https://doi.org/10.56093/JISAS.V78I1.7
Available online at http://isas.org.in/isa/jisas
JOURNAL OF THE INDIAN SOCIETY OF AGRICULTURAL STATISTICS 78(1) 2024 53–62

Comparative Study of EMD based Modelling Techniques for Improved Agricultural Price Forecasting

Bikramjeet Ghose[1], Pramit Pandit[1], Chiranjit Mazumder[2], Kanchan Sinha[3] and Pradip Kumar Sahu[1]

[1]Bidhan Chandra Krishi Viswavidyalaya, Mohanpur, Nadia
[2]ICAR-Indian Agricultural Research Institute, New Delhi
[3]ICAR-Indian Agricultural Statistics Research Institute, New Delhi

Received 29 March 2023; Revised 05 January 2024; Accepted 16 April 2024

SUMMARY

Forecasting agricultural commodity prices is regarded as a challenging task due to its non-linear and non-stationary nature. As agriculture production is highly reliant on various biological and agro-meteorological factors, traditional smoothing techniques as well as statistical models often fail to model such series satisfactorily. To capture such complex patterns effectively, different data-driven and self-adaptive techniques have been developed time-to-time. Against this backdrop, in this paper, we have assessed the suitability of empirical mode decomposition (EMD)-based neural network and support vector regression (SVR) approaches for forecasting wholesale prices of three major potato markets namely, Agra, Bangalore, and Mumbai. As the benchmark models, autoregressive integrated moving average (ARIMA), time delay neural network (TDNN) and SVR models have been employed for the comparative evaluation. The experimental results clearly reveal the comparative superiority of the EMD-SVR model for the Agra and Bangalore markets and the EMD-TDNN model for the Mumbai market in terms of root mean squared error values and turning point predictions. Moreover, all the EMD-based models have performed better than the other competing models.

Keywords: ARIMA; Empirical mode decomposition; SV; Price forecasting; Potato; TDNN.

1. INTRODUCTION

One of the most difficult tasks in time series analysis is price forecasting of agricultural commodities. Accurate price forecasting has emerged as a key concern in agriculture from the perspectives of farmers, policy planners and agro-based industries. Agricultural prices often exhibit random patterns due to their high dependence on weather and other associated factors. As a result, these lead agricultural commodity prices to be non-linear and non-stationary in nature. In literature, many artificial intelligence (AI) models such as artificial neural network (ANN), support vector regression (SVR), etc. have been found to be more useful as compared to the traditional autoregressive integrated moving average (ARIMA) model. The AI models can map any non-linear function without making any assumptions about the characteristics of the data. In addition, these do not require any prior model assumptions. In spite of the recent success of the ANN models in different domains, the problem of the vanishing as well as the exploding gradient and over fitting due to the back propagation algorithm sometimes limit its generalisation capability. As a result, SVR based on the structured risk minimisation principle often provides improved generalising ability over ANN.

Several authors have made effort to compare as well as to improve the performance of different machine learning algorithms. Kajitani *et al.* (2005) compared the performance of artificial neural network models and autoregressive time series models for the Canadian lynx data set. Their results showed that ANN models performed better than Autoregressive (AR) models in the presence of non-linear and non-Gaussian

Corresponding author: Kanchan Sinha
E-mail address: kanchan.sinha@icar.gov.in

characteristics. The importance of detrending and deseasonalisation in neural network forecasting was discussed by Zhang and Qi (2005). They observed that detrending combined with deseasonalisation increases forecasting precision. Duan and Stanley (2011) used the SVR to reduce the impact of cross-correlations between various financial markets to enhance the prediction of financial return data series. They emphasised how the structural risk minimisation approach has improved the forecasting accuracy.

As the conventional mono-scale smoothing techniques often fail to capture the complex non-linear and non-stationary patterns, Huang *et al.* (1998)) proposed the concept of empirical mode decomposition (EMD). It utilises the principle of the divide and conquers rule i.e., it decomposes the non-linear and non-stationary data into several intrinsic mode functions (IMF) and a residue with varying frequencies. Then each IMF along with the residue is modelled and forecasted. These forecasted values are summed up to obtain the final series.

An *et al.* (2012) have mentioned that EMD can detect hidden patterns and trends of time series. Guo *et al.* (2012) have demonstrated the superiority of an EMD-based feed-forward neural network in comparison to the traditional forecasting techniques in anticipating wind speed series. Chen *et al.* (2012) also proposed an EMD-based neural network model to predict tourism demand. Their proposed model outperformed the ARIMA model as well as the ANN model. Cheng and Wei (2014) utilised the EMD-SVR approach to forecast stock prices for the Taiwan stock exchange. They compared it with AR and SVR models and observed the performance of the EMD-SVR model to be superior. Das *et al.* (2019) have compared the EMD-ANN with the EMD-SVR model and found that the EMD-SVR model performed better than the EMD-ANN model. Duan *et al.*(2016) developed a hybrid EMD-SVR model for the short-term prediction of ocean waves, which performed better than the conventional statistical models and also than the wavelet decomposition-based SVR (WD-SVR) model.

With this backdrop, we have attempted to evaluate the appropriateness of empirical mode decomposition (EMD)-based neural network and support vector regression (SVR) approaches for forecasting wholesale prices of three major potato markets namely, Agra, Bangalore, and Mumbai. As the benchmark models, autoregressive integrated moving average (ARIMA), time delay neural network (TDNN) and SVR models have been employed for the comparative evaluation.

