E-ISSN: 2618-0618
P-ISSN: 2618-060X

www.agronomyjournals.com
2024; SP-7(8): 870-880
Received: 20-06-2024
Accepted: 18-07-2024

Karthik VC
ICAR-Indian Agricultural Research Institute, New Delhi, India

B Samuel Naik
Banaras Hindu University (BHU), Varanasi, Uttar Pradesh, India

Veershetty
ICAR-Indian Agricultural Research Institute, New Delhi, India

Shreya S Hanji
ICAR-Indian Agricultural Research Institute, New Delhi, India

ASB Sujith
ICAR-Indian Agricultural Research Institute, New Delhi, India

Rakesh Chhalotre
ICAR-Indian Agricultural Research Institute, New Delhi, India

Corresponding Author:
Rakesh Chhalotre
ICAR-Indian Agricultural Research Institute, New Delhi, India

Spatial and exogenous variable-based machine learning models for enhanced paddy yield prediction

Karthik VC, B Samuel Naik, Veershetty, Shreya S Hanji, ASB Sujith and Rakesh Chhalotre

DOI: https://doi.org/10.33545/2618060X.2024.v7.i8Sk.1402

Abstract
Remote sensing technology has proven crucial in examining the relationship between paddy yield and various vegetation indices. This study, conducted in Petlurivaripalem, Andhra Pradesh, utilized satellite imagery from 2014 to 2023 to extract indices such as NDVI, GNDVI, SAVI, MSAVI, LAI, and LSWI. Accurate yield prediction is vital for India's economy, and this study evaluates the performance of several predictive models for paddy yield forecasting using these indices. The models assessed include traditional parametric approaches (ARIMAX, MLR), machine learning techniques (ANN, SVR, RFR), and advanced ensemble methods like XGBoost. The results indicate that XGBoost consistently outperforms other models, delivering the lowest error metrics across all vegetation indices. Specifically, XGBoost achieved the best results with the GNDVI index, recording an RMSE of 50.85, MAE of 42.1, sMAPE of 12.1, MASE of 1.086, and QL of 20.05. These lower error metrics highlight XGBoost's superior accuracy compared to traditional and machine learning models. This study underscores the importance of remote sensing technology in capturing crop development patterns and forecasting paddy yield with precision, providing valuable insights for agricultural planning and decision-making.

Keywords: Machine learning, vegetation indices, paddy, GNDVI, XGBoost

1. Introduction
Agriculture and allied sectors remain pivotal to the Indian economy, contributing significantly to the Gross Value Added (GVA). Paddy (*Oryza sativa L.*) is a vital crop in India, largely cultivated in regions with abundant water resources. India holds the position of the second-largest rice producer globally, following China. On the international stage, India's paddy cultivation is responsible for approximately 22% of the global rice-growing area and 24% of the total production, underscoring its essential role in global food security. Key rice-producing states in India include West Bengal, Andhra Pradesh, Punjab, Tamil Nadu, and Uttar Pradesh. The nation consistently produces substantial quantities of rice, with production levels reaching around 135.5 million tonnes in recent years, as noted by the Department of Agriculture, India.

Crop yield estimation plays a crucial role in helping farmers mitigate production losses during adverse conditions and optimize yields under favourable circumstances (1). Traditionally, many countries rely on conventional data collection methods and ground-based field reports for estimating crop yields (2). However, in recent years, a range of mathematical models and machine learning techniques have been developed to enhance the accuracy and efficiency of crop yield estimation (3).

Remote sensing involves acquiring information about objects or phenomena without direct physical contact, utilizing electromagnetic radiation as an information carrier to gather data from a distance (4). This technique is particularly valuable in agriculture, offering timely and quantitative information about crops across vast regions (5). Various methods have been developed using remote sensing to estimate crop yields effectively (6). Spectral measurements obtained through remote sensing provide critical insights into numerous crop parameters throughout the growth cycle. These include leaf area index (LAI), plant growth, plant density, crop canopy area, plant population, and Karthikeyan canopy nitrogen status (7).

Such measurements are essential for monitoring and managing crop health and productivity, enabling more informed decisions to enhance agricultural outcomes.

Numerous studies have investigated agricultural crop yield prediction using various machine learning techniques. Traditional time-series models, such as ARIMA and its variants (SARIMA and ARIMAX), have been extensively applied to understand agricultural data relationships Makridakis *et al.* (2018) [8]. For instance, Anggraeni *et al.* (2017) [9] compared ARIMAX and VAR models for rice price prediction in Thailand, finding ARIMAX to perform better. However, challenges with nonstationary, nonlinear, and noisy data have led researchers to explore advanced methods like machine learning (ML) and deep learning (DL) models (10). Naik *et al.* (2023) [11] applied advanced ML techniques, including KNN, DT, SVM, RF, and LASSO regression, combined with VIs, for wheat yield prediction. Hamjah and Chowdhury (2014) [12] used the ARIMAX model to assess the impact of climatic and hydrological factors on cash crop production in Bangladesh, presenting a novel approach in agriculture. HT *et al.* (2024) [13] performed a comparative analysis of time-series models and ML techniques for onion price forecasting. Additionally, Amaratunga *et al.* (2020) [14] demonstrated that optimizing regressors with Artificial Neural Networks (ANNs) could significantly enhance the yield prediction of paddy, highlighting the importance of ML in agricultural yield optimization.

This study addresses a critical gap in existing research by focusing on paddy yield prediction, leveraging both the primary target variable and related vegetation indices (VIs) as exogenous variables (10,15). Additionally, the research conducts a comprehensive comparative analysis of various machine learning techniques, including Artificial Neural Networks (ANNs), Support Vector Regression (SVR), Random Forest Regression (RFR), and XGBoost, alongside statistical models such as Autoregressive Integrated Moving Average with Exogenous variables (ARIMAX) and Multiple Linear Regression (MLR). The aim is to improve paddy yield prediction by integrating machine learning models that incorporate VIs as exogenous factors.

2. Materials and Methods

2.1 Study area

This study has been conducted in the Petlurivaripalem village, which is located in the Palnadu district of Andhra Pradesh state, India. The district's geographical coordinates are latitudes of 16.15'24° N and the longitudes of 80.01'44°. figure 1 depicts the selected paddy growing fields in the Petlurivaripalem village.

