frontiers | Frontiers in Applied Mathematics and Statistics

TYPE Original Research
PUBLISHED 11 December 2024
DOI 10.3389/fams.2024.1435517

OPEN ACCESS

EDITED BY
Talat Nazir,
University of South Africa, South Africa

REVIEWED BY
Alladoumbaye Ngueilbaye,
Shenzhen University, China
Ceylan Yozgatligil,
Middle East Technical University, Türkiye

*CORRESPONDENCE
Ranjit Kumar Paul
✉ ranjitstat@gmail.com

RECEIVED 20 May 2024
ACCEPTED 22 November 2024
PUBLISHED 11 December 2024

CITATION
Singh AK, Yeasin M, Paul RK, Paul AK and Sarkar A (2024) Dynamic ensemble-based machine learning models for predicting pest populations.
Front. Appl. Math. Stat. 10:1435517.
doi: 10.3389/fams.2024.1435517

Dynamic ensemble-based machine learning models for predicting pest populations

Ankit Kumar Singh[1], Md Yeasin[2], Ranjit Kumar Paul[2]*, A. K. Paul[2] and Anita Sarkar[1]

[1]The Graduate School, ICAR-Indian Agricultural Research Institute, New Delhi, India, [2]ICAR-Indian Agricultural Statistics Research Institute, New Delhi, India

Early prediction of pest occurrences can enhance crop production, reduce input costs, and minimize environmental damage. Advances in machine learning algorithms facilitate the development of efficient pest alert systems. Furthermore, ensemble algorithms help in the utilization of several models rather than being dependent on a single model. This study introduces a dynamic ensemble model with absolute log error (ALE) and logistic error functions using four machine learning models—artificial neural networks (ANNs), support vector regression (SVR), k-nearest neighbors (kNN), and random forests (RF). Various abiotic factors such as minimum and maximum temperature, rainfall, and morning and evening relative humidity were incorporated into the model as exogenous variables. The proposed algorithms were compared with fixed-weighted and unweighted ensemble methods, and candidate machine learning models, using the pest population data for yellow stem borer (YSB) from two regions of India. Error metrics include the root mean square log error (RMSLE), root relative square error (RRSE), and median absolute error (MDAE), along with the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) algorithm. This study concluded that the proposed dynamic ensemble algorithm demonstrated better predictive accuracy in forecasting YSB infestation in rice crops.

KEYWORDS

accuracy, dynamic ensemble, machine learning, pest population, yellow stem borer

1 Introduction

Agricultural pests in India pose a complex and significant threat to the country's vital agricultural sector, which is the backbone of its economy and food supply. Improved crop production can be achieved by safeguarding crops from pests and increasing public awareness about the damage that pests can cause (1). Pests can cause extensive harm to crops, transmit dangerous plant diseases, and undermine crop productivity. In addition, temperature alterations impact the growth rates of various insect species. As global temperatures rise, agricultural losses due to insect pests are expected to increase by 10 to 25% for each degree of warming (2). Farmers have traditionally relied on pesticides and chemical treatments to get rid of pests. However, the widespread use of chemical pesticides also raises environmental and health concerns (3). Effectively addressing these agricultural pest challenges in India necessitates a comprehensive strategy that includes integrated pest management, sustainable farming practices, and increased investment in research and education, all of which are crucial for safeguarding crops and ensuring food security for the country's ever-growing population. As agricultural practices shift toward advanced pest management, there is a growing need for innovative tools to reduce the uncertainties involved in estimating the levels of insect pest infestations in crop fields (4). To effectively address this challenge, integrated pest management strategies are essential. Regular monitoring, early detection, and appropriate control measures are crucial to minimize the damage caused by pests and ensure healthy, productive crop production.

Forecasting pest populations holds significant importance in agriculture and pest management for various compelling reasons. Early detection of potential pest outbreaks allows for proactive measures to safeguard crops, ultimately preventing substantial economic losses and ensuring a consistent food supply. In addition, it promotes sustainable agricultural practices by reducing reliance on chemical pesticides through accurate forecasting. This approach encourages the use of eco-friendly practices such as biological control and integrated pest management, which are beneficial to soil health and the overall ecosystem. Pest population forecasting serves as an indispensable tool for regulatory agencies and policymakers. It facilitates efficient resource allocation, enables quick responses to emerging pest threats, and aids in the development of policies and regulations that ensure agricultural productivity and food security.

2 Related work

Many researchers have proposed models for predicting pest populations and have compared them with other popular forecasting models. Li et al. (5) proposed a forecasting model for the vegetable pest flea beetle using the maximum likelihood algorithm. To develop effective pest management strategies for specific agricultural ecosystems, it is essential to gather information on the abundance and distribution of pests in relation to weather conditions (6). Arya et al. (7) used the autoregressive integrated moving average with exogenous variables (ARIMAX) model to predict pest populations using weather covariates in India. Paul et al. (8) applied statistical models to predict sterility mosaic disease in pigeon pea in Gujarat, India. Liu et al. (9) developed a monitoring and forecasting system for pest control based on a deep learning algorithm. Anwar et al. (1) developed a mathematical model to promote farmer awareness in crop pest management. Neta et al. (10) developed a temperature-dependent population dynamics model for the global insect pest *Bemisia tabaci* and verified its accuracy under field conditions.

Some recent studies have incorporated machine learning (ML) techniques to predict pest populations. Lankin-Vega et al. (11) proposed an ensemble model using artificial neural networks (ANNs) to predict the population of *Rhopalosiphum padi* (L.). Yan et al. (12) employed multiple regression (MR) and ANN modeling approaches to predict pest risks in crops. Paul et al. (13) predicted early blight severity in tomatoes using machine learning techniques. Skawsang et al. (14) applied ANN, random forests (RF), and MR analyses to predict the brown planthopper population using abiotic factors and host-plant phenology during the dry farming season from 2006 to 2016 in the Central Plains of Thailand. Paul et al. (15) developed an ML-based hybrid model for forecasting sterility mosaic disease in pigeon pea. Marković et al. (16) proposed an ML model to forecast the daily appearance of insects during a season, considering abiotic factors such as air temperature and relative humidity. Ibrahim et al. (17) applied a fuzzy neural network (FNN) model to predict the population of fruit flies in avocado crops. Paul et al. (18) developed a wavelet-based ML model for predicting the occurrence of spiders in pigeon pea. Sidumo et al. (19) suggested that the support vector machine (SVM) performs better than other machine learning models, such as RF, k-nearest neighbors (kNN), and ANN, in reducing over dispersion in count data. Predictions from ensemble methods, which are generated by combining multiple models, have gained popularity as they tend to provide more efficient forecasts compared to individual models. Galicia et al. (20) proposed ensemble learning for predicting time-series data and investigated two strategies for updating the weights: dynamic and static ensembles. The dynamic ensemble emerged as the better-performing model. Sharma et al. (21) proposed a bagging-based ensemble model for forecasting the number of incidences of disease based on past records. Abdelhamid and Alotaibi (22) proposed a two-level ensemble model, where the first level includes RF, support vector regression (SVR), and light gradient boosting algorithms, and the second level is based on the elastic net regression model. Du et al. (23) proposed a dynamic ensemble model based on Bayesian optimization, which provides a forecast combination for a time series with time-varying underlying patterns. Gangula et al. (24) implemented ensemble ML techniques within hybrid integrations to identify factors linked to the transmission of dengue fever, leading to enhanced performance outcomes. Sun et al. (25) and Jiang et al. (26) are some of the recent studies based on ensemble algorithms in different fields of time series forecasting. Dynamic ensemble methods involve the practice of allocating varying weights to candidate models within the ensemble. In such a scheme, the weights assigned to the base models can fluctuate either over time or according to specific conditions, overcoming the limitation of a fixed-weighted ensemble method. In fixed-weighted ensemble methods, the weights assigned to different candidate models are distinct but remain fixed.

The main objective of the present study is to utilize a time-dependent weighting scheme to ensemble forecasts obtained from machine learning models, such as ANN, SVR, kNN, and RF, using information available on exogenous weather factors for the prediction of pest populations. The weekly population of the yellow stem borer (YSB) pest, along with climatic factors affecting rice crops in Rajendranagar, Hyderabad, and Marteru, Andhra Pradesh, India, has been analyzed. The predictive accuracies of this approach were evaluated against candidate benchmark models, the unweighted ensemble method, and the weighted ensemble method, to forecast the data under consideration.

