Hindawi
Complexity
Volume 2022, Article ID 5305353, 20 pages
https://doi.org/10.1155/2022/5305353

Research Article

Yield Response of Different Rice Ecotypes to Meteorological, Agro-Chemical, and Soil Physiographic Factors for Interpretable Precision Agriculture Using Extreme Gradient Boosting and Support Vector Regression

Md. Sabbir Ahmed,[1] **Md. Tasin Tazwar**,[1] **Haseen Khan**,[1] **Swadhin Roy**,[1] **Junaed Iqbal**,[1] **Md. Golam Rabiul Alam**,[1] **Md. Rafiul Hassan**,[2] **and Mohammad Mehedi Hassan**[3]

[1]*Department of Computer Science and Engineering, BRAC University, 66 Mohakhali, Dhaka, Bangladesh*
[2]*College of Arts and Sciences, University of Maine at Presque Isle, ME 04769, USA*
[3]*Information Systems Department, College of Computer and Information Sciences King Saud University, Riyadh 11543, Saudi Arabia*

Correspondence should be addressed to Md. Sabbir Ahmed; msahmeds1997@gmail.com

Received 17 June 2022; Accepted 29 July 2022; Published 19 September 2022

Academic Editor: Giacomo Fiumara

The food security of more than half of the world's population depends on rice production which is one of the key objectives of precision agriculture. The traditional rice almanac used astronomical and climate factors to estimate yield response. However, this research integrated meteorological, agro-chemical, and soil physiographic factors for yield response prediction. Besides, the impact of those factors on the production of three major rice ecotypes has also been studied in this research. Moreover, this study found a different set of those factors with respect to the yield response of different rice ecotypes. Machine learning algorithms named Extreme Gradient Boosting (XGBoost) and Support Vector Regression (SVR) have been used for predicting the yield response. The SVR shows better results than XGBoost for predicting the yield of the Aus rice ecotype, whereas XGBoost performs better for forecasting the yield of the Aman and Boro rice ecotypes. The result shows that the root mean squared error (RMSE) of three different ecotypes are in between 9.38% and 24.37% and that of R-squared values are between 89.74% and 99.13% on two different machine learning algorithms. Moreover, the explainability of the models is also shown in this study with the help of the explainable artificial intelligence (XAI) model called Local Interpretable Model-Agnostic Explanations (LIME).

1. Introduction

More than 50% of the people in the world consume rice as their main food item and more than 3500 million people rely on rice as it fulfills more than 20% of their day-to-day calories [1]. Since the world's population has grown over time, the amount of cultivable land has reduced day by day, so it is important to grow more crops on a limited amount of land. Therefore, it has been an essential thing to predict the yield of the harvest before sowing the crop. Smart agriculture has been introduced in Agriculture 4.0 [2] for intelligent development of Argo-production, processes, and services [2]. In research [3], thermal infrared (TIR) images of soil have been analyzed using a convolutional neural network for precision agriculture. Machine learning-based soil type identification and crop recommendation system have been proposed in another research [4]. The bagged trees, Weighted k-nearest neighbor (k-NN), and Gaussian kernel-based support vector machines (SVM) have been used for soil type detection and crop recommendation. However, yield response prediction for precision agriculture is such a modern approach in the farming sector that can ensure food security through escalating production and, at the same time, boost the economy of a country.

Kaur and Singh Attwal studied the effect of temperature and rainfall on the yield estimation of paddy [5]. The study used a predictive Apriori algorithm to predict paddy yield based on daily rainfall and daily temperature. Another study looked into the impact of rising air temperatures on yield by applying the Ceres-rice model associated with the Decision Support System for Agrotechnology Transfer (DSSAT) [6]. Various machine learning techniques were used to predict the relationship between rice yield and climate variables in a large region of Sri Lanka by the authors in Ref. [7]. In another study, artificial neural networks were used to estimate paddy yield prediction using only climate data [8]. In Ref. [9], Shook et al. (2020) presented a deep learning-based crop production forecast that took into account genotype and meteorological data. Guruprasad et al. proposed a yield estimation model for paddy based on the different spatial resolution (SR), weather, and soil factors [10]. Disaggregation had been performed through machine learning models using coarser SR level data to predict yields at the finer SR level. Another research [11] shows that Backward interval Partial Least Square Regression (BiPLS) was used in the yield prediction of rice. The BiPLS showed higher prediction accuracy on airborne hyperspectral data. In another research [12], paddy harvest and demand are predicted using recurrent neural network (RNN) and long short-term memory (LSTM), respectively. The corelation between paddy plantation date and rice productivity has been studied in Ref. [13]. In Ref. [14], the authors discussed a deep learning-based rice yield prediction model for selecting the best linear and nonlinear regression models for yield prediction.

The above-mentioned studies had considered various factors separately, e.g., rainfall, temperature, humidity, and soil types for rice yield prediction. However, we have to integrate the meteorological, agro-chemical, and soil physiographic factors for the accurate prediction of rice yields. Therefore, this research combinedly analyzed the meteorological factors such as temperature, rainfall, and humidity; the agro-chemical factors such as the used amount of urea, TSP (trisodium phosphate), and DAP (di-ammonium phosphate) fertilizers; the soil physiographic factors such as highland, low land, grey valley soil, brown hill soil, and soil moisture to predict the rice yield response. The Extreme Gradient Boosting (XGBoost) and support vector regression (SVR) are used for predicting the yield response of rice. Furthermore, rice productivity varies greatly depending on the rice ecotype. Therefore, this research separately analyzed the factors contributing to the rice production of three major rice ecotypes, namely, Aus, Aman, and Boro. The Aus, Aman, and Boro ecotypes are cultivated in July-August, December-January, and March-May seasons, respectively.

Finally, we have added "explainability," or in other words, "interpretability," to the ML models we have used in this study by using the XAI model. The terms "explainability" and "interpretability" are being used by the researchers as synonyms [15, 16]. In our work, we have specifically used Local Interpretable Model-agnostic Explanations (LIME) [17] to interpret our models. A user may study and comprehend how inputs are mathematically translated to outputs in an interpretable system [18]. The problem of demystifying the black boxes is at the heart of XAI models, which also implies responsible AI since it may aid in the creation of transparent models [19]. We demonstrated how different subsets of features are effective in forecasting yield for all three ecotypes and how certain features are more significant than others for each ecotype by utilizing LIME for both XGBoost and SVR.

Specifically, the key contributions of the paper are summarized as follows:

- We introduced a system capable of predicting the yield of Aus, Aman, and Boro rice ecotypes in the same land in different years of different land areas
- We formulate and integrate meteorological, agro-chemical, and soil physiographic factors for yield prediction
- We apply two different machine learning models to show the comparison while predicting the yields of the rice ecotypes
- We show the explainability of the two ML models for each rice ecotype by applying an XAI model called LIME

The rest of the paper is organized as follows: The related work is discussed in Section 2, methodology is covered in Section 3, result analysis is discussed in Section 4, and discussion and conclusion are covered in Sections 5 and 6, respectively.

2. Related Work

Previously, some research studies were conducted related to yield prediction and improving paddy production. A case study was conducted in the Indian region using machine learning technologies [10]. At different spatial resolution (SR) levels in India, they showed a yield estimation model for paddy based on weather and soil-related data. Their accuracy of the yield estimation model was provided in their research using numerous sets of features and different techniques of machine learning. They have included several spatial resolution (SR) levels in this experiment, like coarser SR level (district level) and finer SR level (taluk level). Moreover, they have given a vast analysis of their system based on the accuracy of the result they got. They have also used a number of machine learning (ML) techniques in their experiments. The accuracy results of these ML techniques have also been discussed in this paper. Furthermore, they have applied disaggregation on the coarser SR level yield data. The disaggregation has been performed by applying the machine learning trained models using coarser SR level data to predict yields at the finer SR level. For predicting yields of paddy they used machine learning algorithms like support vector machine (SVM), random forest (RF), and neural network (NN). They showed bar charts over weekly and monthly data and analyzed their accuracy. Among 210 bars, only 16 percent bars had an error more than 10 percent. 117 bars (equivalent to 56 percent of total bars) had error of less than 5 percent. The remaining 28 percent bars had errors

ranging from 5 percent to 10 percent. The taluk (an area of a district in India) yield data of disaggregation of a district had a minimum error of 6 percent and a maximum error of 25 percent.

Another research group used statistical techniques for getting a prediction model of the yield of rice [11]. They worked based on Indonesian fields and conducted research using airborne hyperspectral data and yield data. Backward interval Partial Least Square Regression (BiPLS) was used to estimate the yields of rice. They showed that BiPLS application to hyperspectral data gave a good accurate prediction. In their training dataset of 2011, the root mean square error (RMSE) was 76.1 percent after the first derivation and 69.7 percent after the second derivation. The accuracy with reflectance data performed better than the first and second derivation. They suggested that their proposed method can be used by the Indonesian government for getting the yield estimation of paddy.