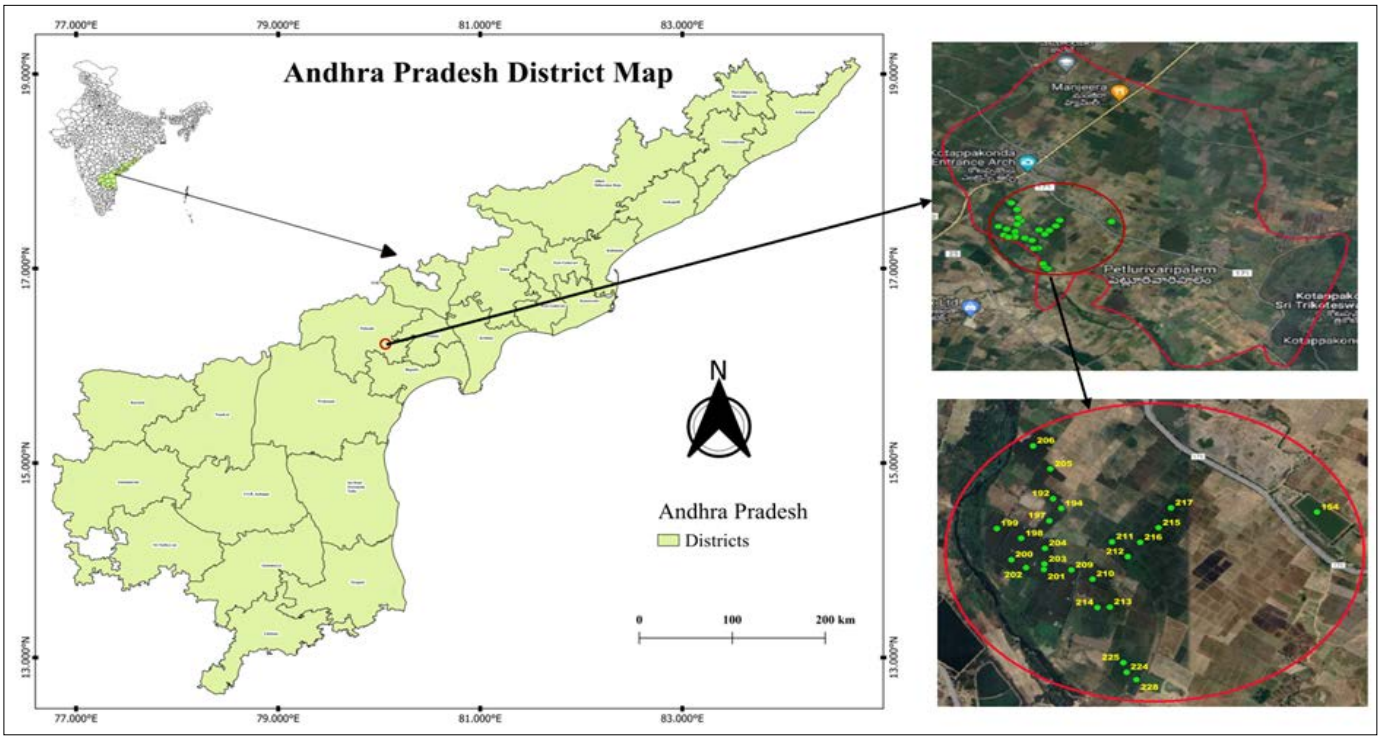

Fig 1: Selected paddy growing fields in Petlurivaripalem village

2.2 Data description

Experimental data consists of three parts *i.e.* ground truth data, satellite data and ancillary data.

2.2.1 Ground Truth data

In this study paddy yield data in time series from 2015-2023 was taken from Petlurivaripalem village Rythu Bharosa Kendra (RBK). Data includes yield and latitude and longitude of 25 fields were collected.

2.2.2 Satellite data

Satellite data were derived from Sentinel-2 (S-2) multispectral data, which consist of 13 spectral bands, each with specific wavelength ranges, allowing for detailed analysis and interpretation of the Earth's surface characteristics (https://lta.cr.usgs.gov/sentinel_2). The S-2 data were used for land cover and land use map preparation and for the generation of various vegetation indices. For this study, S-2 images collected from 2015-2023 during crop window period of each year.

2.2.3 Ancillary data

In addition to the S-2 multispectral data, ancillary data were used, including the district boundary map, village boundary map of Petlurivaripalem which were digitized using ArcGIS software and allowed for precise spatial analysis and integration with

other datasets. A village boundary map of Petlurivaripalem in the form of a shapefile was acquired from a survey of India website.
https://onlinemaps.surveyofindia.gov.in/FreeMapSpecification.aspx.

2.3 Methodology for Sentinel-2 Data Extraction

Initially, the ancillary data along with ground truth data were used for the development of region of interest (ROI). Further, the S-2 images were selected from the google earth engine (GEE) catalogue by applying a cloud cover filter, ensuring that only scenes with minimal or no clouds were included. Specifically, a threshold of less than 5% cloud cover was set for the filtering process. The selected S-2 images are from 2015 to 2023, corresponding to the paddy growing season, which spans from germination to full maturity. After the pre-processing of the S-2 image various vegetation indices were calculated based on the ground truth data.

In this study various vegetation indices are extracted such as normalized difference vegetation index (NDVI), soil-adjusted vegetation index (SAVI), green normalized difference vegetation index (GNDVI), modified soil-adjusted vegetation index (MSAVI), leaf area index (LAI). Figure 2 illustrates the schematic diagram of digital image processing steps for generation of vegetation indices using GEE.

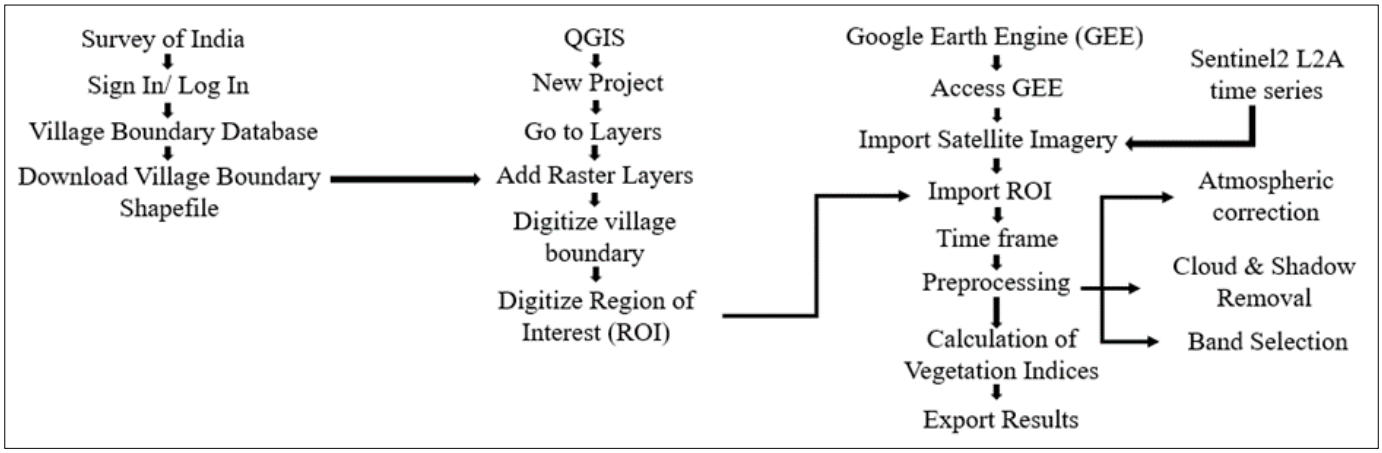

Fig 2: Schematic diagram of digital image processing steps for generation of vegetation

2.3.1 Normalized Difference Vegetation Index (NDVI)

Rouse *et al*. (1974) first proposed NDVI which is a commonly used remote sensing tool for evaluating vegetation presence and health. NDVI is calculated based on the reflectance values from the near-infrared (NIR) and red (R) spectral bands. The NDVI formula is:

$$NDVI = \frac{NIR - R}{NIR + R} \quad (1)$$

NDVI values range from-1 to 1, with values near 1 indicating healthy, dense vegetation, values around 0 suggesting bare soil or urban areas, and negative values typically representing water bodies. This index provides a quantitative measure of vegetation health, aiding in the assessment of the density and vigor of crops, such as paddy.

