilon_{t-2} - \ldots - \delta_q \varepsilon_{t-q} \ldots\ldots\ldots(2)$

Where, μ is the constant mean of the series;

$\delta_1, \delta_2, \ldots \delta_q$ is the coefficients of the estimated error term; ε_t is the error term.

By combining both the models, we get the Autoregressive Moving Average or ARMA models, which has general form as:

$$Y_t = \mu + \phi_1 y_{t-1} + \phi_2 Y_{t-2} + \ldots + \phi_p Y_{t-p} + \varepsilon_t - \delta_1 \varepsilon_{t-1} - \delta_2 \varepsilon_{t-2} - \ldots - \delta_q \varepsilon_{t-q} \ldots\ldots(3)$$

Box and Jenkins argue that a non-stationary series can be transformed either into a stationary or an almost stationary series, if it is differenced an appropriate number of times. Thus, if we have a stochastic process {Y_t, t= 0, ±1, ±2, ... } which is non-stationary and has a trend, we can find a positive integer 'd' such that the transformed series Wt = ∇^dYt becomes stationary, ∇ being the difference operator, viz. $\nabla Y_t = Y_t - Y_{t-1}$, $\nabla^2 Y_t = Y_t - 2Y_{t-1} + Y_{t-2}$ and so on. After the transformed into a stationary or to an almost stationary series, the model transforms to ARIMA [9].

If Y_t is stationary at level or I(0) or at first difference I(1) or at second difference I(2) determines the order of integration. After the stationary of the series was attained, ACF (Auto Correlation Function) and PACF (Partial Auto Correlation Function) of the stationary series are employed to select the order p and q of the ARIMA model. The parameters were estimated using the non-linear least square method as suggested by Box and Jenkins (1976). ε_t is a white noise process, where E(ε_t) = 0, var (ε_t) **= σ^2** >0, cov **($\varepsilon_t, \varepsilon_{t-h}$) = 0, t, h ≠ 0**. Based on the model diagnostic tests and parsimony we obtained the best fitting ARIMA model.

The complete procedure of model building and forecasting are fully described by Box and Jenkins 1976. In short, they have suggested four basic steps viz., (i) Identification of the model, (ii) Estimation of parameters of the model, (iii) Diagnostic Checking of the model, and (iv) Forecasting. The details of the estimation and forecasting process are discussed below.

Identification: The first step of applying Box-Jenkins forecasting model is to identify the appropriate order of ARIMA (p, d, q) model. Identification of ARIMA model implies selection of order of AR(p), MA(q) and I(d). The order of d is estimated through I(1) or I(2) process of unit root stationary tests. The model specification and selection of order p and q involved plotting of autocorrelations functions (ACF) and partial autocorrelations functions (PACF) or correlogram of variables at different lag length. If the PACF displays a sharp cutoff while the ACF decays more slowly (i.e., has significant spikes at higher lags), we say that the series displays an AR signature. However, if the ACF displays a sharp cutoff while the PACF decay more slowly, we say that the series displays an MA signature [14]. The autocorrelation functions specify the order of moving average process, q and partial autocorrelations function select the order of autoregressive process p.

Estimation of the model: ARIMA models are fitted and accuracy of the model has tested based on diagnostics statistics. Once the order of p, d, and q are identified, their statistical significance can be judged by t-distribution. The next step is to specify appropriate regression model and estimate it. ARIMA models are fitted and accuracy of the model was tested based on diagnostics statistics.

Diagnostic checking: Now a question may arise that how we know whether the identified model is appropriate. One simple way to figure that out is by diagnostic checking the residual term obtained from ARIMA model by applying the same ACF and PACF functions. First obtaining the ACF and PACF of residual term up to certain lags of the estimated ARIMA model, and then checking whether the coefficients are statistically significant or not. The best model was selected based on the following diagnostics,

(i) Low Akaike Information Criteria (AIC): AIC is estimated by AIC = $-2\log_e(L) + 2m$, where $m = p + q$ and L is the likelihood function.

(ii) Low Bayesian Information Criteria (BIC): The Bayesian information criterion is a criterion for model selection among a finite set of models. It is based, in part, on the likelihood function, and it is closely related to Akaike information criterion (AIC). Sometimes, Bayesian Information Criteria (BIC) is also used and estimated by BIC = $-2\log_e(L) + \log_e(N)\, m$. Where N is number of observation and m is the number of parameters.