Some authors worked on the prediction of paddy harvest and rice demand [12]. The authors tried to show a smart approach for paddy with two steps. First, they established a prediction model for paddy harvest prediction. Second, the rice demand prediction model was established in their research. They have mentioned the significance of keeping a balance between paddy cultivation and consumer demand for it. They have also figured out some aspects which the paddy crop yield and the demand for rice in a certain area depend upon. According to them, these factors are rainfall, humidity, the lifestyle of the people of that area, and so on. Judging the factors, they have found that it is a difficult task to predict the yield of the paddy and its demand of it. From this paper, we came to know that their goal was to prepare a module that can predict the paddy harvest and the rice demand of a certain area. In this regard, they have chosen two specific machine learning techniques, i.e., recurrent neural network (RNN) and long short-term memory (LSTM) were used to establish the prediction models. The performance was measured based on the Sri Lankan context. They analyzed the performance using mean square error (MSE), root mean square error (RMSE), and the training and testing score of their prediction model. The harvest prediction model showed 0.04 MSE value for training data and 0.11 MSE for testing data. The rice demand prediction model showed 0.17 MSE value for training data whereas 0.34 for testing data.

In Ref. [20], they focused on expanding algorithms for precisely mapping paddy and estimating rice yield in the Ampara area as the district has the second largest paddy production in the Sri Lankan rice harvest. From April through September 2019, the researchers analyzed images from the Sentinel-1 and Sentinel-2 satellites. Maximum likelihood classification and Divisional Secretory Division (DSD) are the two classification techniques utilized to determine the actual paddy area. Using Sentinel-2 feature extracts and ground truth data, an artificial neural network (ANN) model was utilized to forecast paddy rice harvest. These categories had accuracy ratings of 0.92 and 0.86, respectively. The model's mean square error is 0.106, and its mean absolute error is 0.245.

Alam et al.'s aim was to build a rice yield estimation model for Bangladesh [21]. From 2011 to 2016, images from the Moderate Resolution Imaging Spectroradiometer (MODIS) and Normalized Difference Vegetation Index (NDVI) were utilized. To begin with, the rice field area is estimated using NDVI threshold values. To calculate the NDVI threshold values, an improved method was used. To estimate total Boro output in each of Bangladesh's eight districts, four regression models are used: linear, ridge, lasso, and decision tree. For the Barisal, Chittagong, Dhaka, Khulna, Mymensingh, Rajshahi, Rangpur, and Sylhet divisions, the highest R^2 values are 0.492, 0.790, 0.899, 0.891, 0.848, 0.942, 0.777, and 0.848, respectively.

Alebele et al. proposed a Gaussian kernel for the estimation of rice yield by using SAR and optical imagery where ground truth data are used in a limited amount [22]. The primary goal was to study the synergistic use of Sentinel-2 statistical parameters and Sentinel-1 interferometric coherence data for predicting rice grain production using Gaussian kernel regression for the prediction of accuracy was assessed using in situ measured yield data collected in 2019 and 2020 over Xinghua county in Jiangsu Province, China. The Gaussian kernel regression outperformed the probabilistic Gaussian regression and Bayesian linear inference in every aspect. If the RDVI1 and interferometric coherence are combined at the heading stage (RMSE = 0.55 t/ha and $r^2 = 0.81$), the highest prediction accuracy can be achieved. However, the optical red edge difference vegetation index (RMSE = 0.61 t/ha and $r^2 = 0.65$) shows better prediction than with the interferometric coherence (RMSE = 0.79 t/ha and $r^2 = 0.52$).

In Ref. [23], for forecasting rice production in different locations of Bangladesh, a method called Weather-based Prediction System for Rice Yield (WPSRY) is provided. This suggested method begins by developing a model for the prediction of weather parameters where neural networks (NN) will be utilized, and then rice yield will be calculated by using Support Vector Regression (SVR), which takes as inputs anticipated weather from NN as well as existing agricultural data. The simulation results show that the WPSRY technique offers promising prediction accuracy.

In Ref. [24], using historical yield data from 1981 to 2010, a statistical technique was created for six distinct locations in China. Based on 28 ensemble weather forecasts from six cutting-edge models, the data collected from the National Bureau of Statistics were combined with meteorological station observations for checking the effect of different climates such as precipitation and temperature changes on grain yields into the 2030s. Model results from the Coupled Model Intercomparison Project Phase 5 (CMIP5). The findings show that maize, rice, soybean, and wheat respond similarly to climatic differences across China with susceptibility to increasing heat from north to south and from inland to coastal locations. In Central-South China and the East, the yield of every type of crop indicated here is very strongly correlated with weather changes. Future projections based on a moderate emission reduction scenario (RCP4.5) found that the yield of the four types of crops in six locations in China will increase in the 2030s where the range will vary

from 0.02–1.19 hundred ton/ha compared to that in the 2000s.

Gandhi et al. intended to use neural networks to predict the output of rice production and investigate the factors impacting rice crop yield in numerous locations in India's Maharashtra state [25]. The statistics for 27 districts in the Indian state of Maharashtra were taken from publicly available Indian Government sources. Precipitation, lowest temperature, average temperature, maximum temperature, and reference crop evapotranspiration, area, production, and yield were the factors studied for the current study from 1998 to 2002 during the Kharif season which means from June to November. The WEKA tool was used to analyze the dataset. To validate the data, a multilayer perceptron neural network was created using the cross-validation approach. The results revealed 97.5 percent accuracy, 96.3 percent sensitivity, and 98.1 percent specificity.

The paper [26] is the result of a thorough investigation of the usage of machine learning technologies in yield prediction. The authors compare and contrast the use of neural networks against machine learning techniques in this work. Another study used a survey of developing advances in machine learning to estimate crop productivity [27]. They investigated a variety of machine learning approaches as well as advanced techniques such as deep learning for crop yield prediction. They also looked into the effectiveness of hybridized models created by combining multiple techniques. The authors of this study [28] conducted a performance analysis on machine learning techniques for crop yield prediction using a deep learning model. For yield prediction, they employed two separate datasets and used random forest and multilayer perceptron neural network regression models. In a separate study, the authors use machine learning and deep learning to integrate multisource data for rice yield forecasts across China [29]. With publicly available multisource data, they developed a scalable, simple, and low-cost technique for estimating rice output across a vast area in real time. Islam et al. used both parametric and nonparametric techniques to create boro rice yield prediction models utilizing satellite remote sensing-based vegetation indicators at the optimum harvesting period [30]. They discovered that by combining vegetation indices and ground reference mean yield data, more than 70% of the study area can be effectively predicted. Another study employed MODIS-based factors and meteorological data to forecast crop yields at the county level [31]. The researchers wanted to see how well machine learning models predicted crop yields. Considering all of the facts, they came to the conclusion that SVM is an appropriate method for crop yield prediction.

3. Methodology

The proposed yield prediction system is an effective combination of a two-step feature selection strategy, ML regression models, and the explainability of those ML models shown by the XAI model called LIME. It comprises five distinct phases: (1) general data preprocessing with first step of feature selection, (2) formulation of training/testing data, (3) second step of feature selection using RFE, and (4) finding RMSE and R-squared values applying ML-based regression models, and finally, (5) using XAI model called LIME for explainability of the ML models. Figure 1 represents the system overview of the proposed yield prediction system.

In the proposed yield prediction system, a dataset with 43-features (explained in Section 3.1) has been used. After general data preprocessing, the first feature selection has been applied. XGBoost and SVR are being used for regression after the second feature selection using RFE (details in Sections 3.2.2). In this work, it has been shown that a carefully selected combination of features and proper ML-based regression models can be very effective for yield prediction. An XAI model called LIME (discussed in Section 3.4) is used to show the explainability of both the ML models.

3.1. Dataset Description and Preprocessing. This paper uses data from an agricultural dataset recorded during 2008–2017 season found on kaggle.com [32]. The dataset contains 43 features including meteorological factors, agro-chemical factors, and soil physiographic factors. In total, there are 70 rows and 44 columns present in the tabular format.

The dataset includes the geographical location and year-by-year production of Aus, Aman, Boro, Wheat, Potato, and Jute, as well as certain meteorological factors that influence agricultural product output, such as maximum and minimum temperatures, average rainfall, humidity, and storm. This dataset also includes some argo-chemical factors such as Urea, TSP (triple super phosphate), and MAP (mono-ammonium phosphate), and DAP (di-Ammonium Phosphate), as well as various soil physiographic factors such as various types of high, low, and very low land, miscellaneous land, different alluvium, acid basin clay, and different kinds of soil, peat, and soil moisture. In Table 1, all the 43 features' names have been listed sequentially.

We have used scatter plots (Figure 2) for showing the existing Figure 3 yield production Figure 4 of rice based on the year for the particular districts mentioned in the dataset separately for three different rice ecotypes, namely, Aus, Aman, and Boro.

We have discovered that important features for Aus, Aman, and Boro yield prediction were present in our dataset based on our analysis. However, the dataset had several irrelevant features and anomalies. For this reason, we have used data preprocessing, which is an important step in improving data efficiency and improving prediction results. It is a method for resolving issues with data such as incompleteness, unevenness, and errors, as well as determining the optimal dataset for the research. Figure 5 illustrates the general order of data preprocessing methods.