3 Methods

This study predicted pest populations based on historical data and abiotic factors. Pest population data are generally count data that exhibit over dispersion characteristics. In this section, popular machine learning models used as candidate models in the ensemble approach are briefly discussed.

3.1 Candidate model

3.1.1 Artificial neural network (ANN)

An artificial neural network (ANN) is a mathematical model inspired by the human brain, designed to tackle complex problems in scientific research and engineering applications. The fundamental building block of an ANN is the neuron, and these neurons are structured into three different layers: the input layer, the hidden layer, and the output layer. The input layer represents the input parameters, the output layer corresponds to the predicted outcomes, and the hidden layer serves as an artificial layer that captures non-linear relationships. In this study, we considered an ANN with an exogenous variable model, i.e., the ANN-X model (Figure 1), which includes a time series sequence from y_{t-1} to y_{t-n}, with n, the number of lags, and abiotic factors, such as maximum temperature (Tmax), minimum temperature (Tmin), relative humidity-morning (RHM), relative humidity-evening (RHE), and rainfall (RF), as inputs.

To model a time series using an ANN with an exogenous variable, a non-linear function (f) is created, that operates on a sequence of values (y_t) from y_{t-1} to y_{t-n}, where n is the number of lags and x_1, x_2, … x_q denote the abiotic factors.

$$y_t = w_0 + \sum_{j=1}^{h} w_j f\left(w_{0j} + \sum_{i=1}^{n} w_{ij} y_{t-i} + \sum_{k=1}^{q} g_{kj} x_k \right) + e_t \quad (1)$$

In Equation 1, w_{ij}, g_{kj}, and w_j represent the weights, while h, n, and e_t represent the number of hidden nodes, the number of input nodes, and the error term, respectively. The activation function for the hidden layers in an ANN model can take various forms, such as sigmoid, RLU, and tanh. The application of an ANN is well-known in disease and epidemic prediction. Niazkar and Niazkar (35) used an ANN for the prediction of the COVID-19 outbreak.

3.1.2 K-nearest neighbors (kNN)

The kNN algorithm is a non-parametric technique that retains all available data points and makes predictions for numerical targets by assessing similarity, often using distance metrics. In kNN regression, a basic implementation involves computing the mean of the numerical targets from the k nearest neighbors. Alternatively, another approach utilizes a weighted average based on the inverse distance of kNN. kNN regression applies distance functions such as Euclidean, Manhattan, and Minkowski. For example, de Oliveira Aparecido et al. (27) used kNN to predict the incidence of pests in *Coffea arabica*.

The kNN model is given in Equation 2:

$$Output_{kNN} = f\left(u\right) \quad (2)$$

where u contains exogenous variables (abiotic factors) and a sequence of a time series from y_{t-1} to y_{t-n}, and n is the number of lags.

3.1.3 Support vector regression (SVR)

SVR can effectively address non-linear associations between input variables and the target variable by applying a kernel function to relocate the data into a higher-dimensional space. This feature enhances its suitability for regression tasks involving intricate relationships between input variables and the target variable.

The SVR-X model can be formulated as follows:

$$f\left(u\right) = z\varnothing\left(u\right) + b \quad (3)$$

where u contains abiotic factors and a sequence of a time series from y_{t-1} to y_{t-n}, and n is the number of lags. The estimated function of the dataset, i.e., $f\left(u\right)$ in Equation 3, is the output of the model. The process involves applying a kernel function $\varnothing\left(\cdot\right)$ to relocate non-linear data to a higher-dimensional feature space, treating this transformed data as if it were linear, and then computing a dot product with a weight vector, represented as 'z.' This dot product is combined with a bias term 'b' to produce the final estimation. Figure 2 shows the architecture of SVR-X with abiotic factors and lags as inputs.

Thus, the SVR model can be formulated as follows:

$$f\left(u,z\right) = f\left(u,\alpha,\alpha^*\right) = \sum_i \left(\alpha_i - \alpha_i^*\right)\varnothing\left(u,u_i\right) + b \quad (4)$$

where, $\varnothing\left(\cdot\right)$ is the kernel function. The RBF kernel is the most commonly used kernel in SVR model building (28–30). The RBF kernel is presented in Equation 5.

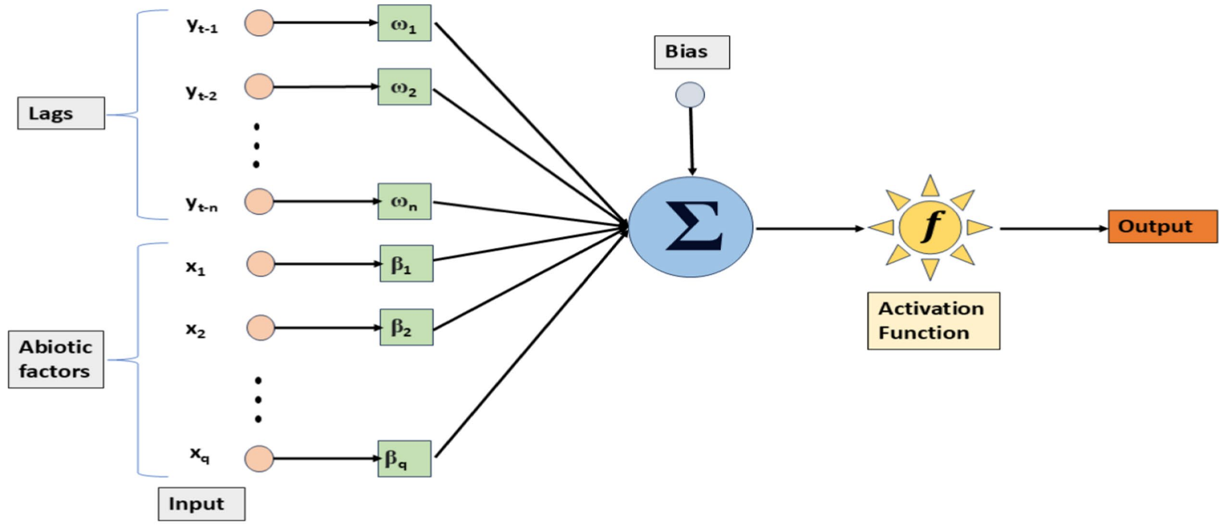

FIGURE 1
Architecture of the ANN-X model, with the lag value and abiotic factors as inputs.

$$\varnothing(x,x_i) = \exp\left(-\frac{\|x - x_i\|^2}{2\sigma^2}\right) \tag{5}$$

The hyper-parameters used in the model are tuned using the Lagrange method. The terms a, a^* mentioned in Equation 4 denote the Lagrange multipliers and should satisfy the following equality: $a_j a_j^* = 0$.

3.1.4 Random forests (RF)

RF (31) is an ensemble learning method based on decision trees. Since its introduction, it has become one of the most popular algorithms in machine learning. Its widespread adoption is attributed to its ability to perform well in diverse applications, even in high-dimensional settings. It is recognized for its computational efficiency and ease of tuning. In classification, a decision tree serves as a straightforward model, where internal nodes represent attribute tests, branches indicate test outcomes, and leaves contain class labels. Decision trees can also handle regression tasks when the target variable is continuous. RF employs a specific technique called bootstrap aggregation, or bagging, where each decision tree is trained on a randomly chosen subsample of the complete training dataset. Successful implementations can be found in various areas, including the prediction of time series data (32).

3.2 Ensemble methods for forecasting

Ensemble methods are a popular approach in forecasting, involving a combination of predictions from multiple individual models to improve the overall forecast accuracy and robustness. These methods leverage the idea that different models can capture different aspects of the underlying data patterns and, when combined, can provide more accurate and reliable forecasts. The predictions from multiple individual models are combined using weights.

