Available online at www.isas.org.in/jisas
Journal of the Indian Society of Agricultural Statistics 76(3) 2022 141–150

Variational Mode Decomposition based Machine Learning Models Optimized with Genetic Algorithm for Price Forecasting

Pankaj Das[1*], Achal Lama[1] and Girish Kumar Jha[2]

[1]ICAR- Indian Agricultural Statistics Research Institute, New Delhi
[2]ICAR- Indian Agricultural Research Institute, New Delhi

Received 30 September 2022; Revised 03 January 2023; Accepted 20 January 2023

SUMMARY

Accurate and timely price information and forecasting help in making efficient plans and strategies. Non-linearity and non-stationarity behaviour of price data create problems in price forecasting. In this paper, variational mode decomposition (VMD) based optimised genetic algorithm (GA) hybrid machine learning (ML) models have been proposed. The VMD algorithm is employed to decompose the price data into intrinsic mode functions (IMFs) which is further forecasted using ML models namely support vector regression (SVR) and random forest (RF). The practical use of the SVR and RF models is limited because the accuracy of ML models heavily depends on a proper setting of hyper-parameters. Therefore, these model hyper-parameters are optimized using GA. Further, the forecasted values of IMFs through the GA optimised SVR and RF are aggregated for the final forecast. The results of the proposed model are benchmarked with the comparative models. The proposed VMD-GA-RF and VMD-GA-SVR models are tested on the weekly onion price of the Delhi and Nashik market. The results clearly demonstrate that the combination of VMD and GA optimized models can improve the performance of the prediction of the dataset.

Keywords: Price forecasting, Machine learning, VMD, Genetic algorithm, Support vector regression, Random forest.

1. INTRODUCTION

Price forecasting is an important topic for any commodity or product. The behaviour of a commodity or product price is generally dynamic and nonlinear. Besides this, the price of a commodity is also affected by the seasonality and prices of other related commodities. The accurate information related to the price of a commodity helps different concerns like producers, buyers, traders, etc. for making their plan. Even the policy makers and government also use this information for making different policies. Hence, price forecasting of a commodity or product is gradually becoming a hot research area. For over a decade, researchers have tried to build a perfect model for price forecasting. Several researchers such as Antiwi *et al.* (1995), Balke and Fomby (1994), De and Kumar (1992) and Giles *et al.* (2001) stated that the non-linearity and non-stationarity behaviour as major hurdle in model building process. Traditional models like Autoregressive integrated moving average (ARIMA) are not suitable under nonlinear paradigm of price data series. Machine learning (ML) techniques like artificial neural network (ANN) (Jha and Sinha, 2014; Choudhury *et al.*, 2019), support vector regression (SVR) (Das *et al.*, 2020), random forest (Kane *et al.*, 2014) etc. have gained popularity to deal with non-linearity problems. The non-stationary pattern in price data also hinders model fitting. Indeed, it has been established that a single forecast model is not sufficient to deal with non-linearity and non-stationarity simultaneously.

In the last few years, many studies have been conducted to enhance the prediction performance of ML models. The studies were mostly on the development of hybrid ML models in which ML models were combined with different statistical and mathematical models. The hybrid model development in time series can be classified into four categories *viz.* 1. Combination of ML and statistical models like principal component

Corresponding author: Pankaj Das
E-mail address: pankaj.das2@icar.gov.in

*Paper presented in Dr. G.R. Seth Memorial Young Scientist Award Session on November 15, 2022 in the 73rd Annual Conference of the Indian Society of Agricultural Statistics held at Sher-e-Kashmir University of Agricultural Sciences and Technology of Kashmir, Srinagar, J&K, India.

analysis (Meng *et al.*, 2020), fuzzy clustering (Peng *et al.*, 2015), co-integration model (Das *et al.*, 2021) etc. 2. ML model optimization using evolutionary optimization techniques such as GA (Kumar, 2021) and particle swarm optimization (PSO) (Fadlallah, 2021). 3. A hybrid ML model based on decomposition in conjunction with preprocessing techniques such as Wavelet (W) (Khandelwal *et al.*, 2015) and empirical mode decomposition (Das *et al.*, 2020). The studies reported that these decomposition or sub-time series based hybrid models have achieved greater accuracy compared to single ML models. 4. Hybrid ML model that combines two or more methods, such as Wavelet-based PSO optimised SVR model (PSO-W-SVR) (Ghimire *et al.*, 2019), SVR with wrapper-based feature selection (Karasu *et al.*, 2020).

In time series literature, the decomposition methods like wavelet, EMD, and EEMD are effective in developing hybrid models. However, these methods have some drawbacks and limitations. Lieu *et al.* (2014) state how the performance and computational time of wavelet decomposition depend on the choice of mother wavelet and wavelet indices. Shi and Yang (2017) demonstrate that the mode mixing effect on the time-frequency representation degrades the accuracy of the model. In EMD, the process of error envelope estimation may be extended more and more which results in lower efficiency. The EEMD process suffers from mode mixing which produces unstable IMFs (Wu and Huang, 2009). Besides this, the decomposition of EMD depends on the stopping criteria and end point effect. Dragomiretskiy and Zosso (2014) developed an adaptive and non-recursive signal analysis technique called the variational mode decomposition (VMD) to resolve the drawbacks of EMD and EEMD. The VMD is similar to EMD process i.e decomposition of time series into IMFs. The advantages of VMD can be attributed to the fact that it is robust to sampling and noise and has excellent performance in frequency search and separation (Dragomiretskiy and Zosso, 2014). With these advantages, VMD has been successfully applied in different areas like hydrology (Seo *et al.*, 2018), renewable energy (Sun *et al.*, 2016), financial and economic fields (Lahmiri, 2016). Although the decomposition methods reduce the error, the researchers (Cong and Meesad, 2015; Kisi *et al.*, 2015; Yao *et al.*, 2018; Wang *et al.* 2020) reported that the addition of an optimization technique can further improve its prediction accuracy.