We began by removing three nonessential features from our dataset to make it more appropriate for improved yield prediction. The features of wheat, potato, and jute production are irrelevant to our study because our goal was to forecast the output of three different rice ecotypes. As a result, we have removed these features. After removing these three features, the dataset included a total of 40 features.

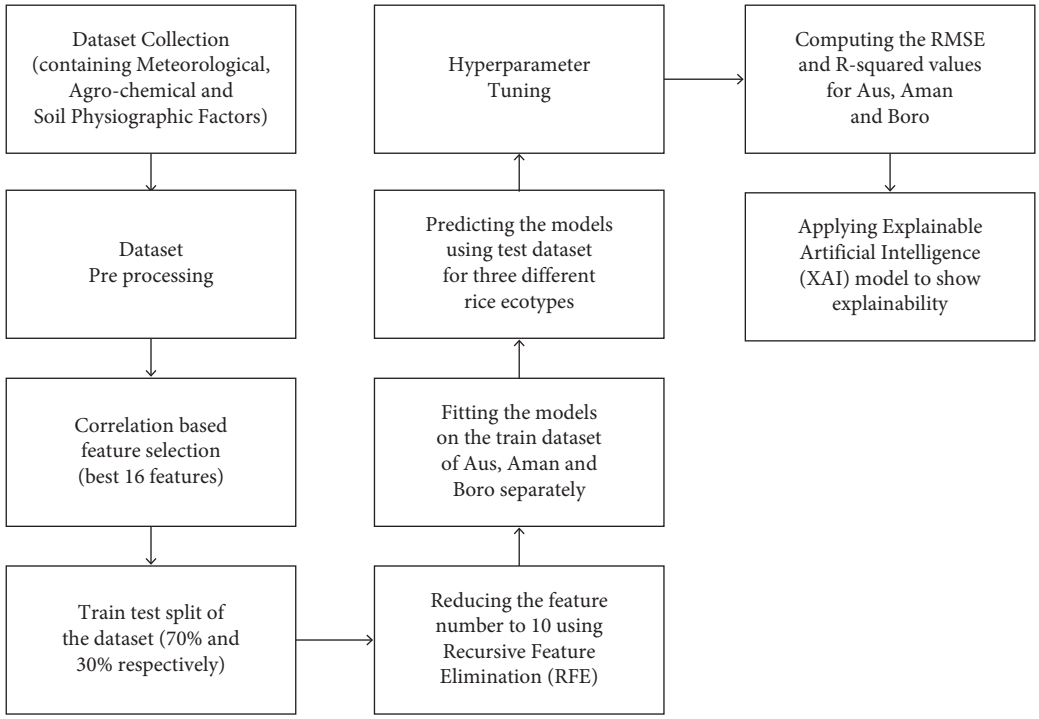

Figure 1: System overview of the proposed yield prediction model.

Filling missing and "0" values was the next preprocessing phase, which is a required data preprocessing method since it might cause lower algorithm performance and irregularities in the whole dataset. We have started by looking for "NaN" values in the data. We have discovered that the "area" feature has 35 "NaN" values or missing data. There are various methods for dealing with missing values. From those, we have applied the imputation technique. With our dataset, the imputation approach is well balanced since it replaces missing values with available information from that particular feature, reducing the risk of missing crucial data. As a result, we have imputed the "NaN" values with the median value for that column in the remainder of the data. Following that, we looked for "0" values in the data. The dataset had a significant number of "0" values, which we observed. We have replaced the "0" values in each column with the mean or median value of that particular column of data based on the best fit for that particular column. The total number was 704, and we replaced the "0" values in each column with the mean or median value of that particular column of data based on the best fit for that particular column. We chose the mean value in most of the columns since the number of "0" values in those columns was quite high, making the usage of the median value unfeasible. To prevent outlier concerns, imputation by the median approach was utilized for just three features, and the remainder of the columns with "0" values were replaced by the column mean values.

After that, we concentrated on feature selection. To begin, we used a heat map to determine the data's correlation. A heat map is a more visual representation of the dataset that helps us to understand how the features are connected. In the case of meteorological, agro-chemical, and soil physiographic factors, we searched for correlated features to ensure that there were no strongly correlated features. The diagonal in the heat map below (Figure 6) indicates that the features are linked with one another, which is why the value is 1. For our dataset, we consider features to be strongly correlated if the correlation is greater than 0.8, and we only maintain one feature from the group of extremely correlated features. We investigated the connection of the three rice ecotypes individually and discovered 27 significantly linked features. We acquire a total of 17 features for our models after eliminating the relevant 23 significantly associated features using manual cross-checking. We omitted the "district" feature column from the 17 features since it is string-type data and has no influence on our yield forecast. Meteorological, agro-chemical, and soil physiographic factors are among the 16 features that are chosen.

We used feature scaling on the 16 features that we chose. Feature scaling is a technique for reducing the number of features in a dataset to a smaller subset. We checked for large differences or outliers among data of the same feature. This is an important observation as we used Support Vector Regression (SVR) for prediction, and SVR is a distance-based algorithm and is affected by the range of features. This is because SVR compares the distances between data points to see how similar they are. We have noticed that several of the dataset's values are vastly different. As a result, we chose the standard scaler as a scaling strategy among the various scaling techniques. The standard scaler is a technique where data is usually distributed within each feature and scales the data in a way that the distribution is centered around 0, with a standard deviation of 1.

Table 1: Feature name list.

Column serial number	Feature name
1	Year
2	Area
3	avg_rainfall
4	max_temperatue
5	min_temperature
6	Aus
7	Aman
8	Boro
9	Wheat
10	Potato
11	Jute
12	Humidity
13	Storm
14	Urea
15	TSP
16	Mp
17	DAP
18	inundationland_Highland
19	inundationland_mediumhighland
20	inundationland_lowland
21	inundationland_mediumlowland
22	inundationland_verylowland
23	Miscellaneous land
24	Calcareous alluvium
25	Noncalcareous alluvium
26	Acid basin clay
27	Calcareous Brown floodplain soil
28	Calcareous grey floodplain soil
29	Calcareous dark grey floodplain soil
30	Noncalcareous grey floodplain soil
31	Noncalcareous dark grey floodplain soil
32	Peat
33	Made-land
34	Noncalcareous Brown floodplain soil
35	Shallow Red-Brown terrace soil
36	Deep Red-Brown terrace soil
37	Brown mottled terrace soil
38	Shallow grey terrace soil
39	Deep grey terrace soil
40	Grey valley soil
41	Brown hill soil
42	Grey piedmont soil
43	Soil moisture

Finally, for the selection of features for three different rice ecotypes, we utilized Recursive Feature Elimination (RFE) to determine the yield of each ecotype individually. Feature selection is a method for selecting a subset of the dataset's most important features (columns). It makes constructing a predictive model easier by reducing the number of input variables. Besides, machine learning algorithms perform better when they have fewer features. For the different ecotypes, we utilized RFE to select distinct sets of features. It is a wrapper feature selection model that creates several models of input features with various subsets and picks the best-performing model from those features. Using this technique, we choose the best-fitted features for the XGBoost and SVR algorithms to improve prediction outcomes for three rice ecotypes. We used a trial and error approach to decrease the features to 10 for both the models and all three rice ecotypes after seeing the prediction outcomes of three ecotypes based on two distinct machine

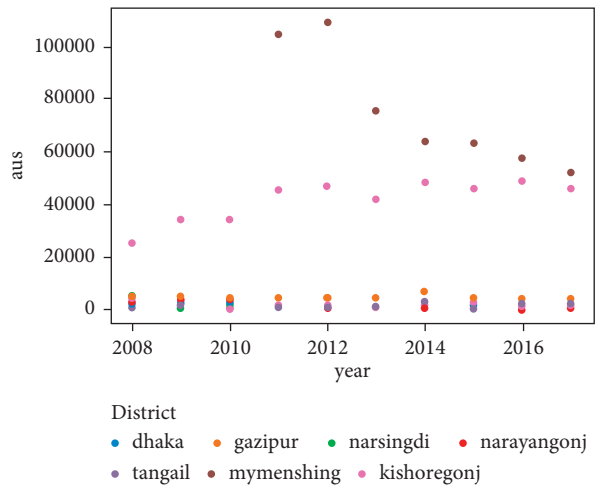

Figure 2: Year-wise (2008–2017) production of Aus.

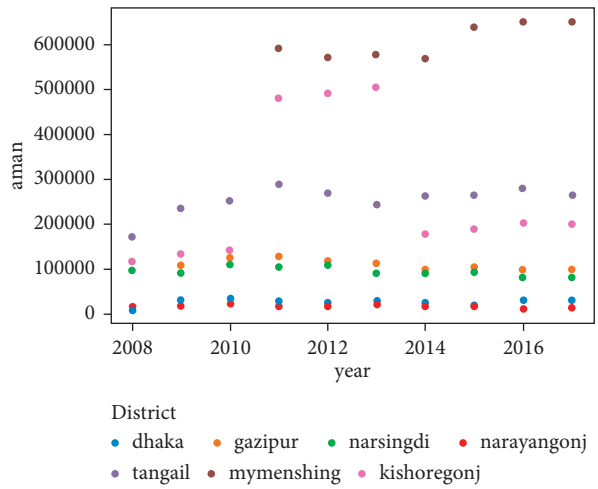

Figure 3: Year-wise (2008–2017) production of Aman.