The unweighted ensemble approach assigns equal weight to each candidate model. Here, suppose $\hat{y}_1, \hat{y}_2, \ldots \hat{y}_N$ are the forecasted values obtained from N number of models. Then, the forecast from the unweighted ensemble method ($\hat{y}_{uw}$) is given in Equation 6.

$$\hat{y}_{uw} = \frac{1}{N}\left(\sum_{i=1}^{N} \hat{y}_i\right) \tag{6}$$

In the weighted ensemble approach, the weights of the candidate models vary but can be determined using an optimization algorithm. Therefore, the forecast from the weighted ensemble method ($\hat{y}_{fw}$) is given in Equation 7.

$$\hat{y}_{fw} = \left(\sum_{i=1}^{N} w_i\, \hat{y}_i\right) \tag{7}$$

where w_i is the weight assigned to the ith candidate model such that $\sum_{i=1}^{N} w_i = 1$. The weights are optimized using population-based optimization algorithms, such as particle swarm optimization (PSO). Population-based optimization algorithms, such as PSO, offer the benefit of concurrently exploring a wide range of potential solutions,

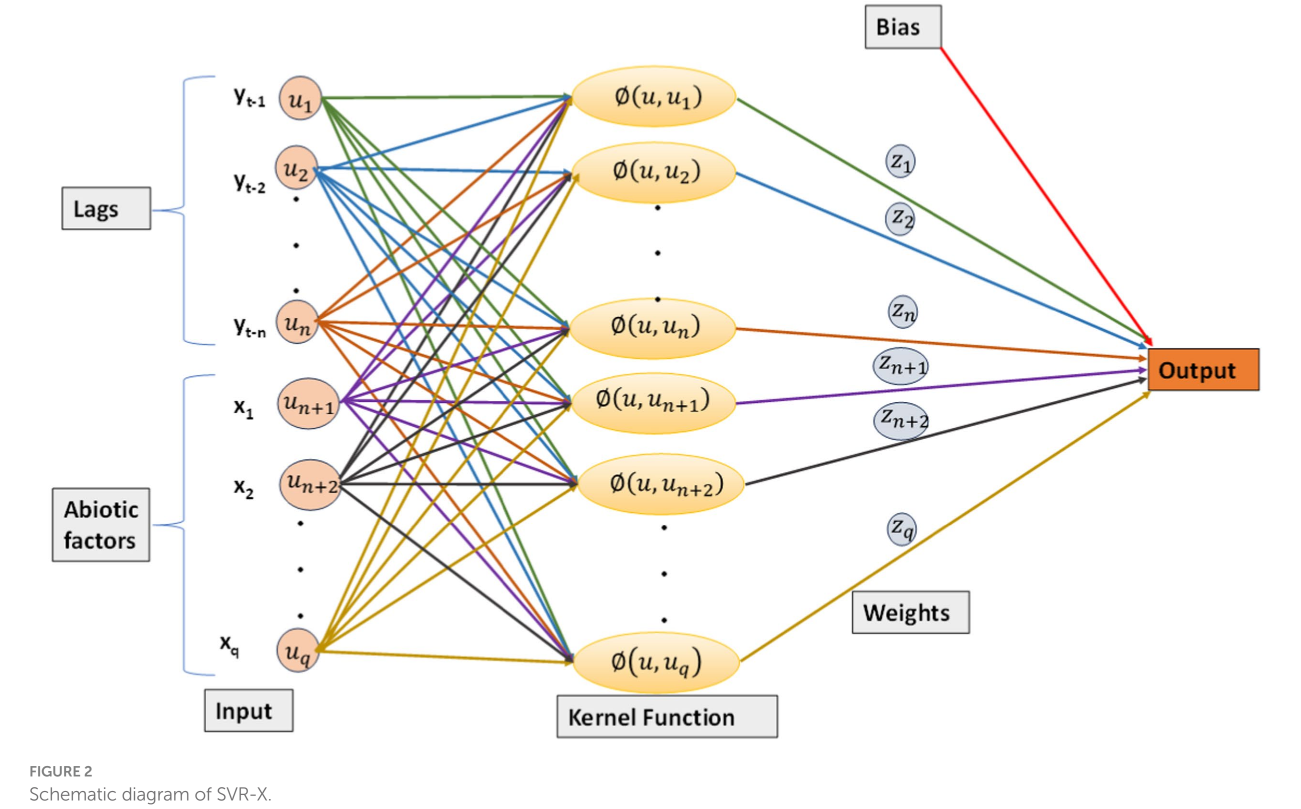

FIGURE 2
Schematic diagram of SVR-X.

enhancing the chances of discovering the global optimum while preventing entrapment in local optima (33).

A dynamic weighting scheme in an ensemble method involves the practice of allocating varying weights to candidate models. These weight adjustments are made in response to model performance or other factors. In such a scheme, the weights assigned to the base models can fluctuate either over time or based on specific conditions.

3.2.1 Time-dependent dynamic ensemble model

Deb and Deb (34) proposed the use of a time-dependent weighting scheme based on an error function for generating predictions using an ensemble approach. In this study, the forecasts obtained from the different machine learning techniques were ensembled using the time-dependent weighting scheme (Figure 3). One of the advantages of using a time-dependent dynamic weighting scheme is that the weights are adjusted at each point of time. Let $\hat{y}_t^{(1)}$, $\hat{y}_t^{(2)}$ … $\hat{y}_t^{(j)}$ represent the values of the output obtained by model (1), model (2) … model (j), i.e., j candidate models at time t. As the dataset is divided into a training set (A) and a testing set (B), $\hat{y}_t^{(1)}$, $\hat{y}_t^{(2)}$ … $\hat{y}_t^{(j)}$ are the fitted values of the j candidate models at time t, where t∊ A, and A represent the training set.

$\hat{y}_{t'}^{(1)}$, $\hat{y}_{t'}^{(2)}$ … $\hat{y}_{t'}^{(j)}$ represent the predicted values from the j candidate models at time t', where t$'$ ∊ B, and B represent the prediction set. $w_t^{(i)}$ is the forecasted weight obtained for the ith candidate model at time t such that $\sum_{i=1}^{j} w_t^{(i)} = 1$.

The weighted ensemble approach to predict $y_{t'}$, t$'$ ∊ B is given in Equation 8.

$$\hat{y}_{t'} = \sum_{i=1}^{j} w_{t'}^{(i)}\, \hat{y}_{t'}^{(i)} \tag{8}$$

The weights of the time-dependent weighting scheme are determined by

$$w_t^{(i)} = \frac{1 \big/ E_t^{(i)}}{\left(\sum_{i=1}^{j} 1 \big/ E_t^{(i)}\right)} \tag{9}$$

In Equation 9, $E_t^{(i)}$ is the error obtained for the ith model at time t, such that $\sum_{i=1}^{j} w_t^{(i)} = 1$.

Therefore, at each point of time in the training set, the weights assigned to each method are determined based on an inverse relationship with the error function. The logistic error function and the absolute log error (ALE) function are used to obtain the weights.

The expression for the logistic error function is given in Equation 10.

$$E_t^{(i)} = \frac{1}{\left(1+\exp(-abs\left(\hat{y}_t^{(i)} - y_t\right))\right)} \tag{10}$$

The expression for the absolute log error function is given in Equation 11.

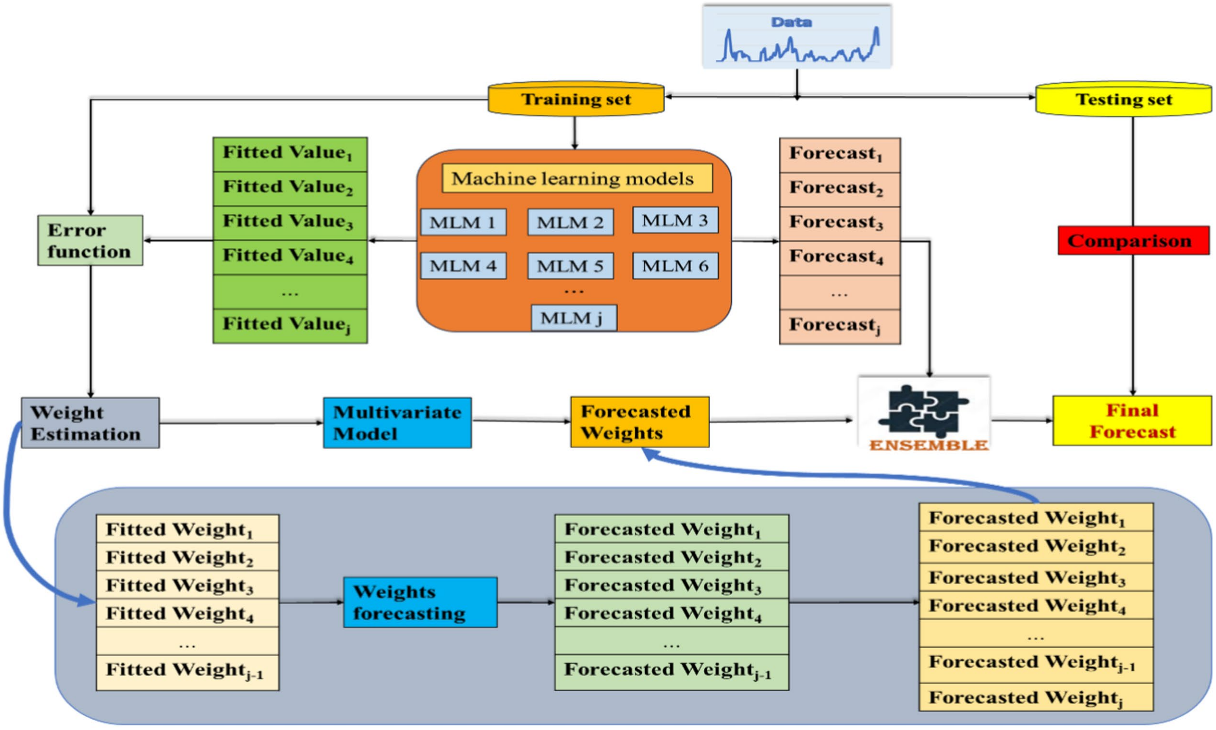

FIGURE 3
Flowchart of the time-dependent weighting scheme ensemble forecasting approach.