2. METHODOLOGIES

2.1 ARIMA Models

The ARIMA model was introduced by Box and Jenkins in the year 1970 as a generalisation of the ARMA model. In ARMA, the forecasted value of the variable is the linear combination of its past (lag) values and past (lag) errors. The general form of the forecasted value at time t generated by the ARMA (p, q) process can be expressed as:

$$y_t = c + \alpha_1 y_{t-1} + \ldots + \alpha_p y_{t-p} + \varepsilon_t - \beta_1 \varepsilon_{t-1} - \ldots - \beta_q \varepsilon_{t-q} \quad (1)$$

where α_i (i=1, 2…., p) and $\beta_j \left(j = 1,2 \ldots q \right)$ are the model parameters.

As the estimate approach is only available for stationary series, the first step in the ARIMA modelling process is to verify the series' stationarity. If 'd' times differencing is needed for the data to become stationary, then the ARIMA model is represented as ARIMA(p, d, q), where p and q refer to the order of the auto regression and moving average, respectively. Based on autocorrelation function (ACF) and partial autocorrelation function (PACF) many ARIMA models were selected. Model parameters are estimated by the method of maximum likelihood. For all estimated models, diagnostic testing for model adequacy is carried out using a plot of the residual ACF using portmanteau tests such the Ljung-Box tests. The most suitable ARIMA model is selected using the smallest Akaike Information Criterion (AIC) or Schwarz–Bayesian Criterion (SBC) value and the lowest root mean square error (RMSE).

The formula for AIC is given as

$$\mathrm{AIC} = -2ln\left(\mathrm{L}\right) + 2\mathrm{k} \quad (2)$$

where L is the maximum likelihood of the model and k is the number of estimated parameters in the model.

The formula for SBC is given as

$$\mathrm{SBC} = \ln\left(\frac{\mathrm{SSE}}{\mathrm{n}}\right) + \frac{\mathrm{p}}{\mathrm{n}}\ln\left(\mathrm{n}\right) \quad (3)$$

where SSE is the sum of squared errors, n is the number of observations and p is the number of estimated parameters of the model.

2.2 Time Delay Neural Network Models

The concept of 'artificial neural network' comes from the biological neural networks that allow learning by example from the representative data. ANNs, like the human brains, have neurons that are coupled to one another in various layers of the networks and these neurons are called nodes. A neural network structure usually contains an input layer, an output layer and one or more hidden layers. Each layer has one or more nodes. During the training process, the information inside one node is associated with random initial weights and biases and then, subsequently passed to the next layer, where it gets transformed by using an activation function. During the back propagation, the weights and biases are adjusted by the gradient descent method accordingly.

In a time delay neural network (TDNN), the lagged observations of the time series variable are utilised in the input layer. Determination of the number of layers and the number of nodes in each layer is carried out based on experimentation. For a single hidden layer TDNN with p input nodes, q hidden nodes and a single output node, the general expression for the forecasted value at time t is given by:

$$\hat{y}_t = h\left(u_0 + \sum_{j=1}^{q} u_j k\left(v_{0j} + \sum_{i=1}^{p} v_{ij} y_{t-i} \right) \right) \quad (4)$$

$\hat{y}_t$ is the forecasted value at time t, v_{ij} is the weight connection between the i^{th} input node and j^{th} hidden node, u_j is the weight between j^{th} hidden node and the output node. u_0 and v_{0j} denote the bias term. h and k represent the activation functions at the output and hidden layer respectively. The error function which is widely used for the training purpose is mean squared error and is given by

$$E = \frac{1}{n}\sum_{t=1}^{n}\left(y_t - \hat{y}_t\right)^2 \quad (5)$$

The weights and biases are updated by using the calculated gradients and a learning rate α. The learning rate is a hyperparameter that controls how frequently the parameters of an optimisation algorithm or a neural network are updated during training. Its impact on how fast or slowly the model learns and converges makes it a crucial parameter. In order to achieve sustained and effective training, selecting the appropriate learning rate is crucial. A very low value can delay the convergence while a high value can cause no convergence at all.

Mathematically the updated weight is given by

$$v_{ij} \leftarrow v_{ij} - \alpha . \frac{\partial L}{\partial v_{ij}} \quad (6)$$

where $\frac{\partial L}{\partial v_{ij}}$ is the partial derivative of the error function with respect to the weight v_{ij}.

2.3 Support Vector Regression Models

The support vector regression is an adaption of support vector machines. In SVR for solving non-linear regression problems, the inputs are first mapped non-linearly into a high-dimensional feature space (F), where they are linearly associated with the outputs. For a given dataset, $G = \left\{\left(x_i,\ q_i\right)\right\}_{i=1}^{n}$, where x_i and q_i represent the input vector and the scalar output, respectively, and n is the size of G, the regression equation is given by:

$$f\left(x\right) = w^T . \Phi\left(x\right) + b$$

where w is the weight vector, $\Phi\left(x\right)$ is a non-linear mapping function in the feature space and T denotes the transpose. The ε-insensitive loss function used for SVR formulation is given as

$$L_{\varepsilon}\left(f\left(x\right), q\right) = \begin{cases} \left|f\left(x\right) - q\right| - \varepsilon & if\ \left|f\left(x\right) - q\right| \geq \varepsilon \\ 0 & otherwise \end{cases} \quad (7)$$

To estimate the weight vector w and constant b, the following regularised risk function has to be minimised:

$$R\left(C\right) = \frac{C}{n}\sum_{i=1}^{n} L_{\varepsilon}\left(f\left(x_i\right), q_i\right) + \frac{1}{2}\left|w\right|^2 \quad (8)$$

Here, $\frac{1}{2}\left|w\right|^2$ is the regularisation term used to estimate the flatness of a function. C denotes the regularisation constant, which determines the trade-off between model flatness and empirical risk. C and ε are both user-determined parameters.