Our focused review of the literature has revealed that most of the researchers have restricted themselves to study the ensemble ML approaches for price forecasting of agricultural commodities. The incorporation of the decomposition method and optimized hyper-parameter information into the ML model for price forecasting of the product has not been attempted yet. In the present study, we have addressed this research gap and proposed a decomposition based optimized ensemble ML model. Our assumption was that this model would easily deal with both non-linear and non-stationary problems with higher accuracy in price data. The proposed hybrid model has been illustrated successfully on real data sets on the monthly onion prices of two Indian markets (Delhi and Nashik) and forecasting performances were compared using different statistical measures. The remaining portion of the paper is divided into materials and methods, results and discussion followed by the conclusion section.

2. MATERIALS AND METHODS

2.1 Data source

In the present study, the weekly onion prices of two different markets, i.e., Delhi and Nashik were used. The price datasets were obtained from the Agmarknet website (https://agmarknet.gov.in). The price series consists from July, 2005 to August, 2022 for the Delhi market and from January, 2009 to August, 2022 for the Nashik market depicted by figure 1.

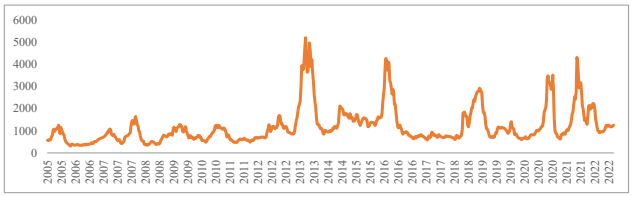


Fig. 1a. Time plot of weekly onion price of Delhi market (Rs/Q)

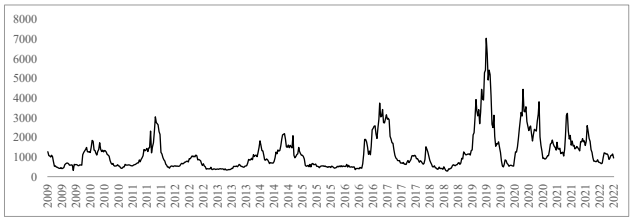


Fig. 1b. Time plot of weekly onion price of Nashik market (Rs/Q)

2.2 Methods

Variational mode decomposition (VMD) model

Variational mode decomposition (VMD) is fully adaptive and non-recursive algorithm for time-

frequency signal analysis. The concept of VMD is originated from Huang *et al.* (1998). VMD decomposes a timeseries data (y) into *m* numbers of intrinsic mode function (IMF). Dragomiretskiy and Zosso (2014) introduced the constrained variational formulation for yielding the IMFs can be written as

$$\min_{\{u_k\},\{\omega_k\}} \left\{ \sum_{k=1}^{m} \left\| \partial t \left[\left(\delta(t) + \frac{j}{\pi t} \right) * u_k(t) \right] e^{-j\omega_k t} \right\|_2^2 \right\}, \; s.t. \sum_{k=1}^{m} u_k(t) = y \tag{1}$$

where δ is the dirac function; j^2 = - 1; $\|\cdot\|_2 = L_2$ distance; ω_k =central frequency; *=the convolution; $u_k(t) = A_k(t)\cos(\phi_k(t))$, the k[th] IMF; ϕ_k =a non-decreasing function and A_k=a non-negative function. Further after applying augmented Lagrangian method, the unconstrained formulation i.e. loss function

$$L(\{u_k\},\{\omega_k\},\lambda) = \alpha \sum_{k=1}^{m} \left\| \partial t \left[(\delta(t) + \frac{j}{\pi t}) * u_k(t) \right] e^{-j\omega_k t} \right\|_2^2 + \left\| y(t) - \sum_{k=1}^{m} u_k(t) \right\|_2^2 + \left\langle \lambda(t), y(t) - \sum_{k=1}^{m} u_k(t) \right\rangle \tag{2}$$

where *L* denotes the augmented Lagrangian,, λ the Lagrange multiplier and <*a,b*> are the scalar product of *a* and *b*. The minimax point (saddle point) of the *L* is obtained by updating $u_k^{n+1}, \omega_k^{n+1}$ and λ_k^{n+1} in a sequence of iterative sub-optimizations using the alternate direction method of multipliers (ADMM). The final updated formulation of the optimization can be expressed as

$$\hat{u}_k^{n+1}(\omega) = \frac{\hat{y}(\omega) - \sum_{i<k} \hat{u}_i^{n+1}(\omega) - \sum_{i>k} \hat{u}_i^{n}(\omega) + \hat{\lambda}^n(\omega)/2}{1 + 2\alpha(\omega - \omega_k^n)^2} \tag{3}$$

$$\omega_k^{n+1} = \frac{\int_0^\infty \omega \left| \hat{u}_k^{n+1}(\omega) \right|^2 d\omega}{\int_0^\infty \left| \hat{u}_k^{n+1}(\omega) \right|^2 d\omega} \tag{4}$$

$$\hat{\lambda}^{n+1}(\omega) = \hat{\lambda}^n(\omega) + \tau \left[\hat{y}(\omega) - \sum_k \hat{u}_k^{n+1}(\omega) \right] \tag{5}$$

where ^ denotes the Fourier transform, *n* is the iteration number, α is a quadratic penalty factor and τ is the time step of dual ascent. The convergence state of the model iteration process is defined as

$$\sum_{k=1}^{m} \frac{\left\| \hat{u}_k^{n+1}(\omega) - \hat{u}_k^{n}(\omega) \right\|_2^2}{\left\| \hat{u}_k^{n}(\omega) \right\|_2^2} < \varepsilon \tag{6}$$

where ε is the user-defined coefficient for the judgement of model convergence. The detail of mathematical background of VMD was also described in Dragomiretskiy and Zosso (2014). The advantage of VMD compared to other existing techniques like EMD, EEMD etc. is its non-sensitivity to noise and sampling. The VMD is a non-recursive algorithm and is free from the problem of mode mixing. The VMD process is summarised as:

Step 1: Initialize $\{u_k^1\}, \{\omega_k^1\}, \hat{\lambda}^1$ and n .

Step 2: update the value of $\{\hat{u}_k^{n+1}\}, \{\omega_k^{n+1}\}$ and $\hat{\lambda}_{n+1}$ according to equation 3 to 5.

Step 3: Check the convergence condition using equation 6. Continue the step 3 to 5 until it meets the convergence condition.

Steps 4: the corresponding mode subsequences are according to the given mode number.