$$E_t^{(i)} = \left|\log\left(\hat{y}_t^{(i)}+1\right)-\log\left(y_t+1\right)\right| \qquad (11)$$

The matrix of the weights $\boldsymbol{W}$ obtained for t ∊ A can be treated as a multivariate time series since the sum of all weights is equal to 1, which shows that the weight components are interdependent among each other.

Where, $$\boldsymbol{W} = \left(w^{(1)}, w^{(2)}, \ldots w^{(j)}\right) = \begin{bmatrix} w_1^{(1)} & w_1^{(2)} & \cdots & w_1^{(j)} \\ w_2^{(1)} & w_2^{(2)} & & w_2^{(j)} \\ \vdots & & \ddots & \vdots \\ w_n^{(1)} & w_n^{(2)} & \cdots & w_n^{(j)} \end{bmatrix}.$$

A matrix with independent columns $\boldsymbol{W}^* = \left(w^{(1)}, w^{(2)}, \ldots w^{(j-1)}\right)$ is constructed from $\boldsymbol{W}$ by deleting the column corresponding to the j$^{\text{th}}$ model to obtain an estimate of the weight. For t$'$ ∊ B, the weights are forecasted by fitting a vector autoregressive (VAR) model to the multivariate time series $\boldsymbol{W}^*$. The forecasts of the weights obtained from the VAR model, corresponding to (j-1) number of models, are arranged in the matrix $\widehat{\boldsymbol{W}}^*$ so that the ith column represents the forecasted weights corresponding to the ith candidate model. Thus, the weight matrix $\widehat{\boldsymbol{W}}^*$ for the h-step ahead forecast obtained from the VAR model is defined as

$$\widehat{\boldsymbol{W}}^* = \left(\hat{w}^{(1)}, \hat{w}^{(2)}, \ldots \hat{w}^{(\text{j}-1)}\right) = \begin{bmatrix} \hat{w}_{\text{n}+1}^{(1)} & \hat{w}_{\text{n}+1}^{(2)} & \cdots & \hat{w}_{\text{n}+1}^{(j-1)} \\ \hat{w}_{\text{n}+2}^{(1)} & \hat{w}_{\text{n}+2}^{(2)} & & \hat{w}_{n+2}^{(j)} \\ \vdots & & \ddots & \vdots \\ \hat{w}_{n+h}^{(1)} & w_{n+h}^{(2)} & \cdots & \hat{w}_{n+h}^{(j)} \end{bmatrix}$$

The column of the forecasted weights corresponding to the j$^{\text{th}}$ model is defined as $\hat{w}^{(j)} = \left\{1-\sum_{i=1}^{j-1}\hat{w}^{(i)}\right\}$. The weight matrix $\widehat{\boldsymbol{W}}$, containing the weights corresponding to all j candidate models, is obtained by augmenting $\widehat{\boldsymbol{W}}^*$ with $\hat{w}^{(j)}$ as $\widehat{\boldsymbol{W}} = \left(\hat{w}^{(1)}, \hat{w}^{(2)}, \ldots \hat{w}^{(j-1)}, \hat{w}^{(j)}\right)'$. The kth row of the matrix $\widehat{\boldsymbol{W}}$ contains the forecasted weights corresponding to the k$^{\text{th}}$ candidate model.

Consider $\widehat{\boldsymbol{Y}} = \left(\hat{y}^{(1)}, \hat{y}^{(2)}, \ldots \hat{y}^{(j)}\right)$, where $\hat{y}^{(i)}$ is the vector of the forecasted value obtained from the ith candidate model. Then, the dynamic weighted ensemble forecast $\hat{y}_{dw}$ is given by Equation 12, which is obtained from the multiplication of the matrix $\widehat{\boldsymbol{W}}$ and $\widehat{\boldsymbol{Y}}$. Therefore, the dynamic forecast can be computed using Equation 12.

$$\hat{y}_{dw} = \widehat{\boldsymbol{W}} * \widehat{\boldsymbol{Y}} \qquad (12)$$

3.3 Performance measures

Accuracy metrics, such as the root mean square log error (RMSLE), root relative square error (RRSE), and median absolute error (MDAE), were used to compare the performance of the proposed dynamic ensemble model with the candidate models, the unweighted ensemble model, and the fixed-weighted ensemble model. The values of the RMSLE, RRSE, and MDAE for the ith model are given by the expressions mentioned in Equations 13–15, respectively.

Root Mean Squared Logarithmic Error (RMSLE)

$$RMSLE = \sqrt{\frac{1}{N}\sum_{t=1}^{N}\left(\log\left(\hat{y}_t^{(i)}+1\right)-\log\left(y_t+1\right)\right)^2} \qquad (13)$$

Root Relative Square Error (RRSE).

$$RRSE = \sqrt{\sum\nolimits_{t=1}^{N}\left(y_t-\hat{y}_t^{(i)}\right)^2 / \sum\nolimits_{t=1}^{N}\left(y_t-\bar{y}\right)^2} \qquad (14)$$

Median Absolute Error (MDAE).

$$MDAE = median\left(\left|y_1-\hat{y}_1^{(i)}\right|, \left|y_2-\hat{y}_2^{(i)}\right|, \ldots \left|y_N-\hat{y}_N^{(i)}\right|\right) \qquad (15)$$

where y_t is the actual value at time t, and $\hat{y}_t^{(i)}$ is the forecasted value from the ith model at time t. $\bar{y}$ is the mean of the actual value, and N is the total number of observations.

4 Results and discussion

4.1 Data description and preprocessing

The weekly population (Y) of the yellow stem borer (YSB), *Scirpophaga incertulas*, in rice, along with climatic factors, including maximum temperature (Tmax), minimum temperature (Tmin), relative humidity-morning (RHM), relative humidity-evening (RHE), and rainfall (RF) in Rajendranagar, Hyderabad, and Marteru, Andhra Pradesh, India, were obtained from the Croppest DSS portal of the Central Research Institute for Dryland Agriculture, India.[1] The data ranged from 2007 to 2011 for Rajendranagar and from 2000 to 2011 for Marteru. Each year comprised 52 standard meteorological weeks (SMWs). The rationale behind the selection of the YSB pest was that it is a major pest of rice and has a widespread, devastating impact on the income of rice farmers. This pest affects rice plants at multiple growth stages, particularly during the vegetative and reproductive phases, causing significant yield losses. The descriptive statistics, kernel density, and box plots of the data under consideration are presented in Tables 1, 2 and Figure 4, respectively.

From Tables 1, 2, it can be observed that the mean of the pest populations exceeded the variance, indicating a characteristic of a

[1] http://www.icar-crida.res.in:8080/naip/index.jsp

TABLE 1 Descriptive statistics of the pest population and climatic data from Rajendranagar.