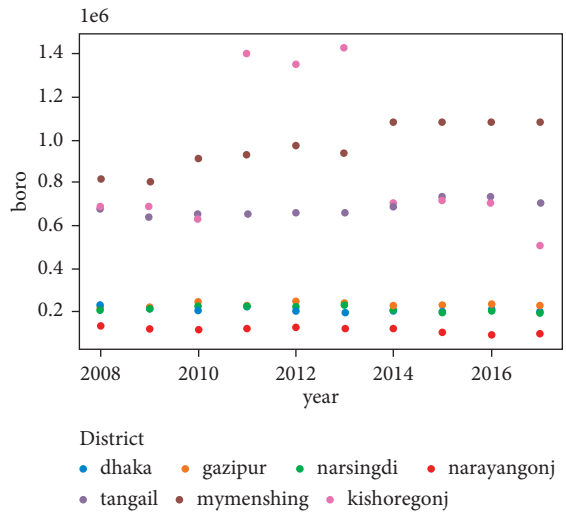

Figure 4: Year-wise (2008–2017) production of Boro.

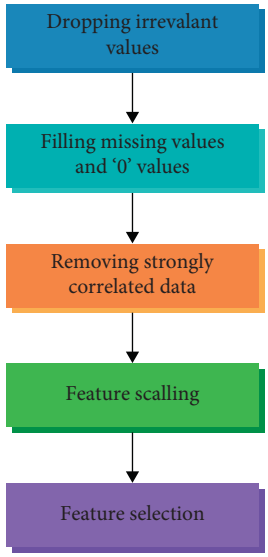

Figure 5: Data preprocessing techniques.

learning models. For all ecotypes and both models, we found that picking 10 features gives us good RMSE and R-squared values. Although the number of features considered in all three rice ecotypes and both models is the same, the features differed from ecotype to ecotype, as well as model to model.

3.2. XGBoost Regression Model. The XGBoost regression model is one of the two machine learning (ML) models that we have used to predict the yield of three different rice ecotypes. XGBoost regression is a powerful supervised regression model. This is very useful for both classification and regression model prediction. The XGBoost decision tree ensemble is a highly scalable decision tree ensemble based on gradient boosting [33]. XGBoost, like gradient boosting, minimizes a loss function to produce an additive expansion of the objective function [34]. This ML model uses the gradient boosting decision tree algorithm called multiple additive regression trees, gradient increasing, gradient boosting machines, or stochastic gradient boosting [35, 36]. This is one of the models that follow increasing techniques like AdaBoost, gradient boosting, extreme gradient boosting, and many more ML models [36–38].

XGBoost dominates structured or tabular datasets in classification and regression problems. XGBoost was chosen as one of the machine learning models since our dataset is structured. This model is also exceptionally quick and performs well because it is built on gradient-boosted decision trees. That is why it is the go-to method for competition winners on the Kaggle competitive data science platform and the most used model by data scientists in different machine learning competitions [33]. In boosting, new models are added to correct earlier models' flaws. Models are introduced one by one until no more improvements are possible. The gradient descent architecture is the basis of XGBoost's boosting. XGBoost further improves the gradient boosting machine architecture through system optimization and algorithmic innovations. Additionally, it is highly flexible and takes advantage of parallel processing. It has regularization capabilities and does a cross-validation test after each iteration. XGBoost was used as one of the models in our dataset to produce a higher overall prediction because of these benefits.

3.2.1. Model Description. The very first step in fitting XGBoost to the training data is to make an initial prediction. This prediction can be anything, but the default value used for both regression and classification is 0.5 [33]. After the initial prediction, the residuals are used to observe the differences between observed and predicted values. This gives an estimation of how right the initial prediction is. Just like extreme Gradient Boost, XGboost fits a regression tree to the residuals. However, XGBoost uses a unique regression tree different from the regular regression tree used by other models. There are more than one ways to build XGBoost trees. However, for the most common method, each tree starts as a single leaf, and all of the residuals go to the leaf. After that, the quality or similarity score is calculated for the residuals.The formula for calculating the similarity score is

$$\text{Similarity Score} = \frac{\text{Sum of Residuals, Squared}}{\text{Number of Residuals} + \lambda}. \quad (1)$$

Here, lambda is a regularization parameter. Its primary job is to reduce the sensitivity of the prediction. Generally, we see that when the residuals in a node are very different, they cancel each other out, and the similarity score is relatively small. In contrast, when the residuals are similar, or there is just one of them, they do not cancel each other out, and the similarity score is relatively large. Next, we quantify how much better the leaves cluster similar residuals than the root by calculating the gain of splitting the residuals into two groups.

The formula for calculating gain is as follows:

$$\text{Gain} = \text{Leftsimilarity} + \text{Rightsimilarity} - \text{Rootsimilarity}. \quad (2)$$

After that, calculate the gain for different thresholds depending on the dataset and the branches of trees deriving from the dataset. After comparing different thresholds, the model will use the threshold that gave the largest gain. The model uses the pruning of XGBoost trees based on their gain value. If the difference between the gain and γ (gamma) of a branch is negative, then the model will remove that particular branch, and if the difference between the gain and γ (gamma) is positive, then the branch will not be removed.

$$\text{Gain} - \gamma = \begin{cases} \text{If positive,} & \text{then do not prune,} \\ \text{If negative,} & \text{then prune.} \end{cases} \quad (3)$$

Output value calculation for the leaves is the next step of the regression model. It used the following formula to calculate the output value:

$$\text{Output Value} = \frac{\text{Sum of Residuals}}{\text{Number of Residuals} + \lambda}. \quad (4)$$

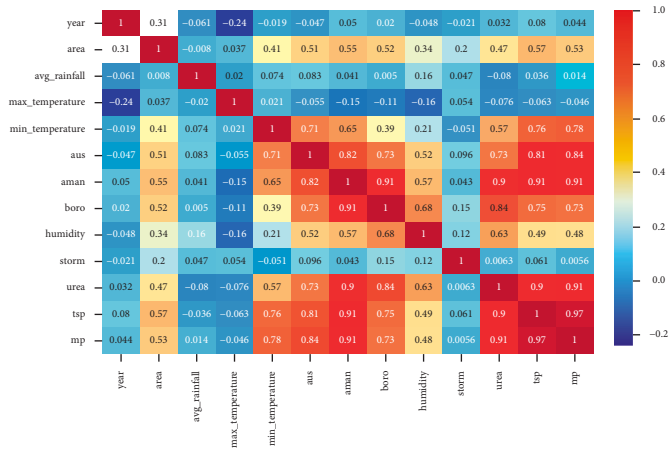

Figure 6: Partial heat map based on features correlations.

After finishing this process, the model will be done by building a new tree for predictions. The model will make new predictions by taking the initial prediction, and then with the help of the learning rate, it will add the output of the tree continuously. It will check the residuals values for going forward by building new trees based on the new residuals. This process will continue until the residuals are super small or it has reached the maximum number.

With the result of more pruning, smaller output values can be achieved for the leaves by using the value of lambda greater than 0 as it initiated more pruning and shrinking of similarity scores. Let us observe and explore the mathematics of XGBoost regression based on the minimizing of the loss function and finding a suitable output value for the model.

From the earlier discussion, we know that XGBoost regression starts with an initial prediction value that usually is 0.5. To determine how good the initial prediction is, the loss function is calculated using the formula given below:

$$L(y_i, p_i) = \frac{1}{2}(y_i - p_i)^2. \tag{5}$$

In general, the formula looks like what as follows:

$$\sum_{i=1}^{n} L(y_i, p_i) = \frac{1}{2}(y_i - p_i)^2. \tag{6}$$

After calculating the loss function, we can observe if the new prediction is improving or not based on the initial prediction that was made. The XGBoost regression uses a formula and uses the calculated loss function for building new trees using the output value in this formula that is calculated. The formula is as given below:

$$\sum_{i=1}^{n} L(y_i, p_i) + \frac{1}{2}\lambda O_v^2. \tag{7}$$

Here, O_v = output value.

This equation contains the loss function and the regularization term and tries to minimize the whole equation. Here, the first part of the formula represents the loss function that we discussed before, and the second part represents the regularization term where lambda and output values are present.

For the output value optimization of the first tree, the P_i is replaced with the initial predictions and output value and the equation looks like as given below:

$$\sum_{i=1}^{n} L\left(y_i, p_i^0 + 0_v\right) + \frac{1}{2}\lambda O_v^2. \tag{8}$$

Then λ (lambda) is set to 0 for simple calculation and the updated equation looks like what as follows:

For $\lambda = 0$,

$$\sum_{i=1}^{n} L\left(y_i, p_i^0 + O_v\right). \tag{9}$$

If the plot shows the output value and loss function that the above equations give, then a parabola-like structure is created. In the parabola-like structure if the lambda value increases, then the parabola shifts towards zero, and this is how regularization works.