2.3.2 Green Normalized Difference Vegetation Index (GNDVI)

An improved version of the traditional NDVI known as GNDVI, incorporates the green spectral band along with the NIR band. This index is particularly effective for monitoring the health and chlorophyll content of vegetation (17). The formula for GNDVI is:

$$GNDVI = \frac{(NIR - G)}{(NIR + G)} \quad (2)$$

Where "G" represents the reflectance of the green band. GNDVI values range from-1 to 1, with values near 1 indicating healthy vegetation with high chlorophyll content, while values closer to-1 typically represent non-vegetated surfaces or water bodies.

2.3.3 Soil Adjusted Vegetation Index (SAVI)

Similar to NDVI, SAVI includes a correction for the influence of bare soil, making it particularly valuable in areas with sparse vegetation. This adjustment reduces the impact of soil background on the vegetation signal. The formula for SAVI is:

$$SAVI = \frac{(NIR - RED)(1 + L)}{(NIR + RED + L)} \quad (3)$$

Where L is a user-defined parameter, typically set at 0.5, to account for soil background effects. SAVI values range from-1 to +1, with higher values indicating healthier vegetation. Although SAVI was designed to correct for soil brightness, it can still be influenced by soil background variations due to the adjustment parameter L (Major *et al*. 1990) [18]. This index is especially useful for agricultural crop monitoring.

2.3.4 Modified Soil-Adjusted Vegetation Index (MSAVI)

MSAVI enhances SAVI by further minimizing soil background effects, thereby improving the accuracy of vegetation signal detection. This makes it especially useful in regions with sparse vegetation (19). The formula for MSAVI is:

$$MSAVI = \frac{\left(2 \times NIR + 1 - \sqrt{((2 \times NIR + 1)^2 - 8 \times (NIR - Red))}\right)}{2} \quad (4)$$

MSAVI values range from-1 to +1, where higher values indicate healthier vegetation, and lower values correspond to less vegetation or bare soil. MSAVI is particularly beneficial in agricultural settings where soil background can affect vegetation indices, offering more precise assessments of vegetation cover and health.

2.3.5 Leaf Area Index (LAI)

LAI quantifies the total leaf area relative to ground area and is a key indicator of vegetation density and health (20). LAI can be estimated using remote sensing data through various models and indices, such as NDVI or SAVI. The formula for LAI varies depending on the model, but it generally reflects the extent of vegetation cover:

$$\text{LAI} = \frac{1}{\text{K}} \times \left(\frac{(\text{K} \times (\text{NIR} - \text{Red}))}{(\text{NIR} + \text{Red})} + 1 \right) \quad (5)$$

Where the empirical constant KKK is typically 1.5. LAI values are generally positive, with higher values indicating denser vegetation and greater leaf coverage. LAI is essential for estimating foliage amount, light interception, and photosynthetic potential, all of which are critical for predicting paddy yields.

2.3.6 Leaf Surface Water Index (LSWI)

The Land Surface Water Index (LSWI) is a vital vegetation index used to assess water content in vegetation, which is essential for evaluating plant health and water status. It is particularly sensitive to leaf water content, making it valuable for detecting drought stress or assessing water levels during various growth stages (21,22). LSWI is calculated using the NIR and SWIR bands from remote sensing data.

$$\text{LSWI} = \frac{(\text{NIR} - \text{SWIR})}{(\text{NIR} + \text{SWIR})} \quad (6)$$

The index typically ranges from-1 to 1, with positive values indicating higher water content in vegetation. LSWI is often used alongside other indices like NDVI to provide a comprehensive evaluation of crop conditions, especially for monitoring drought, determining irrigation needs, and assessing overall vegetation health. Higher LSWI values suggest adequate water content, essential for maintaining healthy crop growth, while lower values may signal water stress, potentially leading to reduced yields. In agriculture, LSWI aids in informed decision-making regarding irrigation practices and monitoring water availability's effects on crop development, which are critical for optimizing paddy yield predictions. Compared to NDWI, which primarily focuses on vegetation water content, LSWI offers a more detailed assessment by effectively capturing water presence in vegetation, making it particularly useful for hydrological studies and water resource management.

2.4 Methodology for yield prediction

2.4.1 Autoregressive Integrated Moving Average with Exogenous Inputs (ARIMAX)

The ARIMAX model is an extension of the traditional ARIMA model that improves time series forecasting accuracy by incorporating external predictors. The standard ARIMA model, denoted as ARIMA (p, d, q), captures temporal dependencies in the data, where '*p*' represents the order of the autoregressive component, '*d*' is the degree of differencing, and '*q*' indicates the order of the moving average component. ARIMAX builds on this by including exogenous variables—external factors that influence the time series data—making it more adaptable for various applications.

Mathematically, the ARIMAX $(p, d, q)(P, D, Q)_s$ model is expressed as:

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q} + X_t \beta + \varepsilon_t \quad (7)$$

Where Y_t represents the observed value at time t, c is a constant term, $\phi_1, \phi_2, \cdots, \phi_p$ are autoregressive coefficients, $\varepsilon_{t-1}, \varepsilon_{t-2}, \cdots, \varepsilon_{t-q}$ are past error terms, X_t represents the exogenous input variables at time t, β denotes the coefficients for the exogenous variables, and ε_t is the error term at time t.

To estimate ARIMAX models, the autoregressive, differencing, and moving average components are fitted to historical time series data while incorporating exogenous variables. These external inputs allow the model to account for factors beyond the immediate time series that may influence the data, resulting in improved forecasting performance (9). ARIMAX models are particularly useful when a clear temporal pattern exists in the data, and when additional external variables offer critical insights for making more accurate predictions.