Statistic	Pest population	Tmax (°C)	Tmin (°C)	RHM (%)	RHE (%)	RF (mm)
Mean	87.654	32.559	20.169	79.776	46.952	15.435
SD	114.691	3.777	4.558	11.356	18.497	28.918
Median	44.000	31.450	21.750	82.700	42.400	0.850
Minimum	0.000	26.300	7.000	30.900	15.600	0.000
Maximum	641.000	41.400	27.900	95.400	89.100	188.800
Skewness	2.155	0.690	−0.569	−1.299	0.329	2.651
Kurtosis	5.188	−0.616	−0.645	1.852	−1.063	8.212
SE (Mean)	7.113	0.234	0.283	0.704	1.147	1.793
CV	130.845	11.599	22.600	14.236	39.396	187.352
JB test	502.388 (<0.01)	24.745 (<0.01)	18.448 (<0.01)	112.628 (<0.01)	16.636 (<0.01)	1054.182 (<0.01)
ADF test	−5.796 (0.010)	−4.033 (0.010)	−3.489 (0.044)	−3.529 (0.040)	−2.954 (0.174)	−4.037 (0.010)
TV test	45.171 (<0.01)	17.441 (<0.01)	9.125 (<0.01)	11.397 (<0.01)	7.192 (0.27)	9.644 (<0.01)

SD, Standard deviation; CV, Coefficient of variation; JB, Jarque–Bera; ADF, augmented Dickey–Fuller; TV, Teraesvirta. The values in parentheses denote the *p*-values.

TABLE 2 Descriptive statistics of the pest population and climatic data from Marteru.

Statistic	Pest population	Tmax (°C)	Tmin (°C)	RHM (%)	RHE (%)	RF (mm)
Mean	1463.657	31.019	22.810	88.432	64.745	17.126
SD	2737.829	4.146	2.951	4.324	12.823	34.753
Median	390.000	30.600	23.200	89.000	65.700	0.000
Minimum	0.000	23.700	15.900	69.700	24.700	0.000
Maximum	19677.000	71.600	29.700	97.600	95.100	286.400
Skewness	3.403	4.627	−0.299	−1.025	−0.478	2.913
Kurtosis	13.888	37.794	−0.725	1.725	0.135	11.181
SE (Mean)	109.601	0.166	0.118	0.173	0.513	1.391
CV	187.054	13.366	12.936	4.889	19.805	202.922
JB test	6264.617 (<0.01)	39633.01 (<0.01)	22.736 (<0.01)	188.573 (<0.01)	24.374 (<0.01)	4163.524 (<0.01)
ADF test	−7.579 (<0.01)	−5.754 (<0.01)	−6.583 (<0.01)	−6.171 (<0.01)	−5.921 (<0.01)	−5.315 (<0.01)
TV test	9.466 (<0.01)	109.649 (<0.01)	28.368 (<0.01)	1.178 (0.555)	3.912 (0.141)	39.729 (<0.01)

SD, Standard deviation; CV, Coefficient of variation; JB, Jarque–Bera; ADF, augmented Dickey–Fuller; TV, Teraesvirta. The values in parentheses denote the *p*-values.

TABLE 3 Correlation value between the pest population and abiotic factors.

Regions	Tmax	Tmin	RHM	RHE	RF
Rajendranagar	0.383***	0.172**	−0.2675***	−0.2279***	−0.1375*
Marteru	0.1772***	0.0678	−0.1884***	−0.2887***	−0.1376**

‘*’, ‘**’ and ‘***’ indicate 5%, 1% and 0.1% level of significance respectively.

negative binomial distribution. Furthermore, the average pest population in Marteru was higher than that in Rajendranagar. Both pest datasets were positively skewed and leptokurtic in nature. Notably, the coefficient of variation (CV) for both datasets exceeded 100%, signifying a high degree of variation. Marteru data exhibited greater variability compared to the Rajendranagar data. The Jarque–Bera test was implemented to check the normality of the datasets, and it was found that all the data series were non-normal in nature. The stationarity of the data was examined using the augmented Dickey–Fuller (ADF) test. The ADF test inferred that, except for the RHE from Rajendranagar, all the data series followed the stationarity assumption. The results of the Teraesvirta neural network test suggested that, except for the RHM from Marteru, all the data series were non-linear. The insights from the descriptive statistics are further substantiated by Figure 2.

The Spearman's correlation of the pest populations with all the abiotic factors was computed and is presented in Table 3. Figure 5 represents the cross-correlation analysis of both regions. The correlation at lag k estimates the relationship between the pest population at time t + k and the climatic factors at time t. The correlation in Figure 5 clearly shows that the climatic factors had a significant effect on the incidence of the pest populations at different

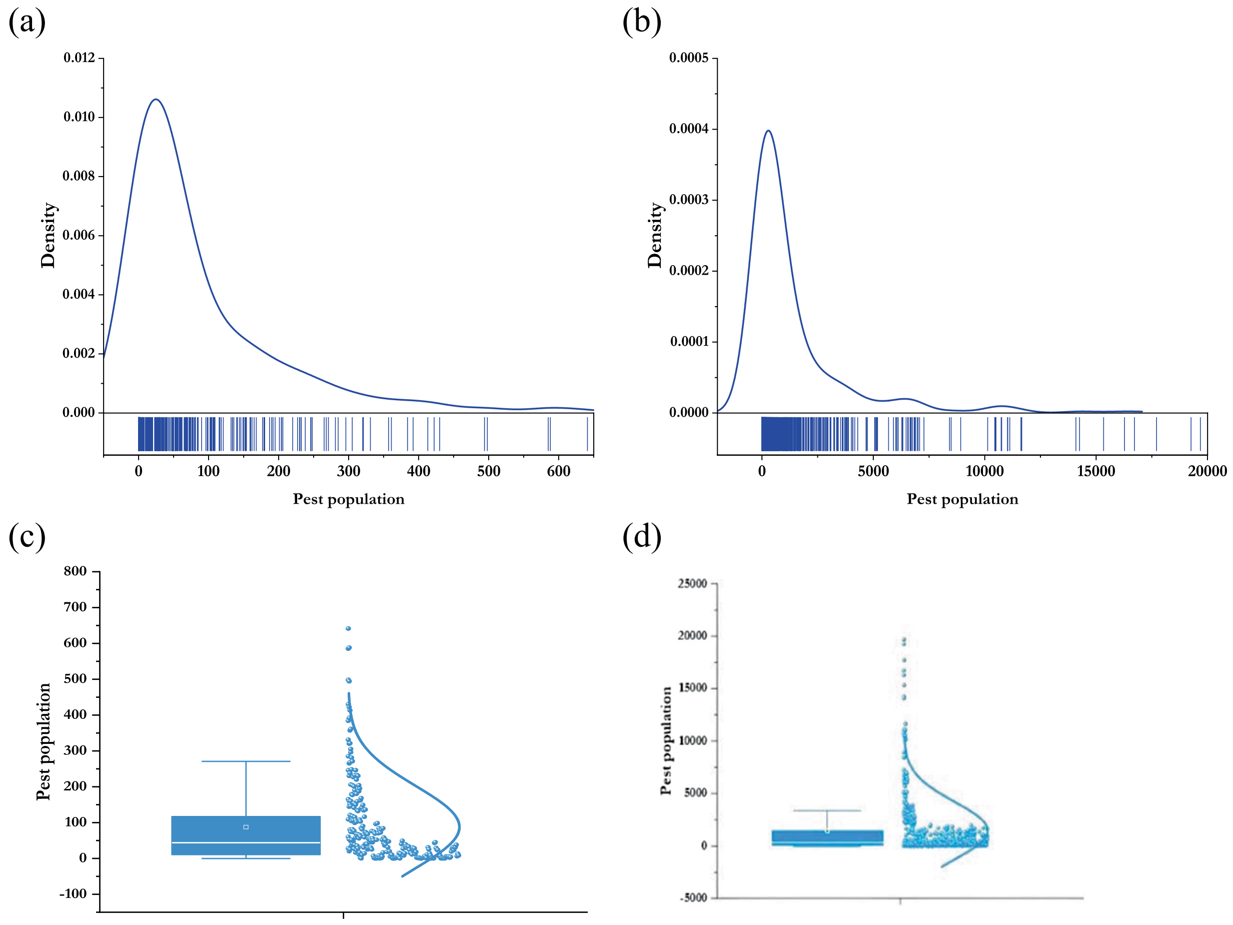

FIGURE 4
Kernel density **(A)** Rajendranagar; **(B)**: Marteru and box plot **(C)**: Rajendranagar; **(D)** Marteru of the pest populations.

lags. For the Rajendranagar region, Tmax, Tmin, RHM, and RHE had significant leading and lagging effects, and RF had significant lagging effects on the pest population. For the Marteru region, Tmax, Tmin, RHM, RHE, and RF had significant leading and lagging effects on the pest population.

Table 3 shows the significant association between the pest populations and abiotic factors. The pest populations had a positive correlation with maximum and minimum temperature and a negative correlation with RHM, RHE, and rainfall.