Support vector regression (SVR) model

Support vector regression (SVR) is a nonlinear prediction model derived from the classic support vector machines algorithm based on Vapnik's concept (Vapnik, 1998) of support vectors. The purpose of SVR is always to minimize the error by adding the hyper plane and maximizing the margin between prediction and actual values. SVR tries to a function that approximate functional dependency between target $T = \{t_1, t_2, \ldots, t_n\}$ defined on R and input $X = \{x_1, x_{2,\ldots}, x_n\}$ where $x_i \in R^n$ and n is the number of data points. The functional form of the approximation can be written as (Das *et at.*, 2020):

$$f(x) = w\varphi(x) + b \tag{7}$$

where $\varphi(x)$ is mapping function that maps the input space vector x to a high dimensional feature space, *w* denotes a weight vector and *b* is a bias term. The proper values of *w* and *b* can be obtained minimizing the following regularized risk function:

$$R_{SVR}(C) = C\frac{1}{n}\sum_{i=1}^{n} L(f(x_i) - t_i) + \frac{1}{2}w^2 \tag{8}$$

where $C\frac{1}{n}\sum_{i=1}^{n} L(f(x_i) - t_i)$ is the empirical error and $\frac{1}{2}w^2$ is the measure of function flatness. The penalty function *C* is constant (C>0) that computes the trade-off between the empirical error and the model complexity (regularization factor). Vapnik proposed a ϵ-insensitive loss function for estimation of empirical error. The functional of the ϵ-insensitive loss function as follows:

$$L(f(x_i)-t_i)=\begin{cases} 0 & if\ \ |f(x_i)-t_i|\le\varepsilon \\ |f(x_i)-t_i|-\varepsilon & otherwise \end{cases} \quad (9)$$

where ϵ denotes the error tolerance.

Then, the optimum parameters are obtained in the following equation by formulating the constrained optimization problem as (Das *et al.*, 2020):

$$\text{minimize } \frac{1}{2}w^2+C\sum_{i=1}^{n}(\xi_i^-+\xi_i^+) \quad (10)$$

$$\text{s.t. } \begin{cases} t_i-(w\varphi(x_i)+b)\le\varepsilon+\xi_i^+ \\ (w\varphi(x_i)+b)-t_i\le\varepsilon+\xi_i^- \\ \xi_i^-,\xi_i^+\ge 0, i=1,2,...,n \end{cases}$$

ξ_i^- and ξ_i^+ are the positive slack variables which denote lower and upper excess deviations respectively. In general, the constrained optimization problem (eq. 4) is very difficult to solve. For this, a dual form of the problem is formulated using Lagrange multipliers.

$$\min_{a,a^*}\frac{1}{2}\sum_{i,j}^{n}(a_i-a_i^*)(a_j-a_j^*)Ker(x_i,x_j)+\varepsilon\sum_{i}^{n}(a_i-a_i^*)+\sum_{i}^{n}y_i(a_i-a_i^*) \quad (11)$$

$$\text{s.t. } \sum_{i}^{n}(a_i-a_i^*)=0 \text{ and } 0\le a,a^*\le C, i,j=1,...,n$$

The final solution obtained is shown below (Das *et al.*, 2020):

$$f(x)=\sum_{i=1}^{n}(a_i-a_i^*)K(x_i,x_j)+b \quad (12)$$

In the above equation (12) a_i and a_i^* are Lagrange multipliers and the term $K(x_i,x_j)$ represents the kernel function map the input space into some higher dimensional space. The most well-known kernel functions in the literature are the polynomial function, radial basis function (RBF), and Gaussian function. RBF is employed as the kernel function in the present research, and the form is specified as follows (Vapnik, 1995):

$$K(x_i,x_j)=\exp(-\gamma x_i,x_j) \quad (13)$$

where γ is the kernel parameter. The combination of C, ϵ and the kernel function parameter γ is crucial for SVR's optimal performance and excellent accuracy. Therefore, it is vital to optimise these parameters using reliable algorithms that can choose their ideal combination on those hyper-parameters. The GA will function as an optimised algorithm for these SVR model parameters.

Random forest (RF) model

Random Forest is a machine learning-driven ensemble method based on Classification and Regression Trees (CART) and the bootstrapping aggregation (Breiman, 2001). It applies the Bagging integrated learning theory's random sub-space approach (Ho, 1998). Bootstrap samples from a data set are used to construct unpruned CART trees. These multiple decision trees are used to provide a joint forecast result that is extremely accurate. The only source from which the variables for each split are selected is a random selection of predictor variables. The forest's trees are all made from individually collected samples and have the same distribution. Individual trees are gathered together to form a forest.From the complete forest, the response variable is predicted as an average. Let $T_N=\{(\mathbf{d_1},\mathbf{z_1}),(\mathbf{d_2},\mathbf{z_2}),...,(\mathbf{d_N},\mathbf{z_N})\}$ be the training set of independent and identically distributed random vectors (**D, Z**) containing N examples. Here $\mathbf{D}=(D^1,D^2,...,D^p)\in\mathbf{R}^p$ represents the features (explanatory variables), p is the number of features, $\mathbf{Z}\in\mathbf{R}$ is the target (dependent variable), and **R** represents the real number set. An integer m_{try} denotes number of candidate features randomly selected to split in each non-leaf node. At each node, the best feature at each m_{try} is picked and the node is split into two child nodes.

In general, an RF is a regression consisting of a collection of numerical tree predictors $\{Q(\mathbf{d},\Theta_l),l=1,2,...,N_{tree}\}$. $\{\Theta_l\}$ are independent identically distributed random vectors drawn from the distribution of random vectors **Z** and **D.** N_{tree} is the number of trees in the forest. The RF numerical predictor is obtained by following formula,

$$\frac{1}{N_{tree}}\sum_{l=1}^{N_{tree}}Q(\mathbf{d},\Theta_l) \quad (14)$$

The root mean squared error (RMSE) can be written as $\text{RMSE}=\sqrt{E_{\mathbf{D},\mathbf{Z}}(\mathbf{Z}-Q(\mathbf{D}))^2}$. RMSE is one of the evaluation criteria of RF model. In software programming, m_{try} and N_{tree} are the important hyper-parameter RF modelling. Small value of m_{try} leads the model closer to the distribution of training data i.e. overfitting while high values lower down the computational speed. Small value of N_{tree} causes insufficient training and high value of N_{tree} increases the computational complexity. In this study, these two parameters m_{try} and N_{tree} will be optimised through GA.