2.4.2 Multiple Linear Regression (MLR)

Multiple Linear Regression (MLR) is a fundamental statistical technique used to model the relationship between a dependent variable Multiple Linear Regression (MLR) is a fundamental statistical technique used to model the relationship between a dependent variable (y) and two or more independent variables $(x_1, x_2, \ldots, x_n)$. The model assumes a linear relationship between the dependent variable and the predictors, represented by the equation:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n + \varepsilon \quad (8)$$

In this equation, β_0 is the intercept, $\beta_0, \beta_1, \beta_2, \ldots, \beta_n$ are the coefficients that measure the influence of each independent variable $x_1, x_2, \ldots, x_n$, and ε is the error term that accounts for the unexplained variability in the data.

The main objective of MLR is to estimate these coefficients $(\beta_0, \beta_1, \beta_2, \ldots, \beta_n)$ in a way that minimizes the sum of the squared differences between the observed values (y) and the predicted values $(\hat{y})$. This is typically achieved using the method of least squares, which minimizes the Residual Sum of Squares (RSS), calculated as:

$$RSS = \sum_{i=1}^{N} (y_i - \hat{y}_i)^2 \quad (9)$$

Here, NNN is the number of observations, and the predicted values $(\hat{y}_i)$ are obtained by multiplying each independent variable $(x_1, x_2, \ldots, x_n)$ by its corresponding coefficient $(\beta_0, \beta_1, \beta_2, \ldots, \beta_n)$, adding the intercept (β_0), and considering the error term (ε). MLR is widely applied across various fields to explore and quantify the relationships between multiple variables, making it a crucial tool for predictive modelling and data analysis (23,24).

2.4.3 Artificial Neural Networks (ANN)

Artificial Neural Networks (ANNs) are a machine learning model inspired by the structure and function of neurons in the human brain, making them particularly effective for regression tasks where they can identify and model complex patterns in data. ANNs consist of layers of interconnected nodes, including an input layer, one or more hidden layers, and an output layer. Each connection between nodes is associated with a weight, and each node processes the weighted sum of its inputs using an activation function.

In regression tasks, the output layer typically consists of a single node that represents the predicted continuous value ($\hat{y}_i$). During training, ANNs adjust their weights through backpropagation, a process that calculates the error between the predicted output and the actual target values (y) and updates the weights to reduce this error. The training process aims to minimize the Mean Squared Error (MSE), a metric that quantifies the average squared difference between predicted and actual values:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad (10)$$

Where, n is the number of data points, $\hat{y}_i$ is the predicted value for the i-th instance, and y_i is the actual target value.

A crucial operation in an ANN involves calculating the weighted sum of inputs (z_i) and applying an activation function (a_i). Common activation functions include the sigmoid function, hyperbolic tangent (*tanh*), and rectified linear unit (ReLU). These functions introduce non-linearities into the model, enabling ANNs to learn and capture complex relationships within the data. The output ($\hat{y}_i$) of the *i*-th node in the network is determined by:

$$\hat{y}_i = a_i(z_i) \quad (11)$$

ANNs are highly capable of learning intricate patterns from data, making them suitable for various regression applications. By adjusting weights and biases during training, ANNs can approximate complex functions, allowing for accurate modeling and prediction of continuous outcomes. Their ability to capture non-linear relationships makes ANNs a powerful tool for regression analysis across diverse fields (15,25,26).

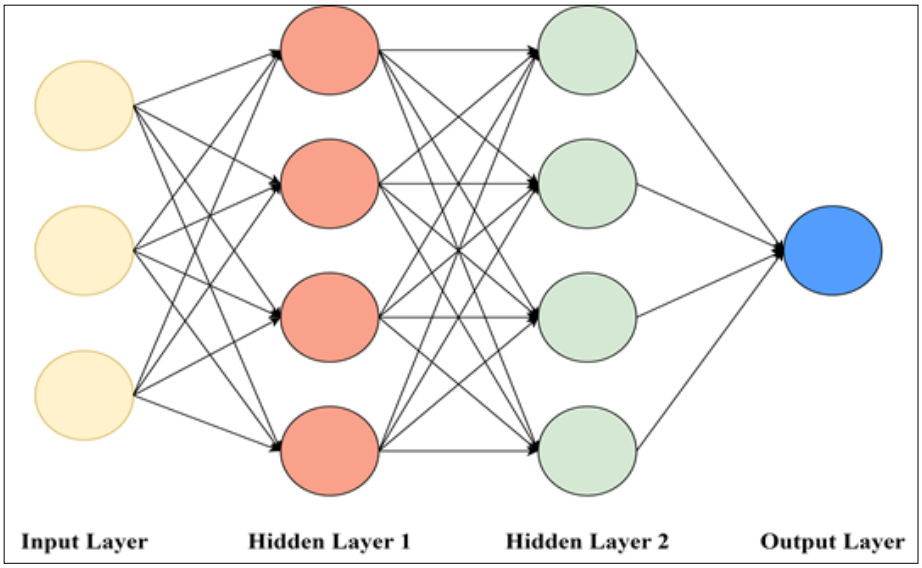

Fig 3: Architecture of ANN

2.4.4 Support Vector Machines (SVM)

Support Vector Regression (SVR) is a robust machine learning algorithm commonly employed for regression tasks. The fundamental idea behind SVR is to identify a hyperplane that best fits the data while maximizing the margin, which is the distance between the hyperplane and the nearest data points, known as support vectors. The objective is to minimize prediction errors while allowing for a defined margin of tolerance.

Mathematically, SVR aims to find a function $f(x)$ that predicts target values (y) based on input features (y). The objective function for SVR is given by:

$$Minimize\ \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\left(max(0, |y_i - f(x_i)| - \epsilon)\right)^2 \quad (12)$$

In this equation, www represents the weights, C is the regularization parameter controlling the trade-off between minimizing error and maximizing margin, ϵ is the margin of tolerance, and (x_i, y_i) are the input-output pairs in the training set. The function $f(x)$ is determined by the dot product between the input features and the weights, expressed as $f(x) = \langle w, x\rangle + b$, where b is the bias term.

SVR optimizes the hyperplane by solving a constrained optimization problem, aiming to minimize errors while balancing data fitting with margin maximization. The final model is heavily influenced by the support vectors, which are the data points closest to the hyperplane. SVR is particularly adept at capturing non-linear relationships using kernel functions, which map input features into a higher-dimensional space where a linear hyperplane can be more effectively applied. This ability to handle non-linear data patterns makes SVR a versatile and powerful tool for regression tasks across various fields and applications (11, 15, 27).