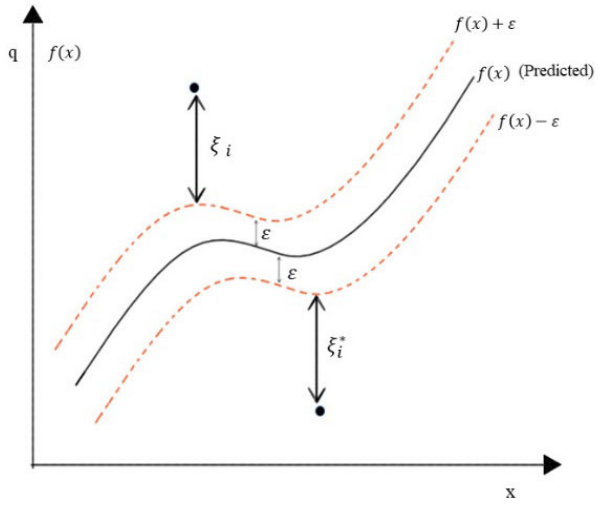

Fig. 1. (a) Diagram of support vector regression

Fig. 1(a) diagrammatically represents the SVR model, where the loss will be considered zero if the forecasted value falls within the ε-insensitive zone. ξ_i and ξ_i^* are the positive slack variables, which measure the deviation of actual observations from the boundaries of the ε-insensitive zone. After using the slack variables, the regularised risk function transforms to the following constrained form:

$$\text{Minimise: } \frac{1}{2}|w|^2 + C\sum_{i=1}^{n}\left(\xi_i + \xi_i^*\right)$$

$$\text{Subject to: } \begin{cases} q_i - \left(w'.\Phi\left(x_i\right)\right) - b \le \varepsilon + \xi_i \\ \left(w'.\Phi\left(x_i\right)\right) + b - q_i \le \varepsilon + \xi_i \\ \xi_i, \xi_i^* \ge 0 \quad for\, i = 1,\ldots..n \end{cases}$$

By utilising the Lagrangian multipliers and Karush-Kuhn-Tucker conditions, this optimisation problem can be further transformed as:

Maximize:

$$L_d\left(\alpha, \alpha^*\right) = -\varepsilon\sum_{i=1}^{n}\left(\alpha_i^* + \alpha_i\right) + \sum_{i=1}^{n}\left(\alpha_i^* - \alpha_i\right)q_i - \frac{1}{2}\sum_{i,j=1}^{n}\left(\alpha_i^* - \alpha_i\right)\left(\alpha_j^* - \alpha_j\right)K\left(x_i.x_j\right)$$

$$\text{Subject to: } \begin{cases} \sum_{i=1}^{n}\left(\alpha_i^* - \alpha_i\right) = 0 \\ 0 \le \alpha_i \le C, i = 1,\ldots.,n \\ 0 \le \alpha_i^* \le C, i = 1,\ldots.,n \end{cases}$$

After calculating the Lagrangian multipliers, the SVR function becomes:

$$f\left(x\right) = \sum_{i=1}^{n}\left(\alpha_i - \alpha_i^*\right)K\left(x_i.x_j\right) + b \tag{9}$$

Here, $K\left(x_i.x_j\right) = \Phi\left(x_i\right).\Phi\left(x_j\right)$. The kernel function providing the inner product of two vectors in the feature space is denoted by $K\left(x_i.x_j\right)$. The radial basis function (RBF) is the most commonly used kernel function and is mathematically expressed as $e^{\left(-\gamma\left\|x_i - x_j\right\|^2\right)}$.

2.4 Empirical Mode Decomposition

The empirical mode decomposition is a self-adaptive time series decomposition technique to decompose non-linear and non-stationary time series data into several oscillatory functions (intrinsic mode functions) along with a trend component (residue). However, each of these IMFs should follow two conditions. Firstly, the number of extrema (sum of the number of maxima and minima) and the number of zero crossings should differ by at most one. Secondly, for an IMF, the mean value of the envelope defined by local maxima and the envelope defined by local minima must be zero at all points.