Genetic algorithm (GA)

Genetic algorithm (GA) is a stochastic optimization and search method that mimics biological evolution as a problem-solving strategy (Zhang *et al.*, 2015). The development of the evolutionary algorithm was based on genetic principles such as selection, mutation, and crossover (Sivanandam and Deepa, 2007). An optimization problem's solution is found by GA using stochastic search across a solution space. For solving problems, the method uses chromosomes, which are strings of integers. The each integers in the chromosome is called gene. The process of GA can be described as the following steps:

1. Choose the initial population of individuals.
2. Evaluate the fitness of each individual in that population.
3. Repeat on this generation until termination: (time limit, sufficient fitness achieved, etc.)
   a. Select the best-fit individuals for reproduction.
   b. Breed new individuals through crossover and mutation operations to give birth to offspring.
   c. Evaluate the individual fitness of new individuals.
   d. Replace least-fit population with new individuals.

The GA algorithm was used to fix the hyper-parameters of SVR i.e. C, ϵ, γ and RF *i.e.* m_{try} and N_{tree}. The detail of GA optimization process was described in the flowchart (figure 3).

Proposed VMD based GA optimised SVR and RF models

For nonlinear and time series data, VMD based GA optimised ML models have been proposed in this paper. While ML model dealt with non-linearity pattern in data, VMD was used to deal with non-stationary problem in data. SVR and RF models were used in this work to make predictions. The improvement of ML models' hyper-parameters, however, is crucial for model accuracy. In this situation, GA was used to improve the ML models. A time series of data is broken down into IMFs by VMD. The GA-optimized SVR and RF model trained and forecasted each of the IMFs. By averaging all of the IMFs' final forecast values, the series' final forecast was determined. Fig. 3 provides a summary of the entire investigation. An R package VMDML (Das *et al.*, 2022) has been also developed for the analysis.

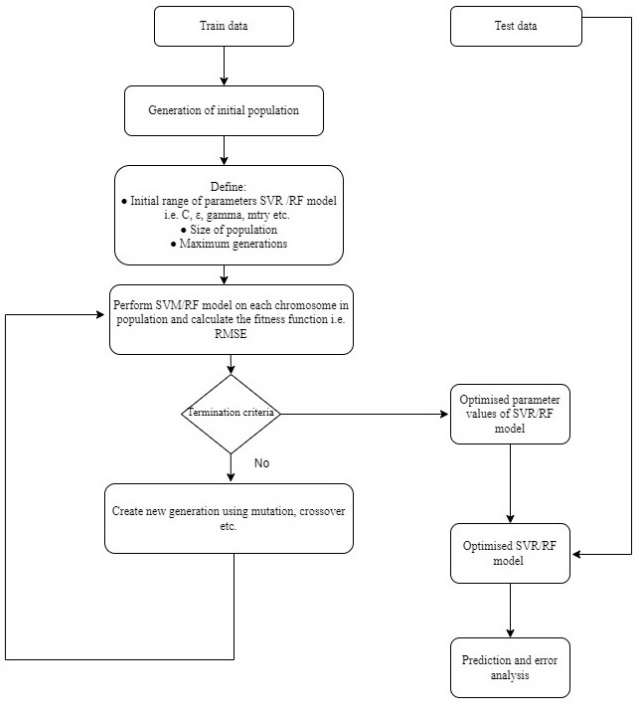


Fig. 2. Flowchart for GA based SVR/RF model

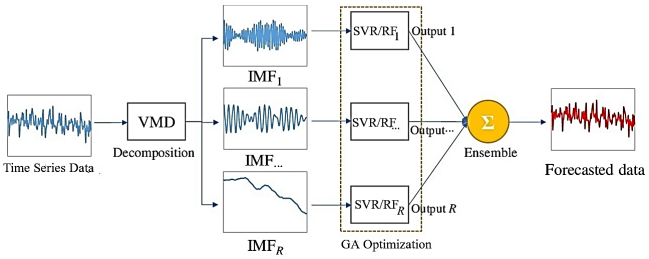


Fig. 3. Proposed VMD based GA optimized ML model

3. RESULTS AND DISCUSSION

To begin with we present the summary statistics of the datasets in the Table 1 to have an idea about it. It was observed that both the price series were positively skewed. Further Jarque-Bera test (Jarque and Bera, 1987) was used to check the normality behaviour of the both market data series (Table 1). The p values of the test was less than 0.01 for the data series. It indicated that the variables follow non-normal distribution and datasets were leptokurtic in nature. The entire analysis was carried out using "VMDML" and "GA" R packages.

After having a closer look at the important statistics of the dataset, we proceed to stepwise tests. We start with the test for stationarity. The test results (Table 2) guided us to the first order differencing of the series.

Table 1. Descriptive statistics and Normality test of the monthly data series

Descriptive statistics	Delhi	Nashik
Mean	1179	1183
Median	943	875
Maximum	5191	7050
Minimum	311	288
Std. Dev.	807.1830	922.4729
CV (%)	68.4799	77.9888
Skewness	2.1145	2.2837
Kurtosis	4.8844	7.0897
Jarque-Bera Test	1251.7102	1756.8120

Table 2. Stationarity test of data series

Series		Augmented Dickey-Fuller		Phillip-Perron	
		t-statistic	Prob.	t-statistic	Prob.
Delhi	Level	-1.6325	0.25	-1.5591	0.72
	1st difference	-8.8337	<0.001	-12.073	<0.001
Nashik	Level	1.6325	0.24	1.3232	0.13
	1st difference	-3.6971	<0.001	-4.3890	<0.001

Next, we checked the non-linearity of the data series using the BDS (Brock *et al.*, 1996) test (Table 3). The results indicated that the weekly onion in two markets followed a nonlinear pattern.