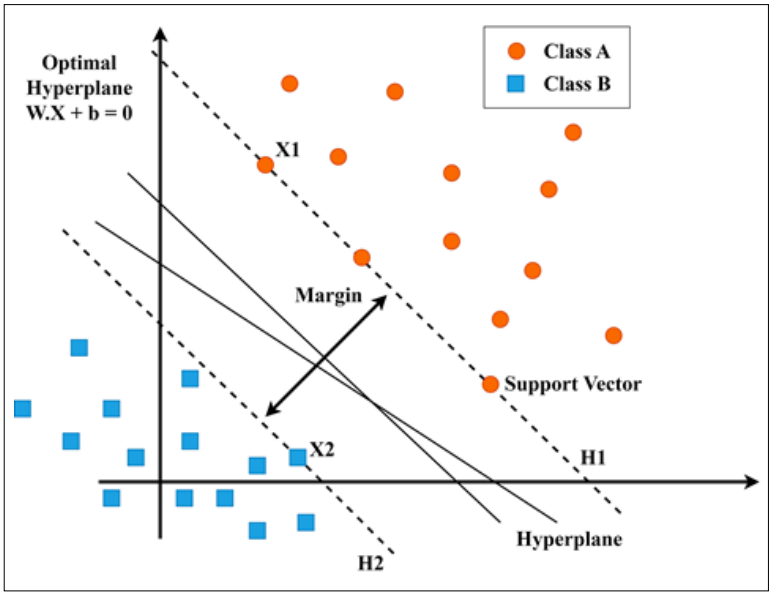

Fig 4: Architecture of SVM

2.4.5 Random Forest (RF)

Random Forest (RF) is a popular regression technique widely used in data analysis for its ability to create an ensemble model by leveraging the predictive power of multiple decision trees. Unlike traditional regression methods, RF constructs several decision trees based on an input vector (x) that includes various relevant features from the training data. The ensemble model is created by generating K regression trees and averaging their predictions. The RF regression predictor $\left(\hat{f}_k(x)\right)$ for the input vector x is computed as follows:

$$\hat{f}_k(x) = \frac{1}{K}\sum_{k=1}^{K} T(x) \qquad (13)$$

Here, $T(x)$ represents the individual regression trees developed by RF. To enhance diversity among these trees and reduce correlation, RF employs a technique known as bagging. In bagging, subsets of the training data are generated by randomly resampling the original dataset with replacement. This process involves selecting data points from the input sample to create subsets $\{h(x, \Theta_k), k = 1, \ldots, K\}$, where $\{\Theta_k\}$ are independent random vectors with the same distribution. Some data points may be repeated, while others might be excluded, which improves stability and prediction accuracy, especially when faced with slight variations in input data (11,15,28).

A significant advantage of RF is its ability to select the optimal feature or split point from a randomly chosen subset of features for each tree. This method reduces correlation between trees and helps minimize generalization errors. RF trees are grown without pruning, maintaining computational efficiency. Moreover, RF uses out-of-bag samples to evaluate model performance, eliminating the need for a separate test dataset. As the number of trees in the forest increases, the generalization error tends to stabilize, reducing the risk of overfitting. Additionally, RF provides valuable insights into the importance of various features, making it a reliable and powerful tool for accurate predictions in regression tasks.

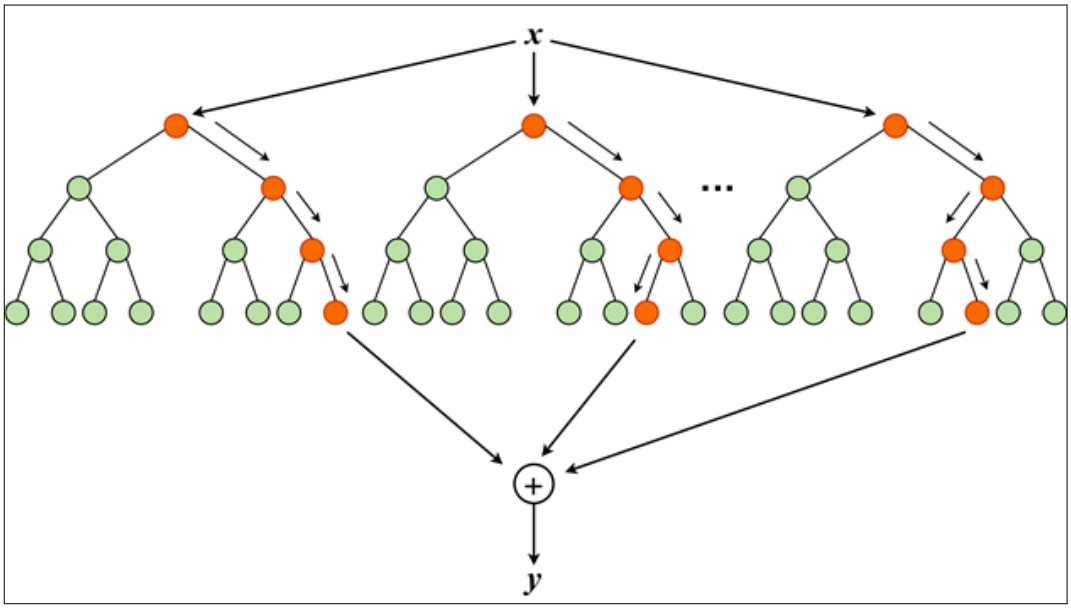

Fig 5: Architecture of Random Forest

2.4.6 Extreme Gradient Boosting (XGBoost)

XGBoost is a highly effective machine learning algorithm widely used for regression tasks, known for its ability to enhance predictive accuracy through a gradient boosting framework. Unlike traditional methods, XGBoost builds multiple decision trees sequentially, each one improving upon the predictions of the previous trees. The algorithm aims to minimize an objective function that balances model fit and complexity by combining a loss function with a regularization term. The objective function for XGBoost regression is given by:

$$Objective = \sum_{i=1}^{n}\left(\frac{1}{2}.(y_i - \hat{y}_i)^2 + \lambda.\Omega(f)\right) \quad (14)$$

In this equation, y_i represents the actual target value, $\hat{y}_i$ is the predicted value, and n is the number of data points. The term $\Omega(f)$ represents the regularization function, with λ controlling the strength of regularization.

XGBoost's effectiveness comes from its iterative approach, starting with an initial prediction $\hat{y}_i^{(0)}$ and updating it in each iteration by adding the output from a new decision tree:

$$\hat{y}_i^{(t)} = \hat{y}_i^{(t-1)} + f_t(x_i) \quad (15)$$

Here, t represents the current iteration, $f_t(x_i)$ is the prediction from the t-th tree for the input x_i, and $\hat{y}_i^{(t)}$ is the updated prediction.

XGBoost enhances the accuracy of its trees by optimizing their structure, selecting the best split points based on the gradient of the loss function. It calculates the first-order and second-order gradients for each data point, using these gradients to identify optimal splits. Additionally, XGBoost includes a regularization term that controls the complexity of individual trees, helping to prevent overfitting and improve generalization. By iteratively refining the predictions of multiple trees, XGBoost produces highly accurate regression models, making it an outstanding tool for a wide range of data analysis tasks (15, 29).