(iii) The minimum Root Mean Square Error (RMSE) and Mean Absolute Percent Error (MAPE) are used as a measure of accuracy of the models. RMSE=$\sqrt{\sum_{t=1}^{n}(X_{Actual,t} - X_{Forecast,t})^2/n}$ and MAPE $=\frac{1}{n}\sum_{t=1}^{n}[\frac{X_{Actual,t}-X_{Forecast,t}}{X_{Forecast,t}}]^2 x\ 100,$

Where, $X_{Actual,t}$ and $X_{Forecast,t}$ are actual and forecast output at time t,

(iv) These may also be judged by Ljung-Box Q (LBQ) statistic[ii] under null hypothesis that autocorrelation co-efficient up to lag k is equal to zero. LBQ is used to assess assumptions after fitting a time series model (ARIMA), to ensure that the residuals are independent.

Forecasting: Once the first three steps of ARIMA model are over, then we can obtain the forecasted values by estimating the appropriate model, which is free from problems. The forecasted values are reported for a maximum of 5 years, as long-term forecasting might not be appropriate.

The major drawback of ARIMA model is presumption of linearity, hence, no nonlinear patterns can be recognized by ARIMA model. Sometimes, the time series often contain nonlinear components; under such condition the ARIMA models are not adequate in modeling and forecasting [23]. To overcome this difficulty, GMDH (Group Method of Data Handling) model has been successfully used. To deal with *uncertainty, linearity or nonlinearity of time series data in a wide range of disciplines GMDH is more effective.*

3.3 Group Method of Data Handling (GMDH):

GMDH is a family of inductive algorithms for computer-based mathematical modeling of multi-parametric datasets that features fully automatic structural and parametric optimization of models [26]. GMDH is an original method for solving problems of structural and parametric identification under conditions of uncertainty [13]. It is an important model of time series data which is one sub-model of ANN[iii] (Artificial Neural Network). The main idea of the GMDH is to build an analytical function in a feed-forward network based on a quadratic node transfer function whose coefficients are obtained by using a regression technique. The GMDH is a self-organizing, unidirectional structure with multiple layers, each of which is composed of several neurons that have a similar structure. Weight is inserted inside each neuron as definite and constant values based on singular value decomposition method by solving normal equations [15].

The GMDH was introduced as a multivariate analysis method for modeling and identification of complex systems. In this model, the general connection between inputs and output variables can be expressed by a complicated polynomial series in the form of the Volterra series, known as the Kolmogorov-Gabor polynomial [11].

$$y_n = a_0 + \sum_{i=1}^{n} a_i x_i + \sum_{i=1}^{n}\sum_{j=1}^{n} a_{ij} x_i x_j + \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n} a_{ijk} x_i x_j x_k + \cdots , \qquad \ldots\ldots (4)$$

where{ x_1, x_2 , ... x_k , ……} is the vector of input variables and {a_0, a_i, a_{ij}, a_{ijk},…..} is the vector of coefficients of variables in the polynomial, *n* is the number of inputs, *Y* is a response variable, x_i and x_j are the lagged time series to be regressed. However, for most application the quadratic form are called as partial descriptions (PD) for only two variables is used in the form

$$y_n = G(x_i, x_j) = a_0 + a_1 x_i + a_2 x_j + a_3 x_i x_j + a_4 x_i^2 + a_5 x_j^2$$

to predict the output. The input variables are set to {x_i,x_j , x_k , ……,x_n} and output is set to {y}. The aim of the GMDH algorithm is to find a_i unknown coefficients of Volterra series. The coefficients (weights) a_i, for *i* = 0, 1, 2, 3, 4, 5 are determined using the least square method for each pair of x_i and x_j input variables [10].

The GMDH algorithm considers all pairwise combinations of p lagged time series. Therefore, each combination enters each neuron. Using these two inputs, a model is constructed to estimate the desired output. In other words, two input variables go in a neuron, one result goes out as an output. The structure of the model is specified by the Ivakhnenko polynomial in equation 4 where n = 2. This specification requires six coefficients in each model to be estimated [16].

The main function of GMDH is based on the forward propagation of signal through nodes of the net similar to the principle used in classical neural nets. Every layer consists of simple nodes, each of which performs its own polynomial transfer function and passes its output to nodes in the next layer. The computation process comprises three basic steps [8]:

Step 1: Select input variables $\{x_1, x_2, x_k, \ldots\ldots, x_n\}$ where n is the total number of input. The data are separated into training and testing data sets. The training data set is used to construct a GMDH model and the testing data set is used to evaluate the estimated GMDH model.

Step 2: Construct L numbers of new variables Z $=\{z_1, z_2, z_3, \ldots\ldots, z_L\}$ in the training data set for all independent variables and choose a PD of the GMDH. Conventional GMDH has been developed using polynomial, PD of the following form

$$z_l = G(x_i, x_j) = a_0 + a_1 x_i + a_2 x_j + a_3 x_i x_j + a_4 x_i^2 + a_5 x_j^2 \quad for\ l = 1,2,3..,, L.$$

where, $L = n(n-1)/2$

Select new variables as input of the next middle layer and truncate the subsequent computation. With the identification of the optimal output of partial polynomials at each layer, the selection of new variables enables the network to quickly converge to the target solution. Once the partial polynomial equations at each unit are selected, the residual error in each layer is further checked to determine whether the set of equations of the model should be further improved within the subsequent computation.