If λ (lambda) is set to a 0 value, then the optimal output value is situated at the bottom of the parabola, and here, the derivative is also zero. XGBoost generally uses second-order Taylor approximation for regression models. The updated equation containing both loss function and output values is as follows:

$$\begin{aligned} &L\left(y_1, p_1^0\right) + g_1 O_v + \frac{1}{2}h_1 O_v^2 + L\left(y_2, p_2^0\right) + g_2 O_v \\ &+ \frac{1}{2}h_2 O_v^2 + \cdots + L\left(y_n, p_n^0\right) + g_n O_v + \frac{1}{2}h_n O_v^2 + \frac{1}{2}\lambda O_v^2. \end{aligned} \tag{10}$$

Here, we can observe that the first part of the above equation contains the loss function. The next part of the equation represents the first derivative of the loss function, and then the later part contains the second derivative of the loss function. Now, XGBoost uses "g" for representing the first derivative part as this part is related to Gradient Descent and as the second derivative came from Hessian, and for that reason, XGBoost uses "h" to represent the second derivative. Replacing these values, the above equation will be

$$L(y, p_i + O_v) = L(y, p_i) + gO_v + \frac{1}{2}hO_v^2. \tag{11}$$

Now, using the summation, we get

$$\sum_{i=1}^{n} L\left(y_i, p_i^0 + O_v\right) + \frac{1}{2}\lambda O_v^2. \tag{12}$$

After expanding the summation, it becomes

$$L\left(y_1, p_1^0 + O_v\right) + L\left(y_2, p_2^0 + O_v\right) + \cdots + L\left(y_n, p_n^0 + O_v\right) + \frac{1}{2}\lambda O_v^2. \tag{13}$$

Then, let us plug in the Taylor approximation in the above equation (13). The equation then looks like as follows:

$$L\left(y_1, p_1^0\right) + g_1 O_v + \frac{1}{2}h_1 O_v^2 + L\left(y_2, p_2^0\right) + g_2 O_v + \frac{1}{2}h_2 O_v^2 + \cdots + L\left(y_n, p_n^0\right) + g_n O_v + \frac{1}{2}h_n O_v^2 + \frac{1}{2}\lambda O_v^2. \tag{14}$$

The next step is to remove those terms from the equation that does not contain output value terms. The remaining function can be minimized by first taking the derivative of the output value. Now, to solve for the lowest point in the parabola, the derivative needs to be set to 0.

Then the output equation will be

$$O_v = \frac{-(g_1 + g_2 + \cdots + g_n)}{(h_1 + h_2 + \cdots + h_n + \lambda)}. \tag{15}$$

For the negative residuals represented by g_i, the formula is as follows:

$$g_i = \frac{d}{dp_i}\left(\frac{1}{2}(y_i - p_i)^2\right) = -(y_i - p_i). \tag{16}$$

The number of residuals in the equation, h_i, can be given as follows:

$$h_i = \frac{d^2}{dp_i^2}\left(\frac{1}{2}(y_i - p_i)^2\right) = 1. \tag{17}$$

Putting the negative residuals, g_i, and number of residuals, h_i value, in equation (15), the output value equation now is as follows:

$$O_v = \frac{(y_1 - p_1) + (y_2 - p_2) + \cdots + (y_n - n)}{(1 + 1 + \cdots + 1 + \lambda)}. \tag{18}$$

Finally, we can write the equation as

$$O_v = \frac{\text{Sum of residuals}}{\text{Number of residuals } + \lambda}. \tag{19}$$

We have already seen this equation in the first part of the model description, where we describe the model without the mathematical terms.

3.2.2. Description of Model Implementation. Initially, we divided our dataset for training and testing. We have kept 70 percent data for training and 30 percent data for testing.

After that, hyperparameter tuning is used to control the XGBoost model's overall behavior. The final aim was to discover the best hyperparameter combination that minimizes a predefined loss function and produces better results. Because finding the appropriate hyperparameter might be time-consuming, the "GridSearchCV" *Python* package is used to implement a grid search method with 10-fold cross-validation. We utilized the grid search method to determine the optimum combination of six hyperparameters for XGBoost that are the most important, which are defined as those with the highest chance of the algorithm producing the most accurate, unbiased results the fastest without overfitting. We found that in case of Aus yield prediction, the best hyperparameters are {"gamma": 0, "learning_rate": 0.12, "max_depth": 5, "min_child_weight": 3, "n_estimators": 1000, "reg_lambda": 3}. For Aman yield, the best hyperparameters are {"gamma": 0, "learning_rate": 0.1, "max_depth": 3, "min_child_weight": 1, "n_estimators": 1000, "reg_lambda": 1}. And in case of Boro, the best hyperparameters found are {"gamma": 0, "learning_rate": 0.15, "max_depth": 3, "min_child_weight": 1, "n_estimators": 1000, "reg_lambda": 1}.

In the next step, the RFE (Recursive Feature Elimination) model is implemented for the XGBoost classifier. It plays an influential role in selecting the most accurate features for getting the predicted value. After using the heat map, 16 features are found, and among them, 10 best features have been identified using RFE. This set of 10 features is different for each rice ecotype. The set of best 10 features for Aus, Aman, and Boro yield prediction is given below.

For the Aus rice ecotype, the selected features using RFE are year, area, max_temperature, min_temperature, aman, boro, humidity, TSP, inundationland_mediumlowland, and shallow Red-Brown terrace soil. For the Aman rice ecotype, the selected features using RFE are year, area, max_temperature, boro, humidity, TSP, inundationland_mediumlowland, noncalcareous grey floodplain soil, shallow red-brown terrace soil, and Aus. For the Boro rice ecotype, the selected features using RFE are area, max_temperature, aman, humidity, tsp, inundationland_mediumlowland, calcareous alluvium, noncalcareous alluvium, noncalcareous grey floodplain soil, and Aus.

It is seen that the best 10 features are not the same for all rice ecotypes. We then applied XGBoost for all three rice ecotypes and generate RMSE and R-squared values. Finally, we have applied LIME to show the explainability of the model for each rice ecotype separately.

3.3. Support Vector Regression (SVR) Model. The second ML model used for yield prediction is Support Vector Regression (SVR). A Support Vector Machine (SVM) defined by a separating hyperplane is a classifier [39]. When a support vector machine is used for a regression problem, it is called Support Vector Regression (SVR). SVR is a regression model used for the prediction of a continuous dataset. Since our dataset contains tabular data with constant values, SVR also works well in our dataset. Instead of minimizing the error rate like other regression models, this model tries to fit the

error into a certain threshold. The hyperplane is a separation line between the data classes [40]. For a nonlinear classifier or regression line, a radial basis kernel function is used. The kernel function is used to transform n-dimensional input to m-dimensional input. Here, m is higher than n, and the dot product in higher-dimensional efficiency is found. A linear classifier or regression curve in a higher dimension is converted to a nonlinear classifier or regression curve in a lower dimension through the kernel function, which is the central concept behind using a kernel function. The kernel's main objective is to do calculations in any d-dimensional space where d is greater than 1. As a result, quadratic, cubic, or any polynomial equation of a large degree can be found. That is why it is implemented in our dataset. This model is called the nonparametric technique as it depends on kernel functions. Some of the terminologies of this regression model are as follows:

(1) Kernel: This function is used to convert lower-dimensional data into higher-dimensional data. There are five kernels available for Support Vector Regression.

(2) Hyperplane: In SVR, this line helps to predict the target values or continuous values using the distance method.

(3) Boundary Line: In SVR, boundary lines are used to create a margin just like Support Vector Machine (SVM).

(4) Support Vectors: Support vectors represent the closest data points towards either side of the hyperplane. These points create boundary lines.

The SVR, unlike other regression models, aims to fit the best line within a threshold value rather than minimizing the error between the real and predicted value. The threshold value is calculated by measuring the distance between the hyperplane and the boundary line. Another reason to consider SVR is that it offers outstanding generalization and prediction accuracy. Additionally, when compared to other regression models, SVR requires less processing and is simple to apply. Because of these advantages, SVR was chosen as the second ML model for our dataset.

3.3.1. Model Description. First, there are two boundary lines selected from the epsilon distance of the hyperplane. For simplicity, let us assume that epsilon is represented by "e." Let us take that the hyperplane goes through the Y-axis and is a straight line. Then, the equation is

$$wx + b = 0. \tag{20}$$

From this equation, we can derive the two equations of the boundary lines. These equations are, respectively, given as

$$\begin{aligned} Wx + b &= +e, \\ Wx + b &= -e. \end{aligned} \tag{21}$$

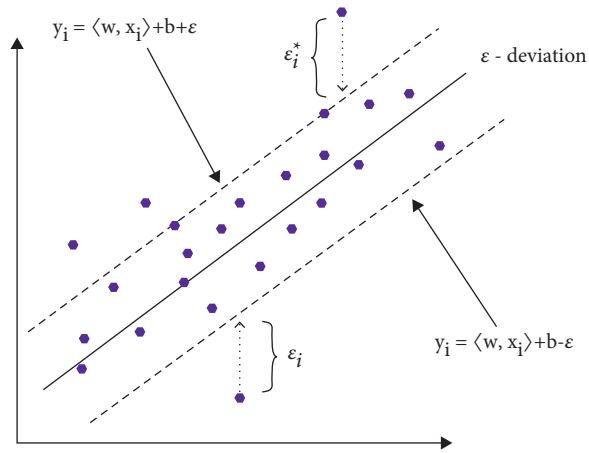

Figure 7: Support Vector Regression model.