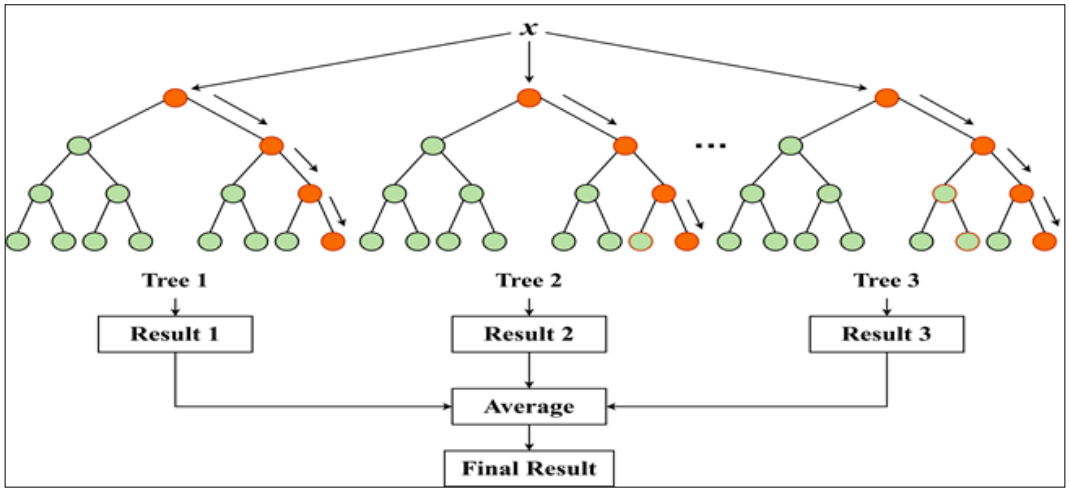

Fig 6: Architecture of XGBoost

2.5 Study Methodology

Creating forecasting models comprises five essential stages: Data Collection, Data Pre-processing, Model Compilation, Model Training, and Model Evaluation. The methodology adopted in this study is depicted through Figure 7. The subsequent section elaborates on these phases in detail.

2.6 Test for Stationarity

An essential element of time-series analysis is determining whether the data is stationary, meaning that the series has a consistent mean and variance over time. To assess this characteristic, the Augmented Dickey-Fuller (ADF) test was used (30). The findings, as shown in Table 1, offer definitive proof regarding the stationarity of the series.

Table 1: Unit root test results of top prices

Data	ADF test			Remarks
	Statistic	P-Value	Lags	
Paddy	-4.09	0.0003	2	Stationary

2.7 Test for nonlinearity

The Brock-Dechert-Scheinkman (BDS) test, a nonparametric technique, was used to assess the presence of nonlinearity in the data series. According to the results presented in Table 2, the probability values computed within the range of 0.5σ to 2.0σ strongly indicate nonlinearity in the series, particularly for embedding dimensions 2 and 3.

Table 2: BDS test results of paddy

Epsilon	Embedding dimensions		P-Value	Remarks
	2	3		
0.5σ	251.52	1169.84	< 0.0001	Nonlinear
1.0σ	242.41	804.17	< 0.0001	Nonlinear
1.5σ	207.58	349.25	< 0.0001	Nonlinear
2.0σ	194.75	351.23	< 0.0001	Nonlinear
1.0σ	231.85	549.75	< 0.0001	Nonlinear
1.5σ	209.38	547.64	< 0.0001	Nonlinear
2.0σ	239.48	512.77	< 0.0001	Nonlinear

However, the machine learning models used to analyse agricultural time series data are free from assumptions and excel at efficiently extracting relevant information from time-dependent data.

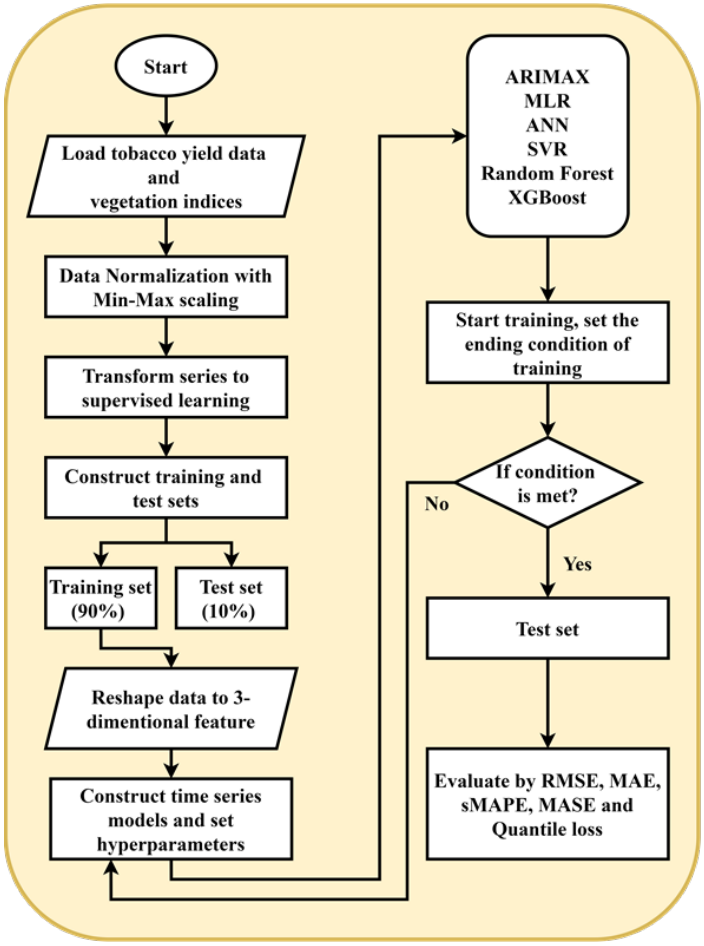

Fig 7: Procedural flowchart of the methodology

2.8 Data Pre-processing

Data pre-processing is vital for converting raw data into a format suitable for effective analysis. Through the application of data mining techniques, pre-processing improves the usability and effectiveness of the data, ensuring its quality and reliability for further analysis.

2.8.1 Model building

The development of forecasting models involves two main stages: model training and hyperparameter tuning.

2.8.2 Model Training

The datasets are divided into two segments: a training set and a test set, using a 90:10 split, respectively. Following this division, the data values are normalized to a range between 0 and 1 while maintaining their original distribution. This normalization is achieved using the following equation:

$$X_i' = \frac{X_i - X_{min}}{X_{max} - X_{min}} \quad (16)$$

In this equation, X_{min} and X_{max} represent the minimum and maximum values, respectively, while X_i is the observed value at time *i*, and X_i' is the rescaled value. The training set, comprising 90 percent of the data, captures the historical information, whereas the test set, containing the remaining 10 percent, consists of sequential data points corresponding to the forecasted days. This approach enables effective model training and evaluation.

During training, the target variable (paddy yield) is analyzed alongside exogenous variables such as precipitation, temperature, and their combined effects. Including these additional factors aims to enhance the model's forecasting performance. The model is fine-tuned by training on these datasets, accounting for the interactions between all variables. This method ensures a comprehensive analysis and robust evaluation of the forecasting models.