Table 3. Brock- Dechert-Scheinkman (BDS) test

Series	Embedding dimension				Conclusion
	2		3		
	Statistics	Probability	Statistics	Probability	
Delhi	64.1369	< 0.001	85.5871	< 0.001	Nonlinear
	42.0267	< 0.001	44.6205	< 0.001	
	37.4430	< 0.001	37.0336	< 0.001	
	35.1876	< 0.001	33.7600	< 0.001	
Nashik	61.2138	< 0.001	83.8933	< 0.001	Nonlinear
	37.7240	< 0.001	40.5676	< 0.001	
	32.9430	< 0.001	32.8153	< 0.001	
	30.6707	< 0.001	29.4113	< 0.001	

In the case of non-stationarity of the time series, the problem of estimation of parameters become complex and difficult. Recent days decomposition techniques like EMD, EEMD, and VMD etc. have gained popularity to deal with non-stationary problem in dataset. In the study VMD is used for decomposition of price data series. The predefined parameters of VMD algorithm were moderate bandwidth constraint ($\alpha = 2000$), noise-tolerance ($\tau = 0$), modes/IMFs (k), omega initialization (init = 0) and tolerance of convergence condition (tol = 0). The values of α, τ and ϵ were specified as 2000, 0, and 10^{-7} following Dragomiretskiy and Zosso (2014). The number of IMFs (K) is fixed at 5 as the following IMFs tend to be similar when K>5. The decomposed IMF series are depicted by figure 4. The results showed there was a stable trend in higher frequency IMFs. Further the behaviours of IMFs were checked. They were stationary in nature which is conducive for prediction.

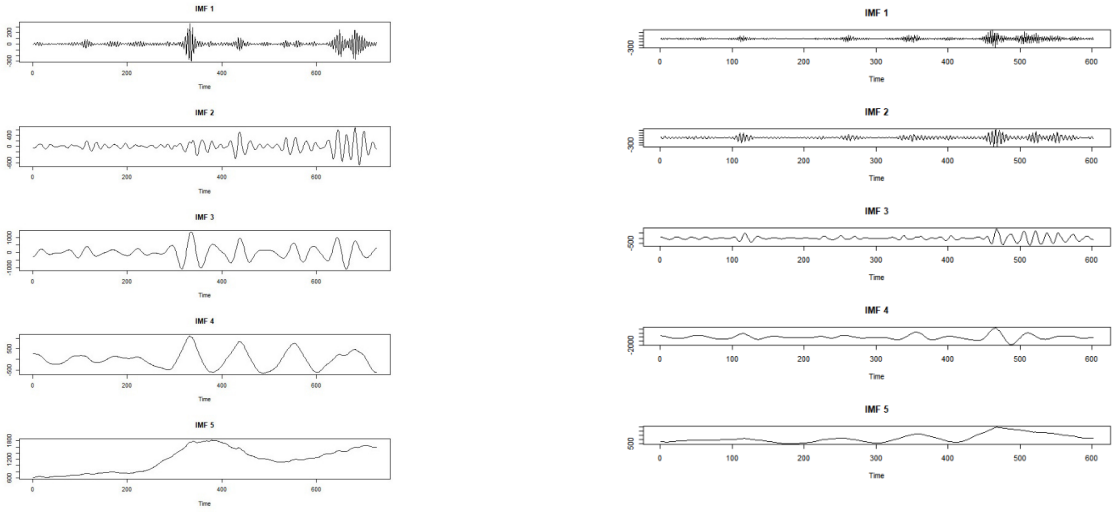
Table 4. GA setting parameters

Algorithm	Parameters	Value/setting
Genetic algorithm	Population size	10, 20, 30, 50, 100
	Crossover's probability	80%
	Mutation's probability	10%
	Type of replacement	Elitist (5% of the population)
	Type of selection	Linear ranking
	Max number of generations/iteration	100

Genetic algorithm was used for optimizing the objective (fitness) function i.e. root mean square error (RMSE) for the hyper-parameters of SVR and RF models. In the present study, the process of GA based parameter optimization involved fitness scaling, selection, crossover and mutation (figure 2). The parameter values were used for GA are mentioned in table 4. The rank fitness scaling method and the linear ranking scheme were adopted for fitness scaling and selection method correspondingly. To investigate the effect of various population sizes, five different values were adopted in this research, i.e. 10, 20, 30, 50 and 100. But the smallest fitness values were achieved at population sizes of 20 and 30 in the Delhi and Nashik markets respectively (Fig. 5). A distance based heuristic crossover method was used with a probability percentage of 10. The process of GA optimization was validated with 10 fold validation for both the SVR and RF models. The optimized values of hyper-parameters were mentioned in Table 5.

Performance of the proposed model

In the present study, VMD method was applied to decompose a nonlinear and non-stationary time series data into 5 IMFs which are further forecasted by GA based optimized SVR and RF models i.e. GA-SVR and GA-RF (Fig. 3). In the process of model forecasting, the model was built and validated on train data (90%) and the test data (10%). Eventually, a final forecast was



(a) **(b)**

Fig. 4. The results of VMD methods (a) Delhi market (b) Nashik market

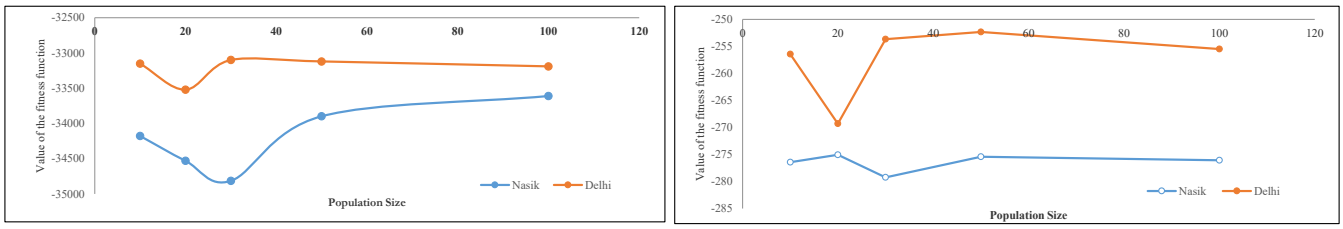
Table 5. Obtained value of hyper-parameters for developed models

GA-SVR model			GA-RF model		
Parameter	**Delhi**	**Nashik**	**Parameter**	**Delhi**	**Nashik**
C	5.2390	8.3079	m_{try}	12	14
ϵ	0.1225	0.2152	N_{tree}	54	150
γ	0.1768	0.0412			

obtained by aggregating all the individual IMF forecast values. The proposed hybrid approach was applied to the weekly prices of onion in Delhi market and Nashik market. Three performance measures i.e. root mean square error (RMSE), mean absolute error (MAE) and mean absolute percentage error (MAPE) (Das *et al.*, 2021) were adopted for the model evaluation. We also have tried to find the effect of the VMD decomposition and GA on model performance. For this purpose, the SVR and RF model was modelled with VMD only i.e. VMD-SVR/RF and without VMD & GA i.e. SVR and RF model.