From these equations, the equation that satisfies SVR for any linear hyperplane can be derived, and that equation is

$$e \leq y - Wx - b \leq +e. \tag{22}$$

The primary purpose here is to create a boundary at distance "e" from both sides of the hyperplane so that the closest data points towards the hyperplane or the support vectors remain in the boundary line that is created.

In Figure 7, the solid black line is the hyperplane here, and the rest two dotted black lines are the boundary lines, and "e" is distant from the hyper-line. This boundary indicates the tolerance's margin, and the points within this boundary will be taken as accepted values as they will have the least error rate. This way, it will provide a better fitting regression model.

Another essential factor of Support Vector Regression is the kernel function. Selecting and using the right kernel is very important for getting a good prediction. There are five kernels for SVR: linear, radial basis function (RBF), polynomial, sigmoid, and precomputed.

The linear kernel function equation is given as

$$K(x_i, x_j) = x_i \cdot x_j. \tag{23}$$

The radial basis function (RBF) kernel function is represented by

$$K(x_i, x_j) = \exp\left(-\gamma \left\| x_i - x_j \right\|^2\right) \gamma > 0. \tag{24}$$

The polynomial kernel function is

$$K(x_i, x_j) = (x_i \cdot x_j + 1)^d d. \tag{25}$$

The sigmoid kernel function equation is

$$K(x_i, x_j) = \tanh\left(b(x_i, x_j) + c\right) b, c. \tag{26}$$

The appropriate kernel function selection is the key to improving a particular model's performance, and this should be given importance while using the Support Vector Regression model for continuous datasets.

3.3.2. Description of Model Implementation. After the preprocessing of the dataset, the Support Vector Regression model is used on the dataset to predict the yield of Aus, Aman, and Boro separately. The necessary packages for this ML model were imported. The dataset is split into train and test sets containing 70% for training and 30% for testing.

Then, hyperparameter tuning is instantiated just like we did in the XGBoost model to find the best combination of the most important hyperparameters of SVR. Just like XGBoost, we used the "GridSearchCV" package with 10-fold cross-validation to implement the grid search method to determine the optimum combination of the five hyperparameters of SVR that we found most important. The result showed that in case of Aus yield prediction, the best hyperparameters are {"*C*": 1.0, "epsilon": 0.1, "gamma": "scale," "kernel": "linear," "tol": 0.001}. The best hyperparameters for Aman yield are {"*C*": 1.0, "epsilon"": 0.2, "gamma": "scale," "kernel": "linear," "tol": 0.001}. For Boro yield, the best hyperparameters are {"*C*": 2.5, "epsilon": 0.1, "gamma": "scale," "kernel": "linear," "tol": 0.01}.

After that, RFE was used to select the best 10 features for running the model. Here, the 10 features were mainly selected after tuning the selection and observing the results. The best 10 features differ for each rice type just like XGBoost. The best 10 features are provided below for three separate rice ecotypes.

For the Aus rice ecotype, the selected features using RFE are year, area, min_temperature, aman, boro, humidity, TSP, inundationland_mediumlowland, calcareous alluvium, and noncalcareous alluvium.

For the Aman rice ecotype, the selected features using RFE are area, avg_rainfall, max_temperature, min_-temperature, boro, humidity, TSP, inundationland_mediumlowland, noncalcareous alluvium, and Aus.

For the Boro rice ecotype, the selected features using RFE are year, area, min_temperature, aman, storm, TSP, inundationland_mediumlowland, noncalcareous alluvium, noncalcareous grey floodplain soil, and Aus.

Just like XGBoost, it is also seen that the set of 10 best features is different for three different rice ecotypes in the case of SVR too. Lastly, based on the features found from RFE, we generated the prediction values of the yield of Aus, Aman, and Boro. For accuracy, we have calculated RMSE and R-squared values. Finally, we have applied LIME for explainability of the SVR model for each rice ecotype.

3.4. Local Interpretable Model-Agnostic Explanations (LIME). LIME is a model-independent explanation approach that uses a local (explainable) intermediary model to explain any machine learning model [41]. The technique begins by creating a fresh artificial dataset of altered samples surrounding the instance of interest, then utilizing the black-box model as an oracle, obtaining their associated predictions. The weighting of the additional data points is dependent on their closeness to the original observation. The new data are then used to perform a feature selection phase, which selects the features that best explain the model output. Finally, a basic model is trained and fitted to the newly picked data, and a conclusion is drawn from it.

Because LIME is a local explainability framework, it calculates the influence and relevance of each feature on a local level (i.e., this is done for each prediction). Instead of providing an overall summary, we wanted to highlight how a set of features affects an individual yield estimate. This way, we would be able to show how meteorological, agrochemical, and soil physiographic factors contribute to yield prediction and if their relevance is equal or not. In comparison to other XAI models, LIME is an excellent fit to serve this purpose. Furthermore, the local predictions of LIME appear to be accurate in their interpretations [42].

3.4.1. Description of LIME. We will give a brief overview of LIME and how it maintains model-agnosticism. LIME targets to identify an interpretable model over interpretable representation also denoted as locally faithful to the classifier. Lime can be useful in a local approximation of an interpretable model though it cannot approximate a black box model globally. Let us denote the original representation of an instance by $x \in R^d$, and by $x' \in R^{d'}$, we represent a vector of the interpretable representation.

Formally, we can represent the explanation model as g: $R^{d'} \longrightarrow R, g \in G$, where G represents a class of potentially explainable models, for example, decision trees, and linear models. Here, given a model $g \in G$, an explanation in visual or textual form can be presented to the user. As not every $g \in G$ is simple enough for interpreting, $\Omega(g)$ can be a measure of complexity for g in both soft constraint and hard constraint.

Now, let f: $R^d \longrightarrow R$ be a model explained in classification, and $f(x)$ is the probability indicating x belongs to a particular class. A proximity measure $\Pi_x(z)$ can be used between z and x to define locality around x.

Next, we can represent a measure of how unfaithful g is in approximating f, where the locality defined by Π_x is $\mathscr{L}(f, g, \Pi_x)$. If we want to ensure both local fidelity and interpretability, then we have to have a low $\Omega(g)$ that is enough to be interpretable by a human with minimized $\mathscr{L}(f, g, \Pi_x)$. We can get the explanation $\xi(x)$ generated by LIME by solving the equation below:

$$\xi(x) = \underset{g \in G}{\operatorname{argmin}} \mathscr{L}(f, g, \Pi_x) + \Omega(g). \tag{27}$$

The formulation given above can be sued with different fidelity functions $\mathscr{L}$, explanation families G, and complexity measures Ω.

4. Result Analysis

The root mean square error (RMSE) and R-squared (r^2) values will be used to analyze Aus, Aman, and Boro yield predictions. The root mean square error (RMSE) represents a model's error in predicting a quantitative value. The RMSE in this study represents the error of the XGBoost and SVR models while predicting the yield value. The lower the RMSE score, the more accurate the research results will be.

The R-squared (r^2) value indicates how well a value fits on a regression line. The r^2 value ranges from 0 to 1. The

Table 2: The exact value and the predicted value of Aus yield in the different districts in different years.

Year	District/sub district	Exact value of Aus yield (in metric tons)	Exact value of Aus yield (after implementing standard scaler)	Predicted value of Aus yield in XGBoost (after implementing standard scaler)	Predicted value of Aus yield in SVR (after implementing standard scaler)
2009	Gazipur	4199	−0.476492	−0.49573976	−0.44002507
2016	Mymenshing	57140	1.318526	1.530111	1.48935865
2014	Dhaka	948	−0.586720	−0.60301137	−0.50136886
2011	Narayangonj	1003	−0.566478	−0.54305744	−0.41852786
2016	Narayangonj	327	−0.607776	−0.5964272	−0.57439756
2015	Gazipur	4071	−0.480832	−0.4924866	−0.5309091
2009	Narsingdi	530	−0.600893	−0.59127367	−0.49242472
2014	Gazipur	6324	−0.404441	−0.48209077	−0.5032799
2016	Dhaka	1431	−0.570344	−0.60374486	−0.55784708

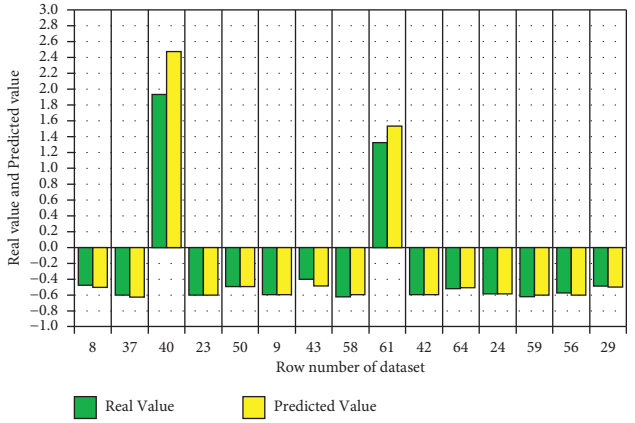

Figure 8: Comparison between Real Values and Predicted Values of Aus yield for XGBoost.