The IMFs are extracted through a sifting procedure as follows.

i) The local maxima and minima of the time series data (y_t) are identified.

ii) All the local maxima points are connected by a cubic spline function to create the upper envelope y_{up}. Similarly, the local minima points are utilised to form the lower envelope y_{low}.

iii) The mean envelope $m_{11}(t)$ is formed by computing the mean values of the lower and upper envelopes.

$$m_{11}\left(t\right) = \frac{y_{up} + y_{low}}{2} \tag{10}$$

iv) In the next step, the mean envelope is subtracted from the actual data series.

$$h_{11}\left(t\right) = y_t - m_{11}\left(t\right) \tag{11}$$

v) The series $h_{11}(t)$ is then checked whether it is fulfilling all the necessary conditions of an IMF or not. If not, the sifting process is again followed on $h_{11}(t)$ until the necessary

conditions are satisfied. The process of obtaining the first IMF after the k^{th} iteration can be expressed as:

$$h_{l(k-1)}(t) - m_{lk}(t) = h_{lk}(t) = c_l(t) \quad (12)$$

vi) To ensure enough physical sense of both amplitude and frequency modulations, Huang et al. (1998) have proposed a conventional criterion,

$$\text{Stopping criterion}(SC) = \frac{\sum_{t=0}^{T}\left(h_{(k-1)}(t) - h_k(t)\right)^2}{\sum_{t=0}^{T} h_{k-1}^2(t)} \quad (13)$$

The value of SC lies between a predetermined limit of 0.2 to 0.3.

vii) After obtaining the first IMF, it is subtracted from the actual series,

$$y_t - c_l(t) = r_l(t) \quad (14)$$

Now, if $r_l(t)$ is not a monotonic function, then it is treated as a new series and the same sifting process is followed again to extract the second IMF.

viii) This sifting process is continued until the residue becomes a monotonic function from which no more IMF can further be extracted. The final residue after the extraction of the n^{th} IMF can be given as

$$r_{n-1}(t) - c_n(t) = r_n(t) \quad (15)$$

Therefore, the actual data series is finally decomposed into the following form:

$$y(t) = \sum_{j=1}^{n} c_j(t) + r_n(t) \quad (16)$$

Proposed ensemble hybrid model

After obtaining the subseries from the Original series by the EMD process, each IMF and residue are fitted and predicted through ANN or SVR. Then all the forecasted values are summed up to obtain the final forecast result. In this study decomposition was done in 'R' software using the 'EMD' package.

2.5 Evaluation of the Forecasting Accuracy

To compare the accuracy of the forecasting performance of the different models, two criteria have been followed and are given as follows.

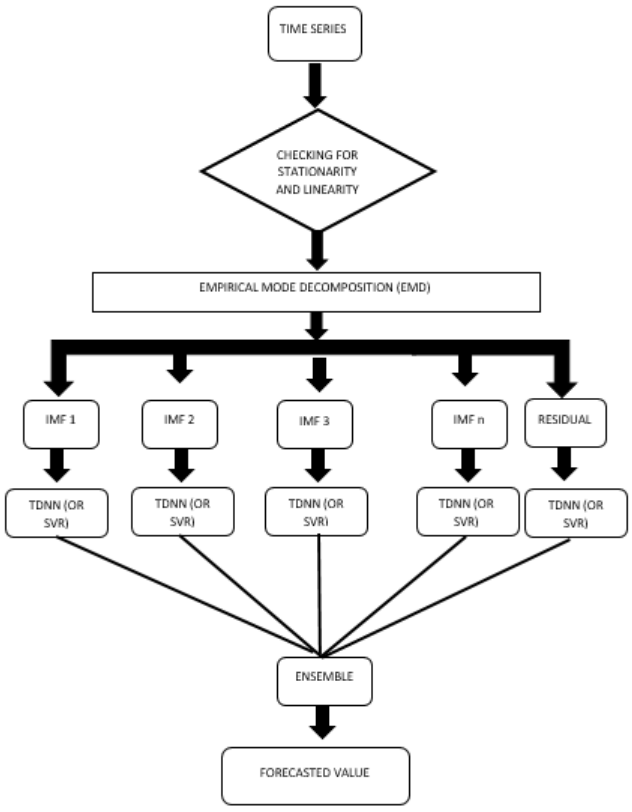

Fig. 1(b). Proposed ensemble hybrid model

Root Mean Square Error (RMSE)

RMSE is widely used for comparing the accuracy of different forecasting models. When comparing models, the one with the lower RMSE is generally preferred because it indicates a better overall fit to the data. It is the average of the squared errors and is given by the following expression

$$RMSE = \sqrt{\frac{\sum_{t=1}^{n}\left(y_t - \widehat{y}_t\right)^2}{n}} \quad (17)$$

Directional prediction statistics (D_{stat})

Directional prediction statistics are used in forecasting accuracy when the direction of change is a critical component of decision-making. These metrics provide insights into a model's ability to make correct directional calls. The expression for the D_{stat} is given by

$$D_{stat} = \frac{1}{n}\sum_{t=1}^{n} a_t \times 100\% \quad (18)$$

$$a = \begin{cases} 1 & if \left[y_{t+1} - y_t \right] \left[\hat{y}_{t+1} - \hat{y}_t \right] \geq 0 \\ 0, & otherwise \end{cases}$$

where y_t and $\hat{y}_t$ are actual and predicted values respectively

3. EMPIRICAL FINDINGS

3.1 Data and Implementation

For this study, monthly wholesale prices (₹/- per quintal) of potato from January, 2005 to December, 2020 for three markets namely, Agra, Bangalore, and Mumbai, have been collected from the National Horticulture Research and Development Foundation (http://nhrdf.org/en-us). The observations from January, 2005 to December, 2019 are used for training purposes and the last 12 months data points are used for the post-sample evaluation (January, 2020 to December, 2020).