Likewise, parameters have been finetuned using the Randomized Search CV tool in python, which returned the best parameters that have been used to train the models so that they can provide good prediction results. To ensure the reliability of the results from the training, k-fold cross validation was used. In our study, we employed six forecasting models for predicting paddy yield, namely ARIMAX, MLR, ANN, SVR, RFR, and XGBoost, for each of the VIs.

2.8.3 Model Evaluation

The models were evaluated on the test dataset, comprising the last 10 percent of the complete dataset. Evaluation metrics included Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Symmetrix Mean Absolute Percentage Error (sMAPE), Mean Absolute Scaled Error (MASE) and Quantile Loss (QL).

a) Root Mean Squared Error (RMSE)

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2} \quad (17)$$

b) Mean Absolute Error (MAE)

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i| \quad (18)$$

c) Symmetric Mean Absolute Percentage Error (sMAPE)

$$sMAPE = \frac{1}{N}\sum_{i=1}^{N}\frac{|y_i - \hat{y}_i|}{(|y_i| + |\hat{y}_i|)/2} * 100 \quad (19)$$

d) Mean Absolute Scaled Error (MASE)

$$MASE = \frac{MAE}{MAE_{naive}} \quad (20)$$

e) Quantile Loss (QL)

$$QL_q = \frac{1}{N}\sum_{i=1}^{N}\left(\rho_q(y_i - \hat{y}_i)\right) \quad (21)$$

Where, $q = 0.5$ and ρ_q is the quantile loss function

$$\rho_q(e) = \begin{cases} q.e & ,if\ e > 0 \\ (q-1).e & ,if\ e \leq 0 \end{cases} \quad (22)$$

Where, y_i is the true values of the variable being predicted, $\hat{y}$ is the predicted values of the variable and N is the number of observations in the dataset.

3. Results and Discussion

Across various vegetation indices like NDVI, GNDVI, SAVI, MSAVI, LAI and LSWI, XGBoost consistently emerged as the top-performing model. It outperformed other models like ARIMAX, MLR, ANN, SVR, and RFR, achieving the lowest RMSE, MAE, and superior performance in sMAPE, MASE, and QL metrics. For instance, with the GNDVI index, XGBoost achieved an RMSE of 50.85, MAE of 42.1, sMAPE of 12.12, MASE of 1.086, and QL of 20.05. This trend was consistent across all indices, making XGBoost the most accurate and reliable model for these analyses, as shown in the Table 3.

Table 3: Comparative performance metrics of forecasting models for paddy yield using vegetation indices as exogenous variables

Vegetation indices	Models	RMSE	MAE	sMAPE	MASE	QL
NDVI	ARIMAX	60.5	48.75	15.85	1.612	23.38
	MLR	63.15	51.35	16.09	1.654	25.68
	ANN	58.2	47.2	14.21	1.573	22.60
	SVR	56.8	46.1	13.54	1.512	23.05
	RFR	54.25	44.5	12.8	1.485	21.25
	XGBoost	51.1	42.6	12.2	1.104	21.01
GNDVI	ARIMAX	58.8	48.25	15.55	1.58	24.13
	MLR	60.45	48.95	15.97	1.645	24.48
	ANN	56.35	46.85	14.07	1.552	23.43
	SVR	54.9	45.6	13.22	1.484	22.80
	RFR	52.7	44	12.51	1.435	21.90
	XGBoost	50.85	42.1	12.12	1.086	20.05
SAVI	ARIMAX	61.85	51.5	15.78	1.643	25.75
	MLR	63.4	54.95	16.9	1.654	27.48
	ANN	59.95	48.1	15.5	1.644	24.05
	SVR	58.5	47.25	14.8	1.575	23.63
	RFR	56.1	45.75	13.5	1.516	21.88
	XGBoost	54.3	44.5	13.06	1.133	21.25
MSAVI	ARIMAX	60.9	49.15	15.9	1.621	24.66
	MLR	62.55	49.65	16.01	1.654	23.83
	ANN	59.25	47.75	15.33	1.597	23.68
	SVR	57.8	46.9	14.75	1.542	23.45
	RFR	55.15	45.25	13.63	1.511	23.63
	XGBoost	53.5	44	13.54	1.147	21.09
LAI	ARIMAX	59.2	48	15.41	1.643	24.10
	MLR	61.1	48.55	15.89	1.642	24.28
	ANN	57.3	46.4	14.63	1.584	23.20
	SVR	55.95	45.5	13.96	1.527	22.75
	RFR	53.7	43.9	13.17	1.475	21.95
	XGBoost	51.85	42.45	12.33	1.098	21.23
LSWI	ARIMAX	60.1	48.4	15.6	1.613	25.20
	MLR	61.75	49.05	15.93	1.645	25.53
	ANN	58.85	47	14.5	1.594	23.50
	SVR	57.25	46.25	13.76	1.556	23.13
	RFR	54.75	44.75	13.05	1.509	22.38
	XGBoost	53.05	43.25	12.51	1.115	21.43

While the ARIMAX model can capture temporal patterns, it struggles with modelling complex nonlinear patterns. Statistical models, in general, are often constrained by stringent assumptions that may not always hold true in real-world scenarios. In contrast, machine learning (ML) models such as ANN, SVM, RFR, and XGBoost are increasingly preferred due

to their data-driven approach and ability to capture nonlinear relationships. XGBoost, in particular, offers several advantages that make it a powerful tool in machine learning and data analysis. Its primary strengths lie in its efficiency with large datasets and high accuracy in predictions. XGBoost excels at capturing complex non-linear patterns through its ensemble learning approach, which combines multiple weak learners into a strong predictive model. Additionally, it includes built-in mechanisms to prevent overfitting, such as regularization and tree pruning, making it robust across diverse datasets. The model's flexibility in handling different data types and its scalability to large datasets further enhance its utility across various applications, including time series forecasting, classification, and regression tasks. These attributes collectively position XGBoost as a top choice for predictive modelling tasks, especially in scenarios where precision and performance are critical.

4. Conclusion

In conclusion, the evaluation of paddy yield prediction using various vegetation indices reveals that XGBoost consistently outperforms other forecasting models, including ARIMAX, MLR, ANN, SVR, and RFR. This is evidenced by XGBoost's superior performance, demonstrated by its lowest RMSE, MAE, and top scores in sMAPE, MASE, and QL metrics across all vegetation indices (NDVI, GNDVI, SAVI, MSAVI, LAI, LSWI). XGBoost's effectiveness can be attributed to its capacity to handle large datasets, capture intricate non-linear patterns, and mitigate overfitting through its ensemble learning approach and built-in regularization features. The model's robustness and adaptability make it a highly efficient tool for predictive modelling in paddy yield forecasting. This research highlights the benefits of employing advanced machine learning models such as XGBoost in agricultural yield prediction, demonstrating their ability to manage diverse datasets and provide accurate forecasts. Consequently, these findings offer significant insights for enhancing predictive accuracy in agricultural economics and lay a solid groundwork for future research in this domain.