Step 3: Estimate the coefficient of the PD. The vectors of coefficients of the PDs are determined using the least square method.

Step 4: Determine new input variables for the next layer. There are several specific selection criteria to identify the input variables for the next layer. In our study, we used two criteria. The first criteria, the single best neuron out of these L neurons, Z′ identified according to the value of mean square error (MSE) of testing dataset. In second criteria, eliminate the least effective variables, replace the column of $\{x_1, x_2, x_k, \ldots\ldots, x_n\}$ by those column $\{z_1, z_2, z_3, \ldots\ldots, z_l\}$ that best estimate the dependent variable y in the testing dataset.

Step 5: Build the final model and compute the predicted value. The final prediction model can be obtained with selected variables in each layer and the coefficients of partial polynomials between the connected layers. Check the stopping criterion. The lowest value of selection criteria using GMDH model at each layer obtained during this iteration is compared with the smallest value obtained at the previous one.

The structure of the GMDH algorithm is illustrated in Figure 1. Those shadowed nodes in Fig 1 that have significant contribution to the output and are selected to be input in the next layer [25].

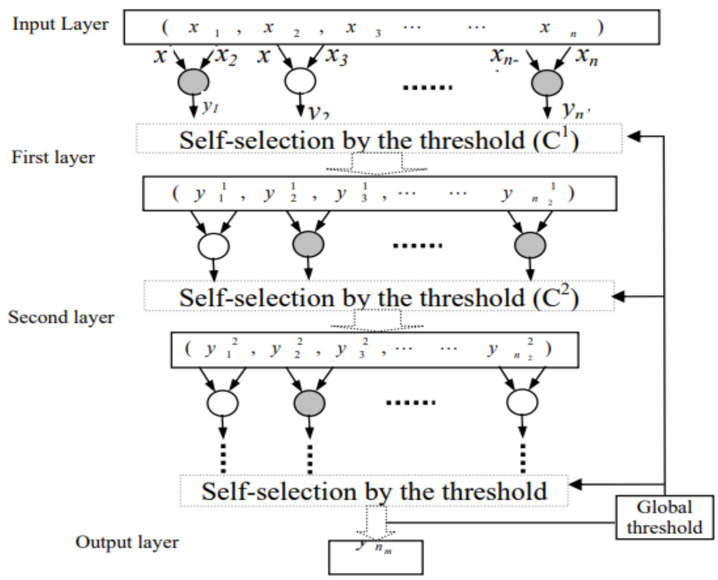


Figure (Fig.) 1: Structure of the GMDH algorithm.

The GMDH algorithm is a system of layers in which there exist neurons. The number of neurons in a layer is defined by the number of input variables. To illustrate, assume that the number of input variables is equal to p; since we include all pair-wise combinations of input variables, the number of neurons is equal to $h = {}^{p}c_{2}$ [16].

3.3.1 Time series prediction by GMDH

A classical method for time series forecasting problem, the number of input nodes of nonlinear model, such as the GMDH is equal to the number of lagged variables ($y_{t-1}, y_{t-2}, y_{t-3} \ldots, y_{t-p}$), where p is the number of chosen lagged. The outputs, y_t, the predicted value of a time series defined as

$$y_t = f\,(y_{t-1}, y_{t-2}, y_{t-3} \ldots, y_{t-p}),$$

However, there is no suggested systematic way to determine the optimum number of lagged p. The number of lagged p is chosen either in an adhoc basis or from traditional Box Jenkins methods. The lagged variables obtained from the Box-Jenkins analysis are the most important variables to be used as input nodes in the input layer of the GMDH model [19]. In our study, a time series model is considered as nonlinear function of several past observations and random errors as follows:

$$y_t = f[\,(y_{t-1}, y_{t-2}, y_{t-3} \ldots, y_{t-p}), (a_{t-1}, a_{t-2}, a_{t-3} \ldots, a_{t-q})]$$

where f is a nonlinear function determined by the GMDH.

3.3.2 Data structure of GMDH

An illustration of time series data structure in GMDH algorithms is presented in Table 1. Since we have a time series data set with t time points and p inputs. We construct the model for the data with time lags, the number of observations presented under the subject column in the table is equal to *t-p*; and the number of inputs i.e, lagged time series, is *p*. In this table, the variable called y is put in the models as a response variable, and the rest of the variables are taken into models as lagged time series x_i, where i = 1,2,...,p. The notations in Table 1 are followed throughout this paper.