For ease of understanding the 12 forecasted values of the proposed optimised hybrid ML models were plotted against the actual values in figure 6. The results indicated that the VMD-GA-SVR almost captured actual data points in the both two markets compared to VMD-GA-RF model.

The single SVR and RF models were fitted to check how these ML worked on the unprocessed data set. The results showed that the VMD based GA optimized ML models (VMD-GA-RF and VMD-GA-SVR) performed best followed by the VMD based models (VMD-RF and VMD-SVR). The single models (SVR and RF without VMD and GA) performed worst in both market data sets. But these models with VMD and GA performed the best which confirmed the improvement in prediction accuracy of the model. The GA algorithm helped the model escape from multiple local minima and improved the rate of convergence. In the Delhi dataset, the VMD-GA-SVR model outperformed the other five alternative models significantly, whereas in the Nashik market, the VMD-GA-RF model outperformed. Besides this, the



a. SVR model b. RF model

Fig. 5. Values of fitness function at different population sizes in GA optimization

Table 6. Performance of fitted models

Market	Model		RMSE	MAE	MAPE
Delhi	Training set (90%)	SVR	172.2842	95.3417	0.0819
		RF	176.3515	96.6336	0.0801
		VMD-SVR	162.6429	80.3200	0.0808
		VMD-RF	127.228	79.1102	0.0682
		VMD-GA-SVR	117.9232	61.4339	0.0495
		VMD-GA-RF	125.8092	78.0171	0.0658
	Testing set (10%)	SVR	258.0524	155.9089	0.0861
		RF	271.3185	190.3788	0.1078
		VMD-SVR	195.4842	127.7532	0.0781
		VMD-RF	199.3744	127.8398	0.0814
		VMD-GA-SVR	187.7978	116.7686	0.0711
		VMD-GA-RF	192.2218	125.4795	0.0800
Nashik	Training set (90%)	SVR	286.5878	158.6196	0.1230
		RF	266.8041	150.5873	0.1228
		VMD-SVR	250.3988	146.4175	0.1213
		VMD-RF	245.7336	149.2953	0.1283
		VMD-GA-SVR	201.6371	123.5885	0.1079
		VMD-GA-RF	201.4721	123.4662	0.1079
	Testing set (10%)	SVR	127.8343	99.8199	0.0954
		RF	102.1941	87.8172	0.0852
		VMD-SVR	98.3243	81.1681	0.0760
		VMD-RF	95.8568	74.7327	0.0725
		VMD-GA-SVR	86.6370	71.0800	0.0683
		VMD-GA-RF	85.9966	71.0227	0.0683

percentage of model performance improvement using the following indices

$$P_{RMSE} = \frac{RMSE_1 - RMSE_2}{RMSE_1} \times 100\%$$

$$P_{MAE} = \frac{MAE_1 - MAE_2}{MAE_1} \times 100\%$$

$$P_{MAPE} = \frac{MAPE_1 - MAPE_2}{MAPE_1} \times 100\% \quad (15)$$

where the prediction results of the SVR basic model were directly used as the standard $RMSE_1$, MAE_1, and $MAPE_1$. The $RMSE_2$, MAE_2, and $MAPE_2$ were the values of the comparison model. SVR is one of the known standard models for time series, hence it was used as the baseline model for comparison in the study.

From data analysis from Table 7, compared with the SVR model, the highest improvement rate was

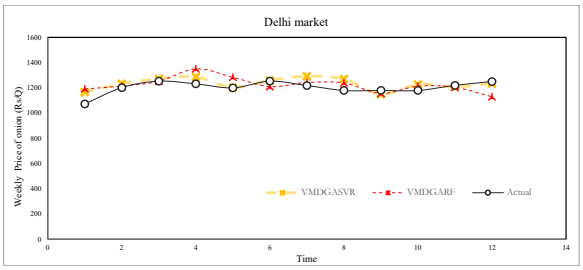


Fig. 6.a. Forecast performance of proposed optimised hybrid ML models in Delhi Market

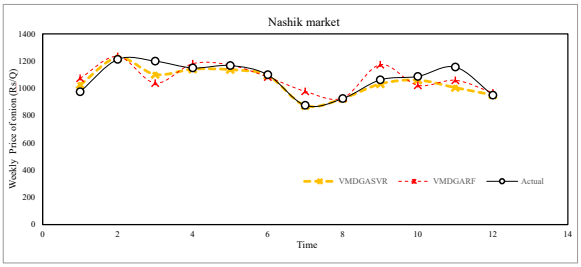


Fig. 6.b. Forecast performance of proposed optimised hybrid ML models in Nashik

achieved in VMD-GA-SVR model and the RF model had the lowest performance rate for Delhi market price. In the Nashik market VMD-GA-RF model and the VMD-GA-SVR produced at par result. In terms of decomposition, the VMD-RF and VMD-SVR models improved at considerably higher rates than the SVR models in both markets. From the perspective of optimization, VMD-GA-SVR and VMD-GA-RF had the higher improvement rate than the other model in both the markets.

The Diebold-Mariano test (Diebold and Mariano, 1995) was then employed to evaluate the fitted models' accuracy. The results of DM test (Table 8) clearly indicated that VMD-GA-SVR had better accuracy than VMD-GA-RF. Similarly, VMD based optimised RF/SVR models outperformed compared to the VMD based RF/SVR. It also find out the accuracy of generic RF/SVR models were lower than VMD based RF/SVR model.