The descriptive statistics of the wholesale prices of the three markets under study are presented in table 1. The average price of potato was the highest in the Bangalore market followed by the Mumbai and Agra markets. However, the highest price volatility was observed in the case of the Agra market, followed by the Mumbai and Bangalore markets.

Table 1. Descriptive statistics of the potato markets

Markets	Agra	Bangalore	Mumbai
Minimum	147	414	394
Maximum	2829	3200	3195
Mean	739.76	1211.77	1109.92
Standard deviation	452.38	490.29	452.34
Kurtosis	3.71	1.54	2.90
Skewness	1.69	1.15	1.41
CV (%)	61.15	40.46	40.75

The seasonal indices of different markets are given in table 2. Seasonal indices are calculated in the following way. Average for each month and the overall average are calculated. Seasonal index for each season is calculated by dividing the seasonal average by the overall average. The resulting seasonal indices represent how much each season deviates from the overall average. It can be seen that prices became higher for the months from July to December in the case of Agra and Mumbai markets, whereas no such pattern was observed for the Bangalore market. Hence, before proceeding further, seasonal adjustments of the data were carried out for the Agra and Mumbai markets.

Table 2. Seasonal indices of potato markets

	Agra	Bangalore	Mumbai
January	0.68	1.00	0.88
February	0.65	0.87	0.82
March	0.76	0.77	0.85
April	0.84	0.90	0.93
May	0.97	1.02	0.99
June	1.07	1.06	1.00
July	1.18	1.07	1.01
August	1.16	0.96	1.03
September	1.18	0.96	1.06
October	1.30	1.02	1.13
November	1.30	1.20	1.20
December	0.91	1.18	1.10

Augmented Dickey-Fuller (ADF) test was used to check the stationarity of the price data. The ADF test compares a test statistic against critical values to help identify whether a time series is stationary or non-stationary. The null hypothesis of the ADF test is that the time series data has a unit root, which means it is non-stationary. The alternative hypothesis is that the data is stationary, meaning it does not have a unit root. The results, shown in table 3, clearly indicate the stationary nature of the Agra and Bangalore market prices. However, Mumbai market prices were found to be non-stationary at level, which became stationary after the first differencing.

Table 3. Results of the ADF test

Markets	Level		1st Difference	
	Statistic	p -Value	Statistic	p-Value
Agra	-3.74	0.02	-	-
Bangalore	-3.81	0.02	-	-
Mumbai	-3.40	0.42	-6.54	<0.01

In the next step, the Brock, Dechert and Scheinkman (BDS) test was performed to check whether the data are non-linear or not. The BDS test compares the distribution of closest neighbour distances in the original data to those of surrogate data sets reflecting linear behaviour to look for nonlinearity in time series data. There are significant discrepancies between these distributions, which suggests nonlinearity. The results,

given in table 4, reject the assumption of linearity for all the price series under study.

Table 4. Results of the BDS test

	AGRA			
	Embedding Dimension =2		Embedding Dimension =3	
Epsilon	Statistic	p-Value	Statistic	p-Value
61.72	2.15	0.03	3.6	<0.01
123.35	2.46	0.01	2.98	<0.01
185.18	2.68	<0.01	2.67	<0.01
246.9	4.26	<0.01	3.5	<0.01
	BANGALORE			
	Embedding Dimension =2		Embedding Dimension =3	
Epsilon	Statistic	p-Value	Statistic	p-Value
86.33	3.36	<0.01	6.12	<0.01
172.66	2.82	<0.01	4.99	<0.01
259	2.05	<0.01	3.72	<0.01
345.33	1.79	0.07	2.91	<0.01
	MUMBAI			
	Embedding Dimension =2		Embedding Dimension =3	
Epsilon	Statistic	p-Value	Statistic	p-Value
69.02	3.54	<0.01	3.5	<0.01
138.03	3.99	<0.01	3.25	<0.01
207.05	3.75	<0.01	2.57	0.01
276.07	3.78	<0.01	2.29	<0.02

After determining the nature of the price series, the training sets were utilised for fitting the ARIMA, TDNN, SVR, EMD-TDNN and EMD-SVR models in this study.

Fitting of the ARIMA Models

The selection of the best ARIMA models was carried out on the basis of the lowest AIC and BIC values as well as the lowest RMSE values. ARIMA(1, 0, 1), ARIMA(1, 0, 1), and ARIMA(0, 1, 1) had been selected for the Agra, Bangalore and Mumbai markets, respectively. Residual diagnostics indicated that the residuals were well-behaved. Parameter estimates of the selected ARIMA models are presented in table 5.

Fitting of the TDNN Models

The TDNN models employed in this study consisted of one input layer, one hidden layer with several nodes and a single output node. Logistic and linear activation functions were used in the hidden and output layers, respectively. TDNN models with 1 input and 7 hidden nodes, 3 input and 8 hidden nodes and 4 input and 8 hidden nodes were found to be the best for the Agra, Bangalore and Mumbai markets, respectively. In this study fitting of TDNN model was done in 'R' software by using the package 'nnet'.

Table 5. Parameter estimates of the ARIMA models

Markets	Parameters	Estimates	p value
Agra	C	693.74 (102.28)	<0.01
	α_1	0.90 (0.03)	<0.01
	β_1	0.40 (0.07)	<0.01
Bangalore	C	1132.32 (119.61)	<0.01
	α_1	0.89 (0.03)	<0.01
	β_1	0.20 (0.08)	0.02
Mumbai	C	7.90 (12.20)	0.52
	β_1	0.30 (0.08)	<0.01

Note: Standard errors are mentioned in paranthesis.