4.2 Implementation of machine learning models

The weekly pest population data were divided into training and test sets in an 80:20 ratio. For Marteru, the training set comprised the first 499 observations for model building, leaving the remaining 126 observations for model validation. Similarly, for Rajendranagar, the training set included the initial 260 observations for model building, while the remaining 52 observations were reserved for model validation. Four machine learning models—ANN, SVR, kNN, and RF—were employed to model the weekly pest population count, utilizing information on abiotic factors. The 12 SMWs were considered the number of lags. Hyperparameter optimization is a crucial step in building machine learning models. The grid search method was used to optimize the hyperparameters systematically. This algorithm explores a predefined range of hyperparameter values for a given model. The hyperparameter set that yielded the highest performance was selected and is presented in Table 4.

In the class of ensemble models, unweighted, fixed-weighted, and dynamic-weighted ensemble models were implemented, as outlined in the methodology. In the fixed-weighted ensemble methods, the weights were optimized using population-based optimization algorithms, such as PSO, and the results are detailed in Table 5. The values of the weights for the fixed-weighted ensemble methods showed that, for the Rajendranagar region, the highest weightage was given to the ANN (0.441), whereas the lowest weightage was given to the RF (0.111). For the Marteru region, the highest weightage was given to the kNN (0.509), whereas the lowest weightage was given to the ANN (0.089). These variations in the performance of the models for both regions suggest that no single model can perform better in all the cases. As for the optimized weights, the ANN was given more importance in the Rajendranagar region but less weightage in the Marteru region among all the candidate models. These differences also support the use of ensemble models for prediction rather than relying

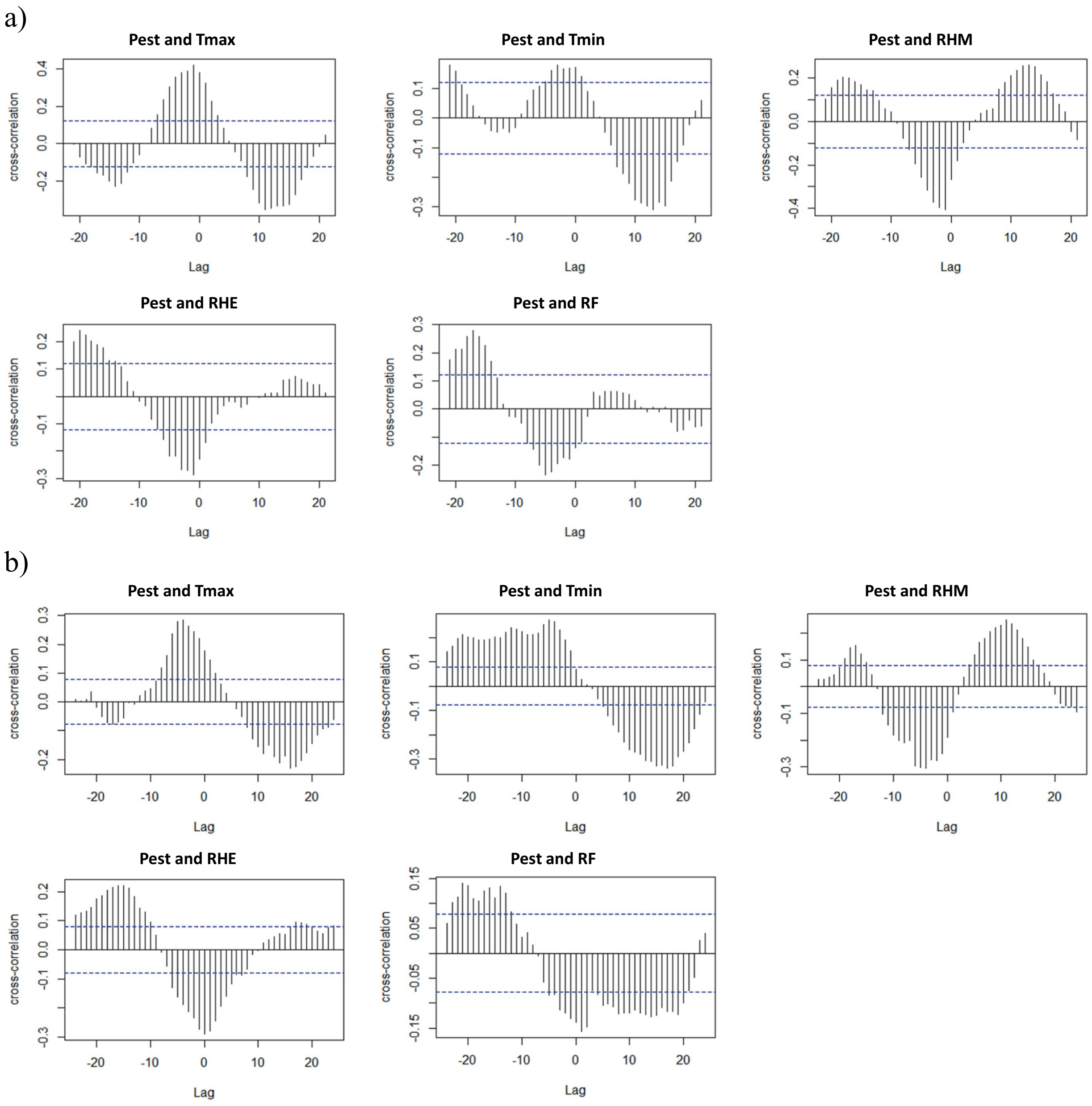

FIGURE 5
Cross-correlation analysis **(A)** Rajendranagar; **(B)** Marteru.

TABLE 4 **Hyperparameter specification for the machine learning models.**

Model	Parameters	Rajendranagar	Marteru
ANN	Model (Input-Hidden-Output)	9–6-1	9–9-1
	Sigma	21.29	17.33
	Learning rate	0.001	0.003
SVR	Cost	0.50	0.25
	Kernel	Radial Basis	Radial Basis
kNN	K	7	9
	Distance	2	2
RF	mtry	3	5

TABLE 5 Optimized weight obtained for the ML model in the fixed-weighted ensemble model.

Datasets	ANN	SVR	kNN	RF
Rajendranagar	0.441	0.264	0.184	0.111
Marteru	0.089	0.111	0.509	0.291

TABLE 6 Performance measures of the implemented models.

Models	Rajendranagar			Marteru		
	RRSE	MDAE	RMSLE	RRSE	MDAE	RMSLE
DWE-ALE	0.917	32.104	2.049	1.275	1251.066	2.527
DWE-Logistic	0.932	34.042	2.104	1.285	1250.814	2.537
FWE	1.007	37.077	2.127	1.281	1273.488	2.531
UWE	0.999	37.875	2.143	1.326	1280.185	2.538
SVR	1.112	40.413	2.145	1.387	1287.615	2.730
RF	1.100	42.970	2.167	1.611	1371.059	2.584
kNN	1.105	43.019	2.188	2.137	1537.189	2.788
ANN	1.123	38.464	2.358	1.570	2350.742	2.754

UWE, Unweighted ensemble; FWE, Fixed weighted ensemble; DWE-ALE, Dynamic weighted ensemble using absolute log error; DWE-logistic, Dynamic weighted ensemble using logistic error function.

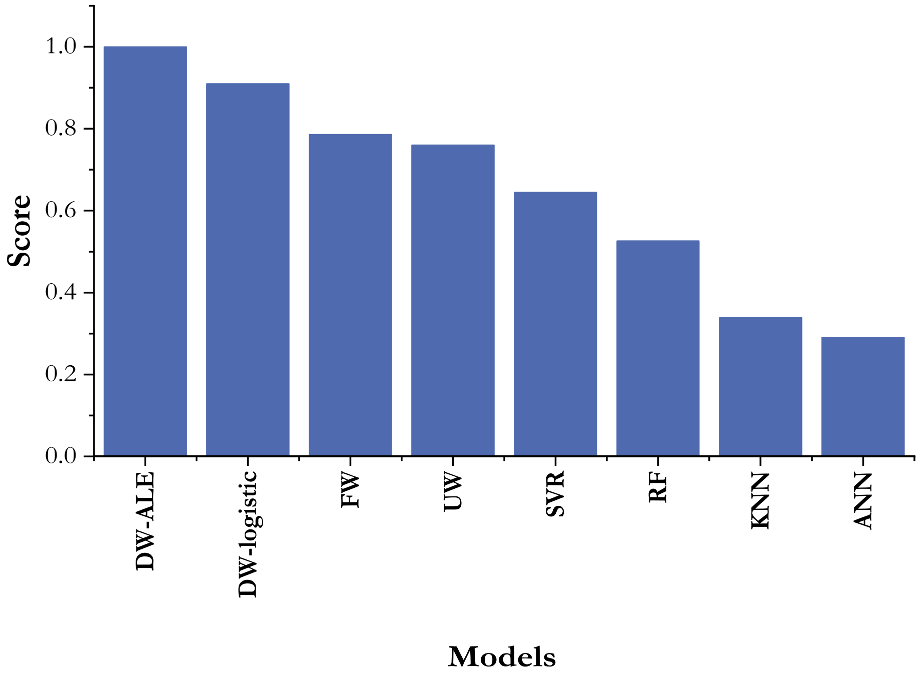

FIGURE 6
Combined TOPSIS scores of all implemented models.

on a single model. Performance measures based on the testing data were calculated and are presented in Table 6.