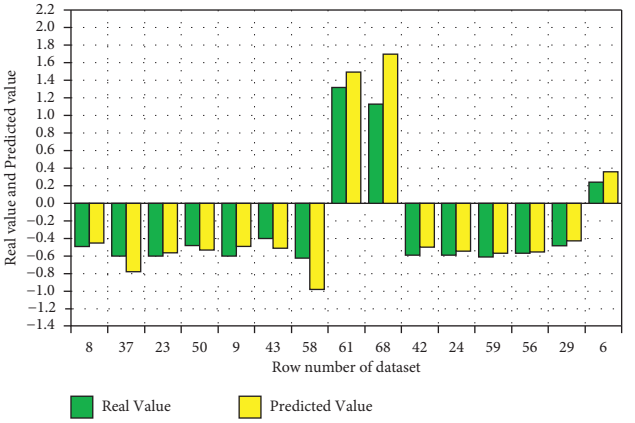

Figure 9: Comparison between Real Values and Predicted Values of Aus yield for SVR.

closer we get to 1 after predicting the yield using XGBoost and SVR, the more accurate the results will be.

4.1. Result Analysis of Aus Yield Prediction. Some of the values of the exact value and the predicted values turn negative after standard scaling, as seen in Table 2. Some random data are provided in tabular form to provide a quick overview of the exact value and expected value of Aus yield. From 2008 to 2017, we have yield values for ten years in our dataset.

Table 2 shows data from 2008 to 2017 to give an overview of the findings. Table 2 shows that the exact value of the Aus yield in the Gazipur district in 2009 is −0.476492, and the predicted value using the XGBoost regression model is −0.49573976, while the predicted value using the SVR model is −0.44002507. The XGBoost model's predicted value is closer to the actual yield value than the SVR model. We can also see that in the same district in 2015, the exact value of Aus yield (after adopting standard scaler) is −0.480832, and using the XGBoost regression model, we obtain −0.4924866 as the predicted value, and using the SVR model, we get −0.5309091 as the predicted value. The XGBoost model's predicted value is substantially closer to the actual yield value than the SVR model.

Figure 8 shows a visual representation of the real and predicted values of the Aus yield for the XGBoost model,

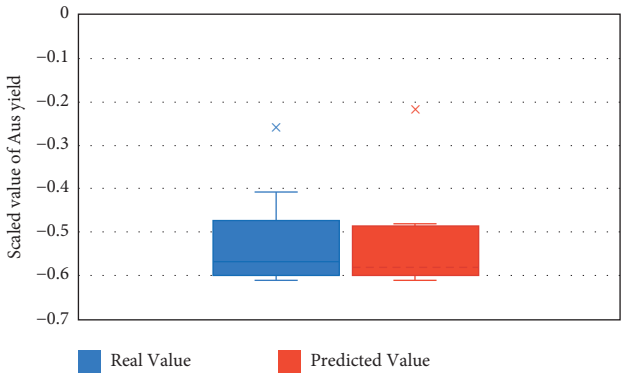

Figure 10: Comparison between real values and predicted values of Aus yield for XGBoost using box plot.

while Figure 9 shows a visual representation of the real and forecasted values of the Aus yield for the SVR model. We can see that the predicted values of Aus yield are pretty comparable to the actual values in both figures.

For the XGBoost and SVR models, respectively, Figure 10 compares the real (Figure 11) value and predicted value of the Aus yield prediction using box plots. Outliers have been eliminated from all the box plots to generalize the comparative studies between real and predicted values. Figure 10 shows that the minimum scaled real value for Aus

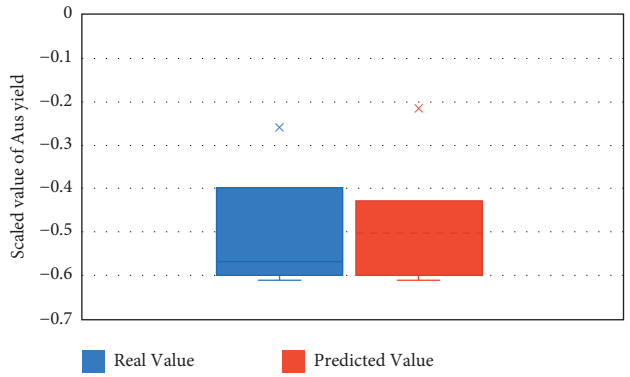

Figure 11: Comparison between Real Values and Predicted Values of Aus yield for SVR using box plot.

Table 3: RMSE and r^2 value after Aus yield prediction.

	Root mean square error	r^2 value
XGBoost	0.236562	0.903298
Support vector machine	0.243721	0.897356

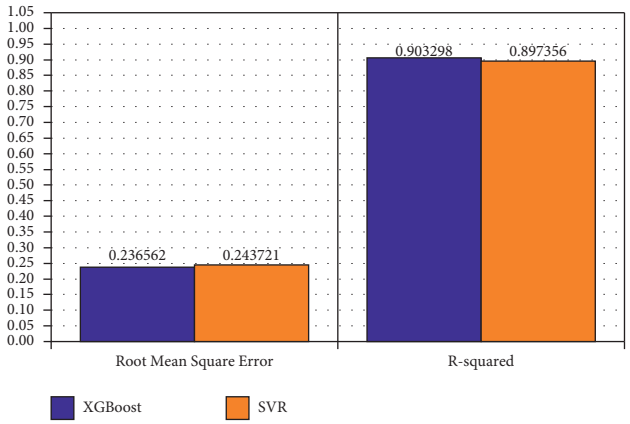

Figure 12: Comparison between the RMSE and R-squared values of XGBoost and SVR for Aus yield Prediction.

yield when applying XGBoost is −0.611878 and the minimum scaled forecasted value is −0.6138531. The real and anticipated values are in the first quartile (Q1) at −0.601334 and −0.59828436, respectively. The median values are, respectively, −0.570344 and −0.5819483. The actual and forecasted values have third quartile (Q3) values of −0.476492 and −0.48761767, respectively. Finally, while employing XGBoost, the maximum values for the real and forecast values of Aus yield are 1.926969 and 2.4729662, respectively.

Figure 11 displays the minimum, first quartile, median, third quartile, and maximum values of the actual and predicted values of Aus yield when applying the SVR model. The values for the real values' minimum, first quartile (Q1), median, third quartile (Q3), and maximum are −0.611878, −0.601334, −0.570344, −0.404441, and 1.318526. The minimum, first quartile (Q1), median, third quartile (Q3), and maximum values for the predicted values using SVR are, respectively, −0.97566598, −0.56540449, −0.5032799, −0.43137586, and 1.69320438.

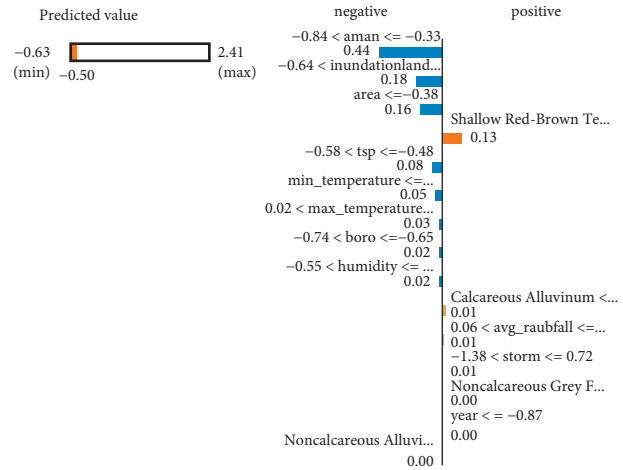

Figure 13: A plot of LIME model values for the XGBoost model and Aus yield data.

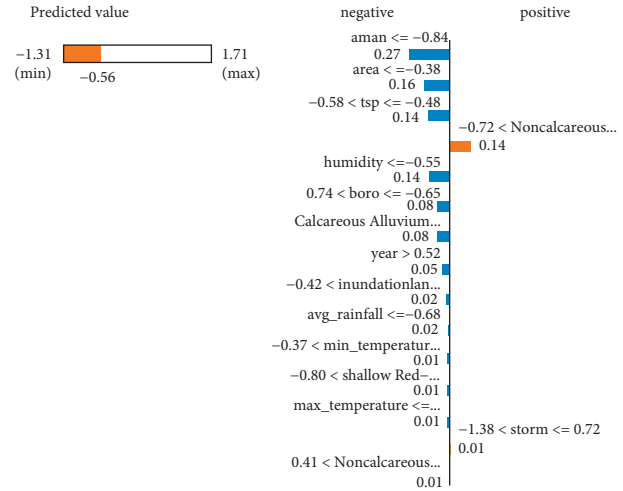

Figure 14: A plot of LIME model values for the SVR model and Aus yield data.