Table 1: An illustration of time series data structure in GMDH algorithms

Subjects	*Y*	x_1	x_2	x_3		x_p
1	y_t	y_{t-1}	y_{t-2}	y_{t-3}		y_{t-p}
2	y_{t-1}	y_{t-2}	y_{t-3}	y_{t-4}		y_{t-p-1}
3	y_{t-2}	y_{t-3}	y_{t-4}	y_{t-5}		y_{t-p-2}
….						
t-p	y_{p+1}	y_p	y_{p-1}	y_{p-2}		y_1

A better model which explains the relation between response and lagged time series is captured via transfer functions.

4. RESULTS AND DISCUSSION:

4.1 ARIMA Model:

The preliminary understating about the nature of data showed that there is no consistency in the production of oilseeds over the time period (Fig. 2). The variable shows increasing trend.

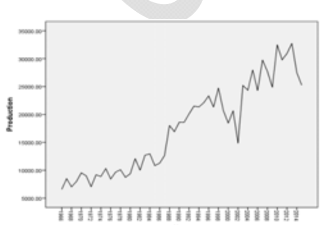


Fig 2. Time series plots of oilseeds

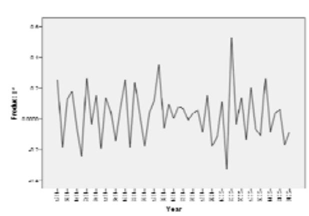


Fig 3. Plots of 1st difference

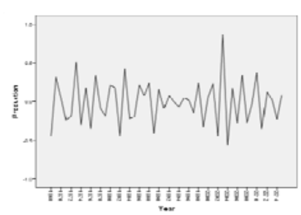


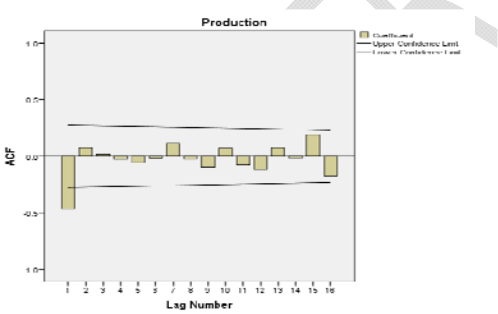
Fig 4. Plots of 2nd difference

Identification:

Identification of the model was concerned with deciding the appropriate values of (p, d, q). Auto regressive and moving average terms are identified based on ACF and PACF values. The ACF helps in choosing the appropriate values for ordering of moving average terms (MA) and PACF for those autoregressive terms (AR).

ARIMA model is generally applied for stationary time series data. Stationary vs. non-stationary can check through correlogram or autocorrelation functions. If autocorrelation coefficients don't die out slowly, then the series is probably non-stationary. The general procedure to convert a non-stationary series to a stationary series is through first difference or second difference. In general, most of the variables are I (1) i.e., first difference or I (2) i.e., second difference, thereby ARMA model is applied at I(1) or maybe I(2). Both the first differences and the second difference time series data of production are given in Fig 3 and Fig 4, respectively. Comparing the figures, it has been observed that in the first figure, difference magnitude of auto correlation is lower than that in the second difference data. Hence, we considered I(1) for making the series stationary.

ACF and PACF of production of oilseeds are presented in Figs 5 and 6. Based on these figures, the initial ARIMA model has been developed. It can be seen from Figs 5 and 6 that there is a slow decay in the PACF, but it also has a cut-off only at lag1, suggesting AR (1). The ACF also has one significant spikes at lag1. This pattern is typical to an MA process of orders 1.



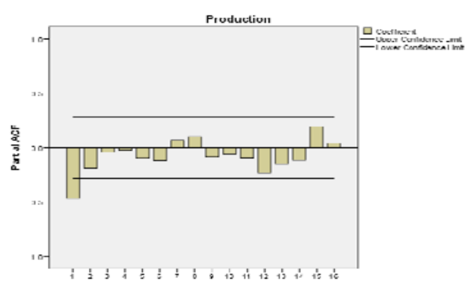


Fig. 5: ACF of 1stdifferenced series by lag Fig.6: PACF of1stdifferenced series by lag

Estimation of the model: Once the orders of p, d, and q are identified, the next step is to specify appropriate ARIMA model and estimate it. With the help of SPSS software, various orders of ARIMA model has been estimated. After the identification process has completed, the number of possible models are identified. According to identification process, the model has been identified as ARIMA (1, 1, 1). However, the coefficient of AR (1) is not statistically significant. Hence in addition to ARIMA (1, 1, 1), the study also attempts to estimate ARIMA (1, 1, 0) and ARIMA (0, 1, 1) model. The results of ARIMA (1, 1, and 1), ARIMA (1, 1, 0) and ARIMA (0, 1, and 1) are summarized in Table 2.