According to Table 6, in the Rajendranagar region, the dynamic-weighted ensemble model outperformed all other models across all three performance measures. Within the dynamic ensemble models, the model utilizing the error function RRSE exhibited superior performance. Both the fixed-effect and weighted ensemble models outperformed the other machine-learning models. Between the fixed-effect and weighted ensemble models, except for the RRSE, the weighted ensemble models proved to be superior based on the other two measures.

For the Marteru region, the DWE-ALE outperformed the others based on the RRSE and RMSLE, while the DWE-logistics excelled in terms of MDAE. However, the fixed-weighted ensemble models performed better than the DWE-logistics model based on the RRSE and RMSLE. Although the proposed model performed better than the

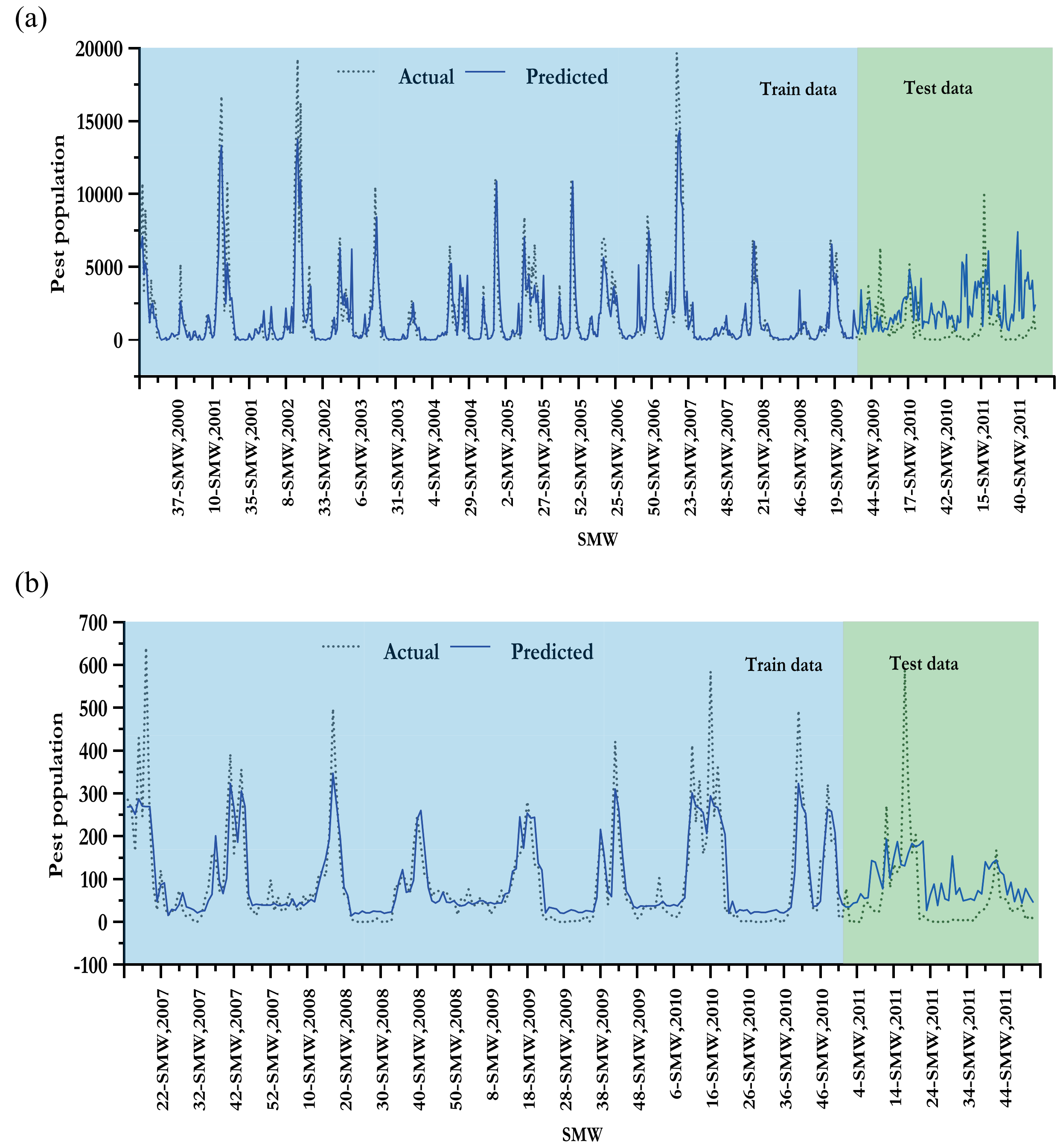

FIGURE 7
Plot of the actual and predicted values of the best-fitted models for **(A)** Rajendranagar and **(B)** Marteru.

other models in the data having extreme variation, the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) was employed to identify the best model among all implemented models across both datasets and three measures. The rankings and TOPSIS scores are presented in Figure 6.

Figure 6 clearly illustrates that the dynamic ensemble model outperformed the other models, and among the dynamic models, the DWE-ALE model exhibited superior performance. Following the dynamic models, the fixed-effect ensemble model performed better than the unweighted ensemble model. Among the machine learning models, the SVR model, followed by the RF model, demonstrated high prediction accuracy, while the ANN model ranked lower in the TOPSIS score. The actual and fitted values of the best model, the DWE-ALE, are depicted in Figure 7.

5 Conclusion

The development of a pest forewarning system remains a significant challenge for farmers and policymakers. This study provides a detailed overview of an ensemble model with a time-dependent weighting scheme and introduces a time-varying dynamic ensemble approach with ALE and logistic error functions using four candidate models, namely ANN, SVR, kNN, and RF, for pest infestation modeling. Abiotic factors, such as minimum and maximum temperature, rainfall, and morning and evening relative humidity, were also considered in the model. This study found that the YSB population of rice had a positive correlation with maximum and minimum temperature and a negative correlation with RHM, RHE, and rainfall. The proposed algorithms were empirically compared with fixed-weighted and unweighted ensemble methods, as well as all four candidate models, using three performance metrics and the TOPSIS algorithm. The dynamic ensemble algorithm demonstrated higher predictive accuracy in predicting the YSB in rice crops in both regions. These algorithms showed promise for implementation in pest risk warning systems, aligning well with observed trends in actual field data for pests. Consequently, farmers can proactively manage pests using the predictive insights provided by our proposed model. This study on dynamic ensemble models paves the way for a new era of hybrid models. In future studies, this model can be further examined using other pest populations from various crops. This model can also be improved by incorporating other error functions and advanced multivariate learning models for weight forecasting.

Data availability statement

Publicly available datasets were analyzed in this study. This data can be found at: http://www.icar-crida.res.in:8080/naip/AccessData.jsp.

Author contributions

AKS: Data curation, Formal analysis, Investigation, Methodology, Writing – original draft. MY: Investigation, Methodology, Software, Validation, Writing – review & editing. RP: Conceptualization, Supervision, Visualization, Writing – review & editing. AP: Conceptualization, Supervision, Validation, Writing – review & editing. AS: Data curation, Formal analysis, Writing – review & editing.