Fitting of the SVR Models

The SVR models for potato market prices were built with the following specifications (Table 6). RBF (radial basis function) was used in each case as the kernel function. SVR model fitting was done in 'R' software by using the package "e1071". The lags for the models were selected based on experimentation. The lags were varied from 1 to 6. The optimum hyper parameter combination was obtained by the grid search method. 10 fold cross-validation was done for overcoming the over fitting problem. A small value of C can create a complex model as it allows more data points and support vectors while a large value of C can produce a simpler model as it penalizes error more heavily developing a model with fewer support vectors. Similarly a smaller value of epsilon creates a narrow margin leading to a more accurate fit to the training data and a large value creates a wider margin leading to a more generalized model. The gamma parameter essentially sets the scale of the RBF kernel.

Table 6. Specifications of the SVR models

Markets	No. of lag(s) used	C	ϵ	γ
Agra	1	500	0.10	0.0001
Bangalore	1	1000	0.05	0.0001
Mumbai	1	1000	0.20	0.0001

Fitting of the EMD-based models

Each series decomposed by EMD resulted in six IMFs and one residue. For a particular series, each mode was fitted with the same class of model (either TDNN or SVR). The modes were divided into training and testing. The last 12 months were meant for testing purpose. The developed model for each mode was used to forecast the respective components and the forecasted value from all IMFs and the residue were summed up to obtain the final forecasted value. The fitting procedure of different models is discussed in the methodologies. In this way, EMD-based TDNN and SVR models were developed. Fig. 2(a) – 2(c) illustrates the IMFs and residues obtained through the EMD process for the Agra, Bangalore and Mumbai markets, respectively.

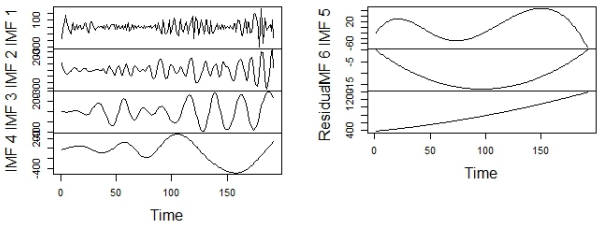

Fig. 2(a). IMFs and residual for the Agra market

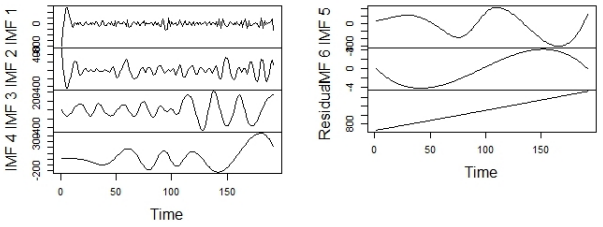

Fig. 2(b). IMFs and residual for the Bangalore market

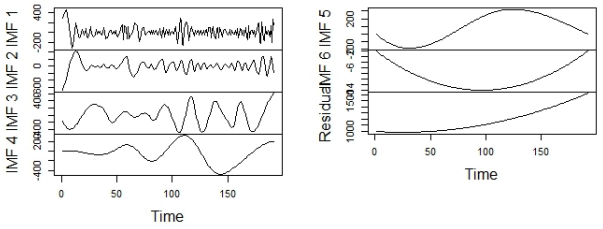

Fig. 2(c). IMFs and residual for the Mumbai market

4. DISCUSSION

Different models employed in this study were compared both in terms of the RMSE (Table 7) and the D statistic values (Table 8). In table 7 RMSE values for both training and testing are given for different models. From the table values, it can be observed that the TDNN and SVR models are performing well compared to the ARIMA model in the case of Agra and Bangalore markets. However, the superior performance of the ARIMA model is evident as compared to the TDNN model for the Mumbai market. It is worth mentioning at this juncture that even though the price series is non-linear, TDNN fails to perform better than the linear ARIMA model due to its inability to handle non-stationarity. The experimental results also reveal the comparative superiority of the EMD-SVR model for the Agra and Bangalore markets and the EMD-TDNN model for the Mumbai market. Moreover, all the EMD-based models have performed better than the other competing models. It is noteworthy that the comparative performance of the TDNN and the SVR models are not similar in the case of with and without EMD. It suggests that the superiority of one model over the other one cannot be generalized over EMD or any such decomposition techniques. The figures 3(a to c) represents the best models for the respective markets. The graphs of only test set and the forecasted results are represented in the figures 4(a to c).