Table2: Coefficients of estimated values of fitted ARIMA models

Sl.No	Variable	Model	Constant	AR(1)	MA(1)
1	Production	ARIMA(0,1,1)	0.028	-	0.596
		SE	0.009		0.126
		t- value	3.216		4.718
2	Production	ARIMA(1,1,0)	0.027	-0.479	-
		SE	0.015	0.128	-
		t- value	1.804	-3.743	-
3	Production	ARIMA(1,1,1)	0.028	-0.093	0.519
		SE	0.010	0.261	0.234
		t- value	2.901	-0.357	2.214

We proceeded to further statistically analyze these two possible models. The best model is selected based on the diagnostics checking.

Diagnostic checking: Now a question may arise that how we know whether the identified model is appropriate. After an estimation of the parameters, we test the adequacy of the model based on Box-Pierce (Q) and Ljung-Box (LB) statistics. The statistics is calculated from the ACF of residual term up to 16 lags of the estimated ARIMA model. We also check the statistical significance of the parameters. An adequate model does not always generate good forecasts. Further, we select the model having low Bayesian Information Criteria (BIC), lowest root means square error (RMSE), lowest mean absolute percent error (MAPE), and highest stationary R-Square and R-Square.

Comparing these three models, the ARIMA (0,1,1) model is found to be the best for oilseeds production. Only in this model, the estimated coefficient is statistically significant. LB and Q statistics of the model is also statistically significant. At the same time, RMSE, MAPE, MAE and BIC of ARIMA (0,1,1) have shown a value lower than that of ARIMA(1,1,0) and ARIMA(1,1,1) models. The summary of the estimates of ARIMA (0,1,1) models is given in Table 3.

Table3: RMSE, MAPE, BIC values and Q statistics of fitted ARIMA models

Sl. No	Variable	Model	RMSE	MAPE	MAE	BIC	Stationary R^2	R^2	Ljung Box Q Statistics	Df
1	Production	ARIMA (0,1,1)	2811.85	11.72	2008.66	16.04	0.26	0.87	13.03	17

Based on the parameter estimates in the Table 2 and model statistics presented in the table 3, the study chose the ARIMA (0,1,1) as the best model for the oilseeds production in the India. The model is thus given as:

$$\nabla Y_t = (1 - 0.596B)\varepsilon_t$$

This model is a special case of ARIMA model, which is called an Integrated Moving Average Model.

Forecasting: Once the identification, estimation of the model and diagnostic checking steps of ARIMA model is over, then we can obtain forecasted values by estimating the appropriate model, which is free from problems. The forecasted values obtained from ARIMA model are reported in Table 4. The forecasted values are reported for a maximum 5 years as long-term forecasting might not be appropriate.

Table 4: Forecast values with ARIMA model

Model	Variable	Value	Years				
ARIMA (0,1,1)	Production (000 tonnes)		2016-17	2017-18	2018-19	2019-20	2020-21
		Forecast	30062	30987	31939	32922	33934
		Lower	22069	22181	22330	22510	22715
		Upper	40062	42195	44372	46601	48887

In our study, ARIMA (0,1,1) is the best model for oilseeds production. Based on this model, forecasted values of oilseeds production will be 30062 thousand tonnes, 30987 thousand tonnes, 31939 thousand tonnes, 32922 thousand tonnes and 33934 thousand tonnes during 2016-17, 2017-18, 2018-19, 2019-20 and 2020-21, respectively. It is clear that oilseeds production will be slightly increasing over time.

4.2 GMDH Model

In this section we analyze the short-term forecasting results of oilseeds production through GMDH-type neural network algorithms[iv] by using GMDH Shell software. GMDH-neural network selects the model of optimal complexity and such a selection depends on the form of external criterion realization. *K*-fold cross validation is one of such criteria. In our study, we used this k fold validation method. In this validation, original sample was randomly partitioned into *k* subsamples. A single subsample was taken as the validation data for testing model, and the other *k* – *1* sub-samples were used as training data. The cross-validation process was repeated *k* times using each of the *k* subsamples exactly once. The value of *k* obtained from the *K* folds can produce a single estimation. The advantage of this method over repeated random sub-sampling is that all observations are used for both training and validation, and each observation is used for validation exactly once. The experiment was carried out using RMSE validation criterion [13]. Therefore, the optimal time series forecasting model was selected by minimum value of RMSE, calculated for the testing sample. This validation criterion defines model selection criterion for both the core algorithm[v] and variables ranking[vi] (Solver, GMDH shell documents). In our time series analysis under GMDH-neural network model, based on k- cross validation criterion, our forecasting model is an optimal with k=2.