The results confirmed the following points-

1. The embedded superior nonlinear capabilities of the SVR and RF model.
2. The superior decomposition capability of VMD for which each IMF can comprehensively represent its decomposed characteristics. Then the ML models i.e. RF and SVR can separately simulate the decomposed data pattern of each IMF which improves the prediction accuracy of the model.

Table 7. Percentage of model performance improvement

SVR vs model	Model	Training set			Testing set		
		P_{RMSE}	P_{MAE}	P_{MAPE}	P_{RMSE}	P_{MAE}	P_{MAPE}
Delhi	RF	-2.3608	-1.3550	2.1978	-5.1409	-22.1090	-25.2033
	VMD-RF	26.1523	17.0246	16.7277	22.7388	18.0035	5.4588
	VMD-GA-RF	26.9758	18.1711	19.6581	25.5106	19.5174	7.0848
	VMD-SVR	5.5962	15.7557	1.3431	24.2464	18.0591	9.2915
	VMD-GA-SVR	31.5531	35.5645	39.5604	27.2249	25.1046	17.4216
Nashik	RF	6.9031	5.0639	0.1178	20.0573	12.0243	10.6899
	VMD-RF	14.2554	5.8784	0.3602	25.0148	25.1325	23.9270
	VMD-GA-RF	29.6997	22.1621	12.2542	32.7281	28.8492	28.4135
	VMD-SVR	12.6275	7.6927	0.1268	23.0846	18.6855	20.3465
	VMD-GA-SVR	29.6421	22.0850	12.2448	32.2271	28.7922	28.4046

Table 8. Results of DM test

Hypothesis	p value		Remarks
	Delhi	Nashik	
H0: The accuracy of both VMD-GA-SVR and VMD-RF is same. H1: The accuracy of VMD-GA-SVR is superior to VMD-GA-RF.	<0.01	<0.01	The accuracy of VMD-GA-SVR is superior to VMD-GA-RF.
H0: The accuracy of both VMD-GA-SVR and VMD-SVR is same. H1: The accuracy of VMD-GA-SVR is superior to VMD-SVR.	<0.01	<0.01	The accuracy of VMD-GA-SVR is superior to VMD-SVR.
H0: The accuracy of both VMD-GA-RF and VMD-RF is same. H1: The accuracy of VMD-GA-RF is superior VMD-RF.	<0.01	<0.01	The accuracy of VMD-GA-RF is superior VMD-RF.
H0: The accuracy of both VMD-SVR and SVR is same. H1: The accuracy of VMD-SVR is superior to SVR.	<0.01	<0.01	The accuracy of VMD-SVR is superior to SVR.
H0: The accuracy of both VMD-RF and RF is same. H1: The accuracy of VMD-RF is superior RF.	<0.01	<0.01	The accuracy of VMD-RF is superior RF.

3. Efficient optimization of hyper-parameters in SVR and RF models using GA improves the performance of the models. The VMD-GA-RF and VMD-GA-SVR had lower RMSE, MAE and MAPE values than the un-optimised models which confirmed that GA optimization parameters can also improve the prediction accuracy.

4. CONCLUSION

This study has put on concentrated efforts to improve the prediction ability of the machine learning techniques with the help of VMD and GA. The proposed hybrid models are applied to the weekly price data of onions. The VMD decomposed original time series data into several stable IMFs which are free from mode mixing. Then each IMFs was forecasted by applying the GA optimised SVR and RF model. Finally, all the forecasted values of IMF components were aggregated as the final forecast. The performance measures RMSE, MAD and MAPE were employed as accuracy measures for SVR, RF, VMD-SVR, VMD-RF, VMD-GA-SVR and VMD-GA-RF models. The results indicated an improvement in forecasting capabilities of GA optimised machine learning models i.e. VMD-GA-SVR and VMD-GA-RF. Besides, this study suggested that VMD based models with GA (VMD-GA-SVR and VMD-GA-RF) or without GA (VMD-SVR and VMD-RF) compared to the generic SVR and RF model are superior. An R package VMDML has been developed for decomposition based ML models. The findings also indicated that VMD can be an alternative method of time series decomposition. The proposed approach can be applied to a variety of agricultural price series and we strongly believe that it will be add on to the rich literature of machine learning models.

ACKNOWLEDGEMENTS

The authors are thankful to the Director, ICAR-IASRI and PG school IARI for providing facilities for carrying out the present research. The authors are also thankful to the UGC, India providing the fellowship for the research.

REFERENCES

Antwi A. and Henneberry J. (1995). Developers, non-linearity and asymmetry in the development cycle. *J of Prop Res.*, **12**(3), 217-39.

Balke N. S. and Fomby T. B. (1994). Large shocks, small shocks, and economic fluctuations: outliers in macroeconomic time series. *J Appl Econ.*, **9**(2), 181-200.

Bishop M. C. (1995). *Neural Networks for Pattern Recognition*. Oxford University Press, New York.

Breiman L. (2001). Random forests. *Mach. Learn.*, **45**(1), 5-32.

Brock W. A., Scheinkman J. A., Dechert W. D. and LeBaron B. (1996). A test for independence based on the correlation dimension. *Econom Rev*., **15**, 197-235.

Choudhury K., Jha G. K., Das P. and Chaturvedi K. K. (2019). Forecasting potato price using ensemble artificial neural networks. *Indian J. Ext. Educ.*, **55(1)**, 71-77.

Cong L. N. and Meesad P. (2013). Meta-heuristic algorithms applied to the optimization of type-1 and type 2 TSK fuzzy logic systems for sea water level prediction. In: *2013 IEEE 6th International Workshop on Computational Intelligence and Applications*. IEEE New York, pp. 69-74.

Das P., Jha G. K., Ahmad T., Lama A. and Mouselimis L. (2022). VMDML: Variational Mode Decomposition Based Machine Learning Models. https://CRAN.R-project.org/package=VMDML.

Das P., Jha G. K., Lama A., Parsad R. and Mishra D. (2020). Empirical mode decomposition based support vector regression for agricultural price forecasting. *Indian J. Ext. Educ.*, **56**(2), 7-12.

Das P., Lama A. and Jha G. K. (2021). An Improved Cointegration based Time Delay Neural Network Model for Price Forecasting. *J. of the Indian Society of Agri. Stat.*, **75**(3), 187-192.