Table 8. Comparison of forecasting accuracy in terms of D statistic values

Markets	ARIMA	TDNN	SVR	EMD-TDNN	EMD-SVR
Agra	75	83.33	75	91.67	91.67
Bangalore	75	83.33	83.33	83.33	91.67
Mumbai	91.67	58.33	58.33	91.67	66.67

5. CONCLUSION

In the current investigation, we have assessed the suitability of EMD-based TDNN and SVR models for forecasting wholesale potato prices of the Agra, Bangalore, and Mumbai markets. As the benchmark models, ARIMA, TDNN and SVR models have also

Table 7. Comparison of forecasting accuracy in terms of RMSE values

MARKETS	ARIMA		TDNN		SVR		EMD-TDNN		EMD-SVR	
	TRAINING	TESTING	TRAINING	TESTING	TRAINING	TESTING	TRAINING	TESTING	TRAINING	TESTING
Agra	160.34	738.12	217.47	496.19	109.23	263.83	167.15	223.12	162.44	201.34
Bangalore	229.25	960.96	350.54	841.43	154.65	345.77	167.20	268.39	172.74	201.9
Mumbai	123.01	645.86	382.59	733.11	127.11	275.37	132.06	210.15	152.58	217.72

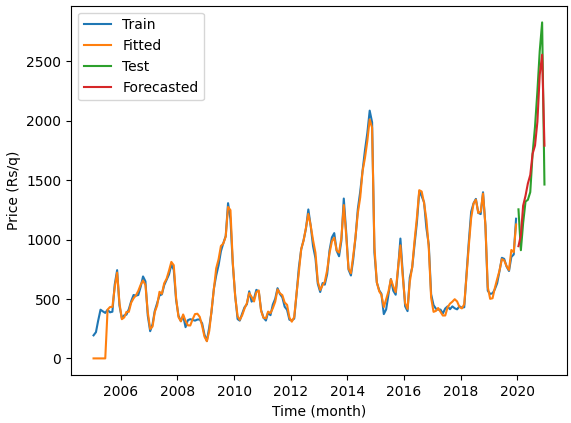

Fig. 3(a). Actual and (EMD-SVR) predicted series for potato wholesale price of Agra market

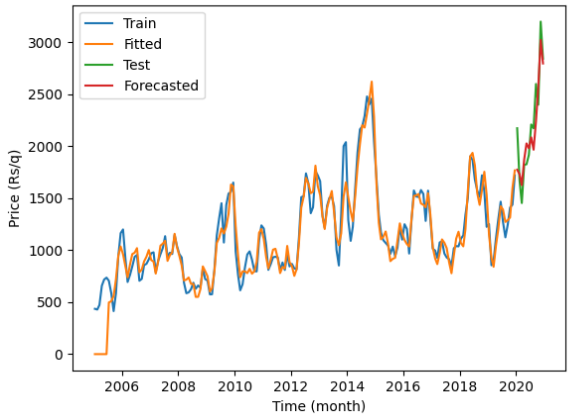

Fig. 3(b). Actual and (EMD-SVR) predicted series for potato wholesale price of Bangalore market

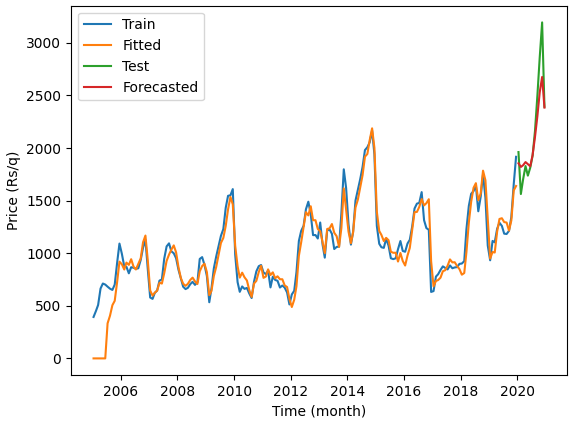

Fig. 3(c). Actual and (EMD-TDNN) predicted series for potato wholesale price of Mumbai market