In this model the variables ranking are selected by error. Variables are dropped after rank 600. The neural–type method used as a core of algorithm in our model. The summary of the results of our model depict that model complexity (it informs about the number of coefficients in the model and the number of layers) is 2 of 6. It means that the model has two layers and six coefficients or weight of polynomial. Maximum number of layer selected in our model are 33 with initial layer[vii] width 1. The Criterion value of this model is 0.060354 which informs about the value of validation criterion configured in the Solver module[viii]. Top-ranked model has the smallest criterion value. Our model's low criterion value indicates that the model is suitable for this data.

The formula of suggested forecasting model under GMDH –neural network is given by

$$Y_t = 6677.04 + 1.036\, Y_{t-15} + 0.005\, Y_{t-23}$$

Accuracy of model shows different accuracy metrics for the model selected in the model browser. Model contains accuracy measures calculated for observations used to create the model. Error measure is used to choose a metric for calculation of the mean and the root mean errors. Available metrics are the absolute (MAE and RMSE), which outputs mean error values "as is" and the target percentage (MAPE), where for each model value we calculate percentage deviation from the actual value and then the percentage deviations are averaged [20]. The model statistics of GMDH - neural network are presented in Table 5.

Table 5: RMSE, MAPE, MAE values of fitted GMDH neural network models

Sl. No	Variable	Model	RMSE	MAPE	MAE	R^2
1	Production	GMDH	1833.72	5.275	1473.56	0.99

Calculation of magnitude of predicted variable involves only the observations that are used for training and testing. The forecasting values are presented in Table 6. In our study, GMDH neural networks model forecasting oilseeds production will be 28176 thousand tonnes, 22145 thousand tonnes, 32864 thousand tonnes, 32008 thousand tonnes and 35751 thousand tonnes in 2016-17, 2017-18, 2018-19, 2019-20, 2020-21, respectively.

Model	Variable	Value	Years				
GMDH	Production (000 tonnes)		2016-17	2017-18	2018-19	2019-20	2020-21
		Forecast	28176	22145	32864	32008	35751
		Lower	24508	18477	29196	28340	32083
		Upper	31844	25813	36532	35676	39419

Table 6: Forecast values with GMDH neural network model

5. COMPARISON BETWEEN ARIMA and GMDH-Neural Network Model:

Now the question that arises is which model is better and appropriate for forecasting the oilseeds production. To find the solution, we compare the model statistics of ARIMA and GMDH-neural network in terms of RMSE, MAE and MAPE. Model with lower values of RMSE, MAE and RMPE as compare to the other model, is better. The model statistics of GMDH-neural network and ARIMA both are presented in Table 7. The table indicates that GMDH-neural network is better model than ARIMA in all respect.

Variable	Model	RMSE	MAE	MAPE	R^2
Production	ARIMA (0,1,1)	2811.85	2008.66	11.71	0.88
	GMDH	1833.72	1473.89	5.275	0.99

Table 7: RMSE, MAPE, MAE statistics of fitted ARIMA models and GMDH

To verify our results, we considered similar research works such as Srinivasan, 2008, [22] and Xu et.al. 2012, [24]. Srinivasan (2008) used a GMDH-type neural network and traditional time series models to forecast predicted energy demand. It was shown that a GMDH-type neural network was superior in forecasting energy demand compared to traditional time series models with respect to MAPE. In another study, Xu et.al. (2012) applied a GMDH algorithm and ARIMA models to forecast the daily power load. According to their results, GMDH-based results were superior to the results of ARIMA models in terms of MAPE for forecasting performance.

Since the above analysis lends support to the choice of GMDH-neural network over ARIMA type modeling we would propose the values obtained from GMDH-neural network as the forecast outcome.

6. FINAL FORECASTING:

The outcome of GMDH model are presented precisely in Table8,

Table 8: Forecast values with GMDH- neural network model

Variable	Model	Predicted				
Production (000 tonnes)	GMDH	2016-17	2017-18	2018-19	2019-20	2020-21
		28176	22145	32864	32008	35751

The graphical presentation of forecasted value of oilseeds production under GMDH- neural network is depicted in Figure 7. In the diagram below, time is measured along the horizontal axis and the vertical axis measures level of production. Actual value is presented by black line and fitted value is shown in blue. The red line indicates the forecast value of oilseeds production whereas confidence band has been represented by shaded area.