De G. J. G. and Kumar K. (1992). Some recent developments in non-linear time series modelling, testing, and forecasting. *Int. J. Forecast*., **8**(2), 135-56.

Dickey D. A. and Fuller W. A. (1979). Distribution of the estimators for autoregressive time series with a unit root. *J Am Stat Assoc*., **84**, 427-431.

Diebold F. X. and Mariano R. (1995). Comparing predictive accuracy. *J Bus Econ Stat.*, **13**, 253-265

Dragomiretskiy K. and Zosso D. (2014). Variational mode decomposition*. IEEE Trans. Signal Process*, **62**, 531–544.

Fadlallah S. O., Anderson T. N. and Nates R. J. (2021). Artificial Neural Network–Particle Swarm Optimization (ANN-PSO) Approach for Behaviour Prediction and Structural Optimization of Lightweight Sandwich Composite Heliostats. *Arab J Sci Eng.,* **46**, 12721-12742.

Ghimire S., Deo R. C., Raj N. and Mi J. (2019). Wavelet-based 3-phase hybrid SVR model trained with satellite-derived predictors, particle swarm optimization and maximum overlap discrete wavelet transform for solar radiation prediction. *Renew. Sust. Energ. Rev.*, **113**, 109247.

Giles C. L., Lawrence S. and Tsoi A. C. (2001). Noisy time series prediction using recurrent neural networks and grammatical inference. *Mach. Learn.,* **44**(1), 161-83.

Ho T. K. (1998). The random subspace method for constructing decision forests. *IEEE Trans. Pattern Anal. Mach. Intell.*, **20**(8), 832-44.

Huang N. E., Shen Z., Long S. R., Wu M. C., Shih H. H., Zheng Q., Yen N. C., Tung C. C. and Liu H. H. (1971). The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis. *Proc. R. Soc. A*, **454**, 903-95.

Jarque C. M. and Bera A. K. (1987). A test for normality of observations and regression residuals. *Int Stat Rev*., **55**(2), 163-172.

Jha G. K. and Sinha K. (2014). Time-delay neural networks for time series prediction: an application to the monthly wholesale price of oilseeds in India. *Neural Comput Appl*., **3-4**, 563-571.

Kane M. J., Price N., Scotch M. and Rabinowitz P. (2014). Comparison of ARIMA and Random Forest time series models for prediction of avian influenza H5N1 outbreaks. *BMC Bioinform.,* **15**, 276.

Karasu S., Altan A., Bekiros S. and Ahmad W. (2020). A new forecasting model with wrapper-based feature selection approach using multi-objective optimization technique for chaotic crude oil time series. *Energy*, **212**, 118750.

Khandelwal I., Adhikari R. and Verma G. (2015). Time series forecasting using hybrid ARIMA and ANN models based on DWT decomposition. *Procedia Comput. Sci.*, **48**, 173-179.

Kisi O., Shiri J., Karimi S., Shamshirband S., Motamedi S., Petkovi D. and Hashim R. (2015). A survey of water level fluctuation predicting in Urmia Lake using support vector machine with firefly algorithm. *Appl. Math. and Comp.*, **270**(C), 731–743.

Kumar C. S. (2021). Hybrid models for intraday stock price forecasting based on artificial neural networks and metaheuristic algorithms. *Pattern Recognit. Lett.*, **147**, 124-133.

Lahmiri S. (2016). A variational mode decomposition approach for analysis and forecasting of economic and financial time series. *Expert Syst. Appl.,* **55**, 268–273.

Liu Z., Zhou P., Chen G. and Guo L. (2014). Evaluating a coupled discrete wavelet transform and support vector regression for daily and monthly stream flow forecasting. *Hydrol.,* **519**, 2822–2831.

Meng Y., Qasem S. N. and Shokri M. (2020). Dimension reduction of machine learning-based forecasting models employing principal component analysis. *Math.*, **8**(8), 1233.

Peng H. W., Wu S. F., Wei C. C. and Lee S. J. (2015). Time series forecasting with a neuro-fuzzy modeling scheme. *Appl. Soft Comput.*, **32**, 481-93.

Phillips P. C. B. and Perron P. (1988). Testing for a Unit Root in Time Series Regression. *Biometrika*., **75**(2), 335-346.

R Core Team (2018). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. (https://www.R-project.org/.)

Seo Y., Kim S. and Singh V. P. (2018). Machine Learning Models Coupled with Variational Mode Decomposition: A New Approach for Modeling Daily Rainfall-Runoff. *Atmosphere*, **9**, 251.

Shi P. and Yang W. (2017). Precise feature extraction from wind turbine condition monitoring signals by using optimizedvariational mode decomposition. *IET Renew. Power Gener*, **11**, 245–252.

Sivanandam S. N. and Deepa S. N. (2007). *Introduction to Genetic Algorithms*. Springer, Berlin.

Sun G., Chen T., Wei Z., Sun Y., Zang H. and Chen S. (2016). A carbon price forecasting model based on variational mode decomposition and spiking neural networks. *Energies*, **9**, 54.

Vapnik V. (1995). *The Nature of Statistical Learning Theory*. Springer, New York. https://doi.org/10.1007/978-1-4757-3264-1.

Vapnik N. V. *Statistical Learning Theory*. Wiley-Interscience. 1st ed., 1998.

Wang L. L., Li X., Ran Y. H. and Guo Y. L. (2020). Monthly runoff prediction of Zhengyixia in the Heihe river based on singular spectrum analysis-grey wolf optimizer-support vector regression hybrid model. *Remote Sens. Tech and App*., **35**(2), 355–364.

Wu Z. and Huang H. E. (2009). Ensemble empirical mode decomposition: A noise-assisted data analysis method. *Adv. Adapt. Data Anal.,* **1**, 1–41.

Yao Z., Ji-Ping X. U., Kong J. L. and Liu S. B. (2018). Prediction of river water level by GA-ELMAN model. *Journal of Yangtze River Scientific Research Institute*, 35 **(9)**, 34–37.

Zhang Y., Gao X. and Katayama S. (2015). Weld appearance prediction with BP neural network improved by genetic algorithm during disk laser welding. *J. Manuf. Syst.*, **34**, 53-9.