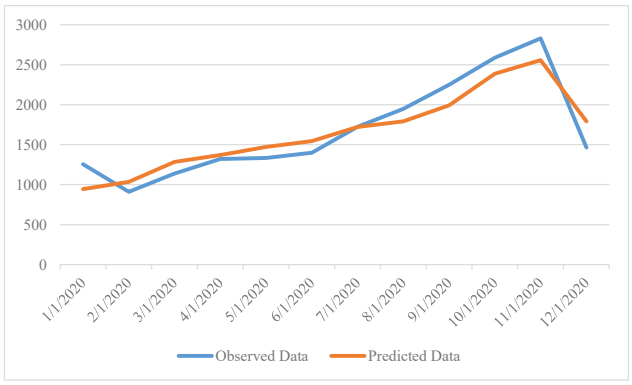

Fig. 4(a) Garph of test series and predicted series of Agra market

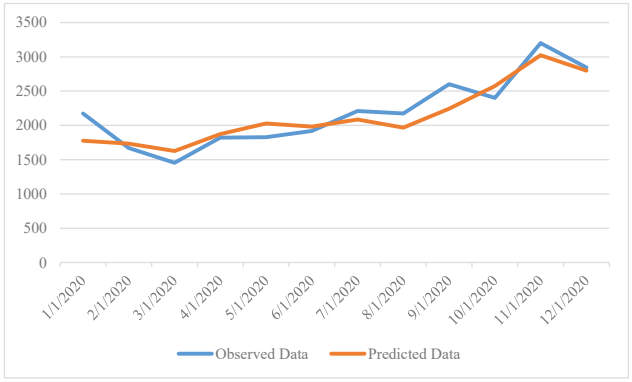

Fig. 4(b) Garph of test series and predicted series of Bangalore market

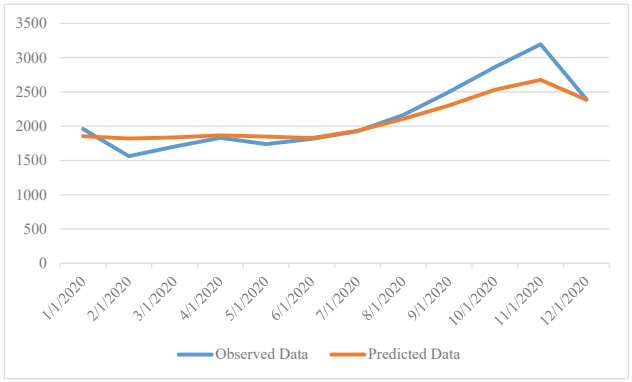

Fig. 4(c). Garph of test series and predicted series of Mumbai market

been employed. The experimental results clearly demonstrate the comparative superiority of the EMD-SVR model for the Agra and Bangalore markets and the EMD-TDNN model for the Mumbai market in terms of root mean squared error values and turning point predictions. The results also indicate that the EMD based models are more capable of handling non-stationary and non-linear data compared to other models. As agricultural price data are nonlinear and nonstationary, these decomposition-based models can perform better in fitting and forecasting of other

agricultural commodities. Still EMD methods suffers from some problems like mode mixing and endpoint effects. In mode mixing multiple IMFs can capture components of different scales or frequencies which can make the interpretation challenging. The generalisation of this study includes the application and comparative evaluation of other mode decomposition techniques such as Ensemble Empirical Mode Decomposition (EEMD), Complete Ensemble Empirical Mode Decomposition with Adaptive Noise (CEEMDAN), Variational mode decomposition (VMD), etc. for improved forecasting of non-stationary and non-linear agricultural commodity price series.

REFERENCES

An, X., Jiang, D., Zhao, M. and Liu, C. (2012). Short time prediction of wind power using EMD and chaotic theory. *Communications in Nonlinear Science and Numerical Simulation*, **17(2)**, 1036-1042.

Broock, W.A., Scheinkman, J.A., Dechert, W.D., and LeBaron, B. (1996). A test for independence based on the correlation dimension. *Econometric Reviews,* **15(3),** 197-235.

Bo, Z., Aiguo, S., (2013). Empirical Mode Decomposition Based LSSVM for Ship Motion Prediction. *Advances in Neural Networks*, Springer, Berlin Heidelberg.

Chen, C.F., Lai, M.C. and Yeh, C.C. (2012). Forecasting tourism demand based on empirical mode decomposition and neural network. *Knowledge-Based Systems*, **26**, 281-287.

Cheng, C., and Wei, L., (2014). A novel time-series model based on empirical mode decomposition for forecasting TAIEX. *Economic Modelling*, **36**, 136-141.

Clements, Michael P., Philip Hans Franses, and Norman R. Swanson. 2004. Forecasting economic and financial time-series with non-linear models. *International Journal of Forecasting,* **20**, 169-83.

Das, P., Jha, G.K., Lama, A. and Kumar, M. (2019). Price forecasting using EMD SVR hybrid model: An ensemble learning paradigm. *In proceedings of Eighth International Conference on Agricultural Statistics*, New Delhi, India.

Das, P.K. and Das, A. (2021) Application of nonlinear stochastic single source of error state space models in the forecasting of mobile subscribers in India. *International Journal of Data Science*, **5(4)**, 333-357.

Dickey, D.A. and Fuller, W.A. (1979). Distribution of the Estimators for Autoregressive Time Series With a Unit Root. *Journal of the American Statistical Association,* **74(366),** 427.

Duan, W. Q. and Stanley, H. E. (2011) Cross-correlation and predictability of financial return series. *Physica A*, **390(2)**, 290-296.

Duan, W.Y., Han, Y., Huang, L.M., Zhao, B.B. and Wang, M.H., (2016). A hybrid EMD-SVR model for the short-term prediction of significant wave height. *Ocean Engineering*, **124**, 54-73.

Guo, Z., Zhao, W., Lu, H. and Wang, J. (2012). Multi-step forecasting for wind speed using a modified EMD-based artificial neural network model. *Renewable Energy*, **37(1)**, 241-249.

Huang, N.E., Shen, Z., Long, S.R., Wu, M.L., Shih, H.H., Zheng, Q., Yen, N.C., Tung, C.C. and Liu, H.H. (1998). The empirical mode decomposition and Hilbert spectrum for nonlinear and nonstationary time series analysis. *In proceedings of the Royal Society London A*, **454**, 909-995.

Kajitani, Y., Hipel, K.W. and McLeod, A.I.(2005). Forecasting nonlinear time series with feed-forward neural networks: a case study of Canadian lynx data. *Journal of Forecasting,* **24(2)**, 105-117.

Kazem, A., Ebrahim, S., Farookh, K.H., Morteza, S., and Omar, K.H. (2013). Support vector regression with chaos-based firefly algorithm for stock market price forecasting. *Applied Soft Computing*, **13**, 947-958.

Lu, Chi-Jie. and Yuehjen, E. Shao. (2012). Forecasting computer products sales by integrating ensemble empirical mode decomposition and extreme learning machine. *Mathematical Problems in Engineering*, **2012**, 831201.

Smola, A.J. and Scholkopf, B. (2004). A tutorial on support vector regression. *Statistics and Computing*, **14**, 199-222.