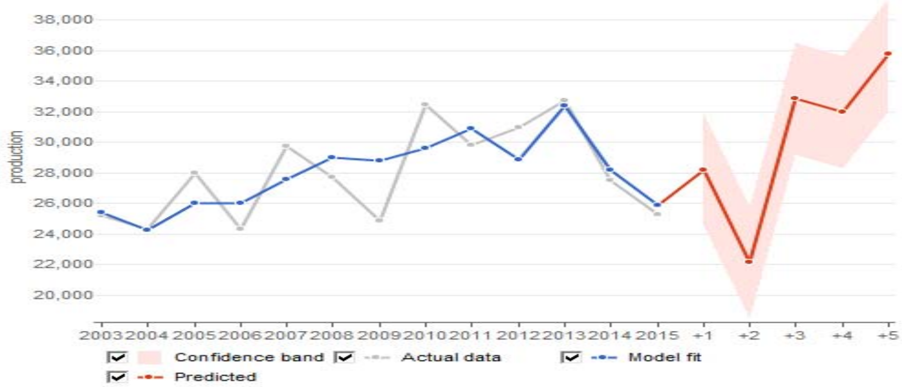


Fig.7:Actual value, fitted value and forecast value and confidence band in GMDH model

From both from Table 8 and Fig 7, it is clear that the expected oilseeds production will increase in India in near future which will reduce the gap between demand and supply of oilseeds. Alternatively, it can be said that this rise in supply will be helpful in meet in the growing domestic demand for edible oil due to increase in population. As a result, the dependence on imported edible oil will reduce substantially, preventing the huge expenditure of already scarce foreign exchange.

7. CONCLUSION

ARIMA models are not always adequate for the time series that contains non-linear structures. In this context, a nonlinear GMDH can be an effective way to improve forecasting performance. Based on the results obtained in our study, one can infer that application of GMDH techniques in modeling and forecasting of time series can increase the forecasting accuracy. More specifically, the GMDH-neural network model performed better for forecasting oilseed production of India as compared to ARIMA models. The results of forecasting in GMDH-neural network methods reveals that India's oilseeds production will be 28176 thousand tonnes in 2016-17. It will decline to 22145 thousand tonnes in 2017-18 and thereafter it will increase to 32864 thousand tonnes in 2018-19, 32008 thousand tonnes in 2019-20 and 35751 thousand tonnes in 2020-21.This production of oilseeds may not be adequate to make our country self-sufficient. This is because the demand for oilseeds grows faster along with rising population. Still the gap between demand and supply of oilseeds will reduce, resulting in reduced dependence on imported of edible oil and drain of foreign exchange from India will be under control.

8. REFERENCES:

1. Adhikari Ratnadip, Agarwal RK (2013) *An Introductory Study on Time Series Modeling and Forecasting*, Lambert Academic Publishing, https://arxiv.org/abs/1302.6613.
2. Agropedia (2011) Oilseeds Scenario of India, submitted by YadavKiran http://agropedia.iitk.ac.in/content/oilseeds-scenario-india
3. Ahmadi MH, Ahmadi MA, Mehrpooya M, Rosen M A (2015) Using GMDH Neural Networks to Model the Power and Torque of a Stirling Engine. *Sustainability, 7*: 2243-2255.
4. Amanifard N, Nariman-Zadeh N, Borji M, Khalkhali A, Habibdoust A (2008a) Modelling and Pareto optimization of heat transfer and flow coefficient in microchannels using GMDH type neural networks. Energy Conversion and Management. 49 (2):311-325.
5. Amanifard N, Nariman-Zadeh, N, Farahani, M H, Khalkhali, A (2008b) Modelling of multiple short-length-scale stall cells in an axial compressor using evolved GMDH neural networks, Energy Conversion and Management. 49(10): 2588–2594.
6. Arivarasi R and Madhavi Ganesan (2015) Time Series Analysis of Vegetable Production and Production and Forecating using ARIMA Model, *Asian Journal of Science and Technology*, 6(10):1844-1848,http://www.journalajst.com/sites/default/files/2456.pdf
7. Borkar Prema & Bodade VM (2017) Application of ARIMA Model for Forecasting Pulses Productivity in India, *Journal of Agricultural Engineering and Food Technology*, 4(1).
8. Chang FJ and Hwang YY (1999) A self-organization algorithm for real-time flood forecast, Hydrological processes, 13: 123-138.
9. Datta LK (2009) Study on Analysis of Financial Data Using Multivariate and Time Series Technique, Ph.D thesis, Department of Statistics, Saurashtra University, Rajkot, Gujarat.
10. Iba H, De Garish. H & Sato T(1995) A numerical approach to genetic programming for system identification, Evolutionary Computation, 3 (4): 417-452.
11. Ivakhnenko AG (1971) Polynomial theory of complex system, IEEE Trans. Syst., Man Cybern. SMCI-1, 1: 364-378.
12. Kumar Manoj & Anand Madhu (2014) An Application of Time Series ARIMA Forecasting Model for Predicting Sugarcane Production In India, *Faculty of Economic Sciences*, 9 (1): 81-94.
13. Latysh Elena, Koshulko Oleksiy (2012)Testing k-value in k-fold Cross Validationof Forecasting Models for Time Series Analysisof G-spreads of Top-quality RUB Bonds,*The 5th International Workshop on Inductive Modelling IWIM'2012'.*
14. Nasiru MO and Olanrewaju SO.(2015) Forecasting Airline Fatalities in the World Using a Univariate Time Series Model, *International Journal of Statistics and Applications*, 5(5): 223-230, http://article.sapub.org/10.5923.j.statistics.20150505.06.html

15. Nariman-Zadeh N, Darvizeh A, Ahmad-Zadeh R (2003) Hybrid Genetic Design of GMDH-type Neural Networks Using Singular Value Decomposition for Modelling and Prediction of the Explosive Cutting Process, *Journal of Engineering Manufacture*, Proceedings of the IMechE Part` B, 217: 779 – 790.