5. Acknowledgements

The authors would like to acknowledge Petlurivaripalem village agricultural assistant (VAA) for providing the ground truth data of paddy fields.

6. Author contributions

All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by KVC, BSN, V, SSH, ASBS and RC. The first draft of the manuscript was written by KVC, BSN, V, RC, ASBS and SSH commented on its improvement. All authors read and approved the final manuscript.

7. Funding

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

8. Availability of data and material

In this study paddy yield data in time series from 2015-2023 was taken from Petlurivaripalem village Rythu Bharosa Kendram (RBK). Data will be available based on the request.

9. Code availability

Code will be available on request to the corresponding author.

10. Declarations

Conflict of interest

The authors have no affiliation with any organization with a direct or indirect financial interest in the subject matter discussed in the manuscript.

11. Ethics approval, consent to participate, and consent for publication

The manuscript does not report on or involve the use of any animal or human data and "not applicable" in this section.

12. References

1. Kavita M, Mathur P. Crop yield estimation in India using machine learning. In: 2020 IEEE 5th International Conference on Computing Communication and Automation (ICCCA). IEEE; c2020, p. 220-224.
2. Jabbar ATS, Albayati MMA, Ziboon ART. Environmental factors and wheat crops yield estimation in multiple scales using different remote sensing data/(Al-Zubaydiyah district) as a case study. In: AIP Conference Proceedings. AIP Publishing; c2023.
3. Bali N, Singla A. Emerging trends in machine learning to predict crop yield and study its influential factors: A survey. Arch Comput Methods Eng. 2022;29(1):95-112.
4. Lillesand T, Kiefer RW, Chipman J. Remote sensing and image interpretation. John Wiley & Sons; 2015.
5. Doraiswamy PC, Hatfield JL, Jackson TJ, Akhmedov B, Prueger J, Stern A. Crop condition and yield simulations using Landsat and MODIS. Remote Sens Environ. 2004;92(4):548-559.
6. Sishodia RP, Ray RL, Singh SK. Applications of remote sensing in precision agriculture: A review. Remote Sens. 2020;12(19):3136.
7. Jiang Y, Shao X, Li L, Wang T, Zhao H, Hou Q, *et al*. Remote Sensing Monitoring Model of Tobacco Growth and Yield Based on Ecological Process and Carbon Cycle. J Biobased Mater Bioenergy. 2023;17(2):211-224.
8. Makridakis S, Spiliotis E, Assimakopoulos V. Statistical and machine learning forecasting methods: Concerns and ways forward. PLoS One. 2018;13(3).
9. Anggraeni W, Andri KB, Mahananto F. The performance of ARIMAX model and Vector Autoregressive (VAR) model in forecasting strategic commodity price in Indonesia. Procedia Comput Sci. 2017;124:189-196.
10. Avinash G, Ramasubramanian V, Ray M, Paul RK, Godara S, Nayak GHH, *et al*. Hidden Markov guided deep learning models for forecasting highly volatile agricultural commodity prices. Appl Soft Comput. 2024;158:111557.
11. Naik S, GH HN, Rao SG. Prediction of wheat yield by using UAV RGB drone imagery and advanced machine learning techniques; c2023.
12. Hamjah MA, Chowdhury MAK. Measuring climatic and hydrological effects on cash crop production and production forecasting in Bangladesh using ARIMAX model. Math Theory Model. 2014;4(6):138-152.
13. HT V, MS J, Avinash G, GH HN. A comparative analysis of time series models for onion price forecasting: Insights for agricultural economics. J Exp Agric Int. 2024;46(5):146-154.
14. Amaratunga V, Wickramasinghe L, Perera A, Jayasinghe J, Rathnayake U. Artificial neural network to estimate the paddy yield prediction using climatic data. Math Probl Eng. 2020;2020:8627824.
15. Nayak GHH, Alam MW, Singh KN, Avinash G, Kumar

RR, Ray M, *et al*. Exogenous variable driven deep learning models for improved price forecasting of TOP crops in India. Sci Rep. 2024;14(1):17203.
16. Rouse JW, Haas RH, Schell JA, Deering DW. Monitoring vegetation systems in the Great Plains with ERTS. NASA Spec Publ. 1974;351(1):309.
17. Shaver T, Khosla R, Westfall D. Utilizing green normalized difference vegetation indices (GNDVI) for production level management zone delineation in irrigated corn. In: The 18th World Congress of Soil Science; c2006.
18. Major DJ, Baret F, Guyot G. A ratio vegetation index adjusted for soil brightness. Int J Remote Sens. 1990;11(5):727-740.
19. Qi J, Chehbouni A, Huete AR, Kerr YH, Sorooshian S. A modified soil adjusted vegetation index. Remote Sens Environ. 1994;48(2):119-126.
20. Zheng G, Moskal LM. Retrieving leaf area index (LAI) using remote sensing: theories, methods and sensors. Sensors. 2009;9(4):2719-2745.
21. Xu X, Konings AG, Longo M, Feldman A, Xu L, Saatchi S, *et al*. Leaf surface water, not plant water stress, drives diurnal variation in tropical forest canopy water content. New Phytol. 2021;231(1):122-136.
22. Zhang C, Pattey E, Liu J, Cai H, Shang J, Dong T. Retrieving leaf and canopy water content of winter wheat using vegetation water indices. IEEE J Sel Top Appl Earth Obs Remote Sens. 2017;11(1):112-126.
23. Vennila S, Singh G, Jha GK, Rao MS, Panwar H, Hegde M. Artificial neural network techniques for predicting severity of Spodoptera litura (Fabricius) on groundnut. J Environ Biol. 2017;38(3):449.
24. Gopal PSM, Bhargavi R. A novel approach for efficient crop yield prediction. Comput Electron Agric. 2019;165:104968.
25. Panapakidis IP, Dagoumas AS. Day-ahead electricity price forecasting via the application of artificial neural network-based models. Appl Energy. 2016;172:132-151.
26. Belouz K, Nourani A, Zereg S, Bencheikh A. Prediction of greenhouse tomato yield using artificial neural networks combined with sensitivity analysis. Sci Hortic. 2022;293:110666.
27. Vapnik V, Chapelle O. Bounds on error expectation for support vector machines. Neural Comput. 2000;12(9):2013-2036.
28. Suparwito H, Polina AM. Prediction of tobacco leaf grades with ensemble machine learning methods. In: 2019 International Congress on Applied Information Technology (AIT). IEEE; c2019, p. 1-6.
29. Ribeiro MHDM, dos Santos Coelho L. Ensemble approach based on bagging, boosting and stacking for short-term prediction in agribusiness time series. Appl Soft Comput. 2020;86:105837.
30. Kundu MG, Mishra S, Khare D. Specificity and sensitivity of normality tests. In: Proceedings of VI International Symposium on Optimisation and Statistics. Anamaya Publisher; c2011.