Funding

The author(s) declare that no financial support was received for the research, authorship, and/or publication of this article.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

1. Anwar N, Ahmad I, Kiani AK, Shoaib M, MAZ R. Numerical treatment for mathematical model of farming awareness in crop pest management. *Front Appl Math Stat*. (2023) 9:1208774. doi: 10.3389/fams.2023.1208774

2. Deutsch CA, Tewksbury JJ, Tigchelaar M, Battisti DS, Merrill SC, Huey RB, et al. Increase in crop losses to insect pests in a warming climate. *Science*. (2018) 361:916–9. doi: 10.1126/science.aat3466

3. Rani L, Thapa K, Kanojia N, Sharma N, Singh S, Grewal AS, et al. An extensive review on the consequences of chemical pesticides on human health and environment. *J Clean Prod*. (2021) 283:124657. doi: 10.1016/J.JCLEPRO.2020.124657

4. Olatinwo R, Hoogenboom G. Weather-based Pest forecasting for efficient crop protection In: DP Abrol, editor. Integrated Pest management: Current concepts and ecological perspective. London: Academic Press (2014). 59–78.

5. Li T, Yang J, Chen Z. The early warning and prediction method of flea beetle based on maximum likelihood algorithm ensembles. *Proc Int Conf Nat Comput*. (2010) 4:1901–5. doi: 10.1109/ICNC.2010.5584642

6. Patel HR, Shekh AM. Pest epidemics and role of meteorological services: an overview. *J Agrometeorol*. (2006) 8:104–13. doi: 10.54386/JAM.V8I1.884

7. Arya P, Paul RK, Kumar A, Singh K. N., Sivaramne N, Chaudhary P. Predicting pest population using weather variables an ARIMAX time series framework. *Int J Agric Statist Sci*. (2015) 11:381–6.

8. Paul RK, Vennila S, Singh N, Chandra P, Yadav S. K., Sharma O. P., et al. Seasonal dynamics of sterility mosaic of Pigeonpea and its prediction using statistical models for seasonal dynamics of sterility mosaic of Pigeonpea and its prediction using statistical models for Banaskantha region of Gujarat, India. *J Indian Soc Agric Statist*. (2018) 72:213–23.

9. Liu C, Zhai Z, Zhang R, Bai J, Zhang M. Field pest monitoring and forecasting system for pest control. *Front Plant Sci*. (2022) 13:990965. doi: 10.3389/fpls.2022.990965

10. Neta A, Levi Y, Morin E, Morin S. Seasonal forecasting of pest population dynamics based on downscaled SEAS5 forecasts. *Ecol Model*. (2023) 480:110326. doi: 10.1016/J.ECOLMODEL.2023.110326

11. Lankin-Vega G, Worner SP, Teulon DAJ. An ensemble model for predicting *Rhopalosiphum padi* abundance. *Entomol Exp Appl*. (2008) 129:308–15. doi: 10.1111/J.1570-7458.2008.00778.X

12. Yan Y, Feng CC, Wan MPH, Chang KTT. Multiple regression and artificial neural network for the prediction of crop pest risks. In Information Systems for Crisis Response and Management in Mediterranean Countries: Second International Conference, ISCRAM-med 2015, Tunis, Tunisia, October 28-30, 2015, Proceedings 2. Springer International Publishing. (2015) 73–84

13. Paul RK, Vennila S, Bhat MN, Yadav SK, Sharma VK, Nisar S, et al. Prediction of early blight severity in tomato (*Solanum lycopersicum*) by machine learning technique. *Indian J Agri Sci*. (2019) 89:1921–7. doi: 10.56093/ijas.v89i11.95344

14. Skawsang S, Nagai M, Tripathi NK, Soni P. Predicting Rice Pest population occurrence with satellite-derived crop phenology, ground meteorological observation, and machine learning: a case study for the central plain of Thailand. *Appl Sci*. (2019) 9:4846. doi: 10.3390/APP9224846

15. Paul RK, Vennila S, Yadav SK, Bhat MN, Kumar M, Chandra P, et al. Weather based forecasting of sterility mosaic disease in pigeonpea (*Cajanu cajan*) using machine learning techniques and hybrid models. *Indian J Agri Sci*. (2020) 90:1952–8. doi: 10.56093/ijas.v90i10.107971

16. Marković D, Vujičić D, Tanasković S, Đorđević B, Ranđić S, Stamenković Z. Prediction of Pest insect appearance using sensors and machine learning. *Sensors*. (2021) 21:4846. doi: 10.3390/S21144846

17. Ibrahim EA, Salifu D, Mwalili S, Dubois T, Collins R, Tonnang HE. An expert system for insect pest population dynamics prediction. *Comput Electron Agric*. (2022) 198:107124. doi: 10.1016/j.compag.2022.107124

18. Paul RK, Vennila S, Yeasin M, Yadav SK, Nisar S, Paul AK, et al. Wavelet decomposition and machine learning technique for predicting occurrence of spiders in pigeon pea. *Agronomy*. (2022) 12:1429. doi: 10.3390/AGRONOMY12061429

19. Sidumo B, Sonono E, Takaidza I. Count regression and machine learning techniques for zero-inflated Overdispersed count data: application to ecological data. *Ann Data Sci*. (2023) 11:1–15. doi: 10.1007/S40745-023-00464-6

20. Galicia A, Talavera-Llames R, Troncoso A, Koprinska I, Martínez-Álvarez F. Multi-step forecasting for big data time series based on ensemble learning. *Knowl-Based Syst*. (2019) 163:830–41. doi: 10.1016/j.knosys.2018.10.009

21. Sharma N, Dev J, Mangla M, Wadhwa VM, Mohanty SN, Kakkar D. A heterogeneous ensemble forecasting model for disease prediction. *N Gener Comput*. (2021) 39:701–15. doi: 10.1007/s00354-020-00119-7

22. Abdelhamid AA, Alotaibi SR. Optimized two-level ensemble model for predicting the parameters of metamaterial antenna. *Comput Mater Continua*. (2022) 73:917–933

23. Du L, Gao R, Suganthan PN, Wang DZ. Bayesian optimization based dynamic ensemble for time series forecasting. *Inf Sci*. (2022) 591:155–75. doi: 10.1016/j.ins.2022.01.010

24. Gangula R, Thirupathi L, Parupati R, Sreeveda K, Gattoju S. Ensemble machine learning based prediction of dengue disease with performance and accuracy elevation patterns. *Mater Today Proc*. (2023) 80:3458–63. doi: 10.1016/J.MATPR.2021.07.270

25. Sun S, Jin F, Li H, Li Y. A new hybrid optimization ensemble learning approach for carbon price forecasting. *Appl Math Model*. (2021) 97:182–205. doi: 10.1016/j.apm.2021.03.020

26. Jiang M, Jia L, Chen Z, Chen W. The two-stage machine learning ensemble models for stock price prediction by combining mode decomposition, extreme learning machine and improved harmony search algorithm. *Ann Oper Res*. (2022) 309:553–85.

27. de Oliveira Aparecido LE, de Souza RG, Moraes JR, CTS C, de Souza PS. Machine learning algorithms for forecasting the incidence of *Coffea arabica* pests and diseases. *Int J Biometeorol*. (2020) 64:671–88. doi: 10.1007/s00484-019-01856-1

28. Cao L-J, Tay FEH. Support vector machine with adaptive parameters in financial time series forecasting. *IEEE Trans Neural Netw*. (2003) 14:1506–18. doi: 10.1109/TNN.2003.820556

29. Cherkassky V, Ma Y. Practical selection of SVM parameters and noise estimation for SVM regression. *Neural Netw*. (2004) 17:113–26. doi: 10.1016/S0893-6080(03)00169-2

30. Tay FEH, Cao L. Application of support vector machines in financial time series forecasting. *Omega*. (2001) 29:309–17. doi: 10.1016/S0305-0483(01)00026-3

31. Breiman L. Random forests. *Mach Learn*. (2001) 45:5–32. doi: 10.1023/A:1010933404324

32. Moon J, Kim Y, Son M, Hwang E. Hybrid short-term load forecasting scheme using random forest and multilayer perceptron. *Energies*. (2018) 11:3283. doi: 10.3390/EN11123283

33. Yeasin M, Paul RK. OptiSembleForecasting: optimization-based ensemble forecasting using MCS algorithm and PCA-based error index. *J Supercomput*. (2024) 80:1568–97. doi: 10.1007/s11227-023-05542-3

34. Deb S, Deb S. An ensemble method for early prediction of dengue outbreak. *J R Stat Soc Ser A*. (2022) 185:84–101. doi: 10.1111/RSSA.12714

35. Niazkar HR, Niazkar M. Application of artificial neural networks to predict the COVID-19 outbreak.. *Global Health Research and Policy*. (2020) 5:1–11.