16. Osman Dag and CeylanYozgatligil (2016) GMDH approach can help energy experts to design Stirling heat engines with high levels of performance, reliability and robustness and with a low degree of uncertainty. *The R Journal,* 8(1): August, 2016.

17. Padhan Purna Chandra (2012) Application of ARIMA Model for Forecasting Agricultural Productivity in India India. *Journal of Agriculture and Social Science*, 8(2): 50–56.

18. Samsudin R, Saad P, Shabri A(2011) A hybrid GMDH and least squares support vector machine in time series forecasting, Neural Network World, 3(11): 251-268

19. Shabri A and Samsudin R (2014) A Hybrid GMDH and Box-Jenkins Models in Time Series Forecasting,*Applied Mathematical Sciences*, 8(62): 3051 -3062.

20. Simulation results, GMDH shelldocuments, http://d.gmdhshell.com/bf3/doku.php?id=processing_results

21. Solver, GMDH shell documents.http://d.gmdhshell.com/docs/solver

22. Srinivasan D (2008) Energy demand prediction using GMDH networks. Neurocomputing, 72(1): 625–629,

23. Rathod Santosha, Singh KN, Patil SG, Naik Ravindrakumar H, Ray Mrinmoy, and MeenaVikram Singh (2018) Modeling and forecasting of oilseed production of India through artificial intelligence techniques, *Indian Journal of Agricultural Sciences,* 88(1): 22-27.

24. Xu H, Dong Y, Wu J, and Zhao W (2012) Application of GMDH to short-term load forecasting. *In Advances in Intelligent Systems*, pages. 27–32. Springer-Verlag.

25. Wang X, Li L, Lockington D, Pullar D and Jeng DS(2005) Organizing Polynomial Neural Network for Modelling Complex Hydrological Processes, Research Report, R861, Department of Civil Engineering, The University of Sydney.

26. WikiVisually - Group method of data handling(2018) https://wikivisually.com/wiki/Group_method_of_data_handling

List of Abbreviations

AR:	Autoregressive
MA:	Moving Average
ARMA:	Autoregressive Moving Average
ARIMA:	Autoregressive Integrated Moving Average
ACF:	Autocorrelations Functions
PACF:	Partial Autocorrelations Functions
GMDH:	Group Method Data Handling
ANN:	Artificial Neural Network
RMSE:	Root Mean Square Error
MAE:	Mean Absolute Error
MAPE:	Mean Absolute Percentage Error
AIC:	Akaike Information Criteria
BIC:	Bayesian Information Criteria
Q Statistics:	Box-Pierce
LB:	Ljung-Box
TMO:	Technology Mission Oilseeds
ISOPOM:	Integrated Scheme on Oilseeds, Pulses, Oil Palm and Maize
PD :	Partial Descriptions

End note

[i]**Extrapolation** techniques make forecasts using only the past data.

[ii]The Ljung-Box Q statistic to test whether a series of observations over time are random and independent. If observations are not independent, one observation can be correlated with a different observation k time units later, a relationship called autocorrelation. Autocorrelation can decrease the accuracy of a time-based predictive model, such as time series plot, and lead to misinterpretation of the data.

[iii]**ANN**: The basic objective of ANNs was to construct a model for mimicking the intelligence of human brain into machine. Similar to the work of a human brain, ANNs try to recognize regularities and patterns in the input data, learn from experience and then provide generalized results based on their known previous knowledge. Although the development of ANNs was mainly biologically motivated, but afterwards they have been applied in many different areas, especially for forecasting and classification purposes [1].

[iv]**GMDH-type neural network** algorithms are modeling techniques which learn the relations among the variables. In the perspective of time series, the algorithm learns the relationship among the lags. After learning the relations, it automatically selects the way to follow in algorithm.

[v]**Core algorithms** perform generation and selection of model structures. Then model coefficients are fitted using the least squares method.

[vi]**Variables ranking** turns on preliminary ranking and reduction of variables. Ranking of variables according to their individual ability to predict testing data.

[vii]**Initial layer** width means how many neurons are added to the set of inputs at each new layer.

[viii]Solver [21] module produces predictive models for target variables.