For Aus yield prediction, we observed that XGBoost outperformed SVR in terms of root mean square error (RMSE). Table 3 shows that when XGBoost is used, the root mean square error is 0.236562. In SVR, however, the error is 0.243721. The r^2 value we have in XGBoost is 0.903298. The r^2 score in SVM regression is 0.897356. As can be seen from the bar chart depiction in Figure 12, XGBoost produces better results for Aus rice yield prediction than SVR.

LIME may be used to analyze both ML models for Aus yield data and interpret a model around specific observations. We illustrate the performance of the XGBoost model for the prediction of Aus yield for a specific data sample in Figure 13, where the true value is −0.476491707 and the predicted value is −0.49573976. With an interpretable linear model, the LIME model finds a local approximation. Figure 13 shows a list of explanations, each showing the contribution of each feature to the prediction of the above-mentioned data sample. This allows for local interpretation,

Table 4: The exact value and the predicted value of Aman yield in the different districts in different years.

Year	District/sub district	Exact value of Aman yield (in metric tons)	Exact value of Aman yield (after implementing standard scaler)	Predicted value of Aman yield in XGBoost (after implementing standard scaler)	Predicted value of Aman yield in SVR (after implementing standard scaler)
2009	Gazipur	105450	−0.437995	−0.4087671	−0.52376373
2013	Narsingdi	89870	−0.519071	−0.4641888	−0.59059468
2013	Mymensing	575001	2.005489	2.0059824	1.90912547
2017	Tangail	262172	0.377566	0.33152503	0.2555941
2011	Narsingdi	103641	−0.447409	−0.46625412	−0.49485915
2015	Gazipur	101542	−0.458332	−0.4217301	−0.42213914
2009	Narsingdi	89811	−0.519378	−0.46398324	−0.67467042
2011	Narayangonj	18736	−0.889244	−0.8903893	−0.6696109
2016	Narayangonj	9945	−0.934991	−0.91264105	−0.73712858
2016	Dhaka	27419	−0.844058	−0.85871255	−0.7228565

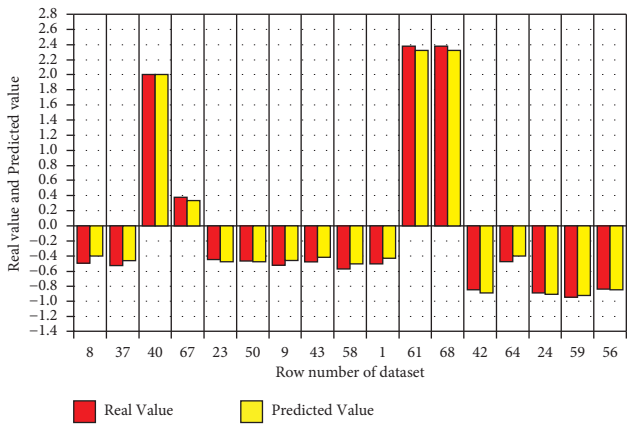

Figure 15: Comparison between real values and predicted values of Aman yield for XGBoost.

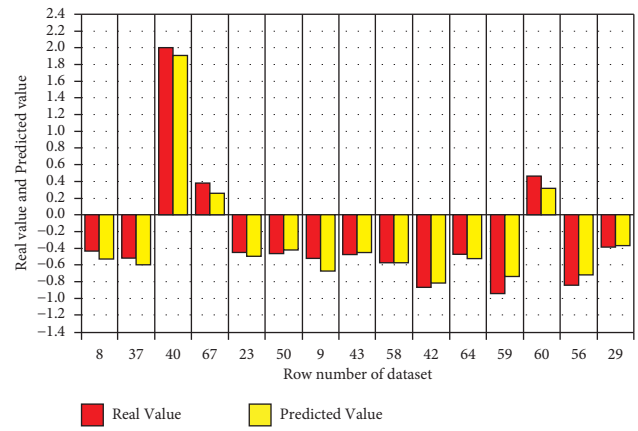

Figure 16: Comparison between real values and predicted values of Aman yield for SVR.

as well as determining which features, such as aman, inundationland_mediumlowland, area, TSP, min_temperature, max_temperature, and Boro will have the most influence on the prediction.

Figure 14 shows the behavior of the SVR model for Aus yield prediction for a specific data sample where the true value is −0.5703435 and the forecast value is −0.55784708. With an interpretable linear model, the LIME model finds a local approximation. A number of explanations can be shown, each indicating the contribution of each aspect to the forecast of the above-mentioned data sample. This enables local interpretation as well as establishing which changes in the features aman, area, TSP, humidity, Boro, calcareous alluvium, and year will have the greatest impact on the prediction.

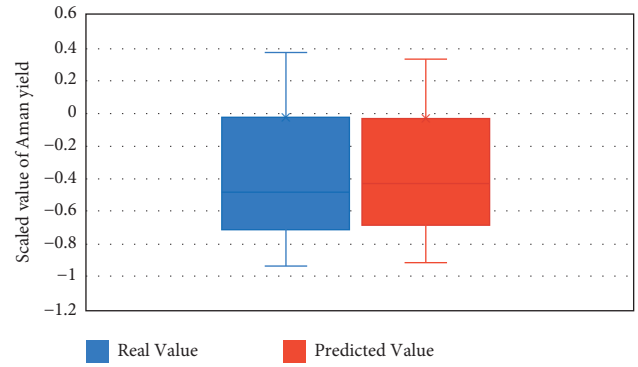

Figure 17: Comparison between real values and predicted values of Aman yield for XGBoost using box plot.

4.2. Result Analysis of Aman Yield Prediction. As we explained in the result part of the Aus yield prediction, after applying standard scaling to Aman yield data, several of the real and predicted values have gone negative, as seen in Table 4. Some random records are presented for Aman yield, as they were for Aus yield.

In Table 4, we have put some data from 2008 to 2017 to get an overall idea of the results. Table 4 shows that the exact value of Aman yield in the Gazipur district in 2009 is −0.437995, and the predicted value using the XGBoost regression model is −0.4087671, while the predicted value using the SVR model is −0.52376373. XGBoost's anticipated values are closer to the actual yield value than SVR's. We can also see that in the year 2013, the exact value of Aman yield (after applying standard scaler) in the Mymensing district is 2.005489, and we obtain 2.0059824 as the projected value when using the XGBoost regression model and 1.90912547 while using the SVR model. The projected value of the

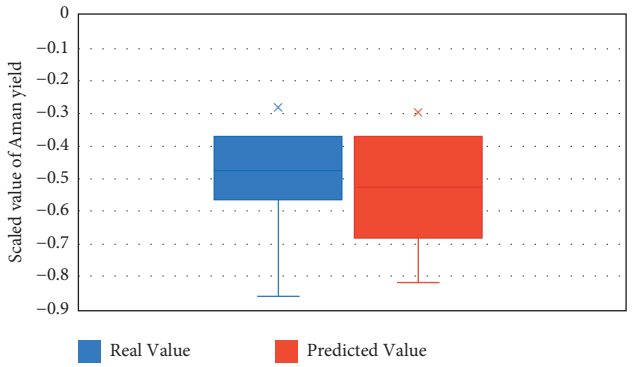

Figure 18: Comparison between real values and predicted values of Aman yield for SVR using box plot.

Table 5: RMSE and r^2 value after Aman yield prediction.

	Root mean square error	r^2 value
XGBoost	0.093824	0.991293
Support vector machine	0.179802	0.968023

XGBoost model is significantly closer to the true value of the yield than that of the SVR model.

Figure 15 shows a visual representation of the real and predicted values of the Aman yield for the XGBoost model, while Figure 16 shows a visual representation of the real and forecasted values of the Aman yield for the SVR model. We can see that the predicted values of Aman yield are pretty comparable to the actual values in both figures.

Figure 17 use box plots to compare the actual Figure 18 value and anticipated value of the Aman yield prediction for the XGBoost and SVR models, respectively. In using XGBoost, the minimum-scaled real value for Aman yield is −0.934991, and the minimum scaled predicted value is −0.91264105, as shown in Figure 17. The first quartile (Q1) values of the real and predicted values of Aman yield are, respectively, −0.7085155 and −0.680206225. The corresponding median values are −0.479548 and −0.43538284. Third quartile (Q3) values for the actual and predicted values are −0.0302145 and −0.037292635, respectively. The maximum values for the real and forecasted values of the Aman yield when using XGBoost are 2.384269 and 2.327917, respectively.

The SVR model's predicted and actual values for Aman yield are shown in Figure 18 along with their minimum, first quartile, median, third quartile, and maximum values. The true values' minimum, first quartile (Q1), median, third quartile (Q3), and maximum values, respectively, are −0.934991, −0.572973, −0.479355, −0.377922, and 2.005489. The minimum, first quartile (Q1), median, third quartile (Q3), and maximum values for the projected values using SVR are, respectively, −0.81097272, −0.67467042, −0.52376373, −0.37076501, and 1.90912547.

In the case of Aman yield prediction, we observed that XGBoost outperformed SVR in terms of root mean square error (RMSE). Table 5 shows that when SVR is used, the root mean square error is 0.179802. However, the error in XGBoost is 0.093824, which is significantly better than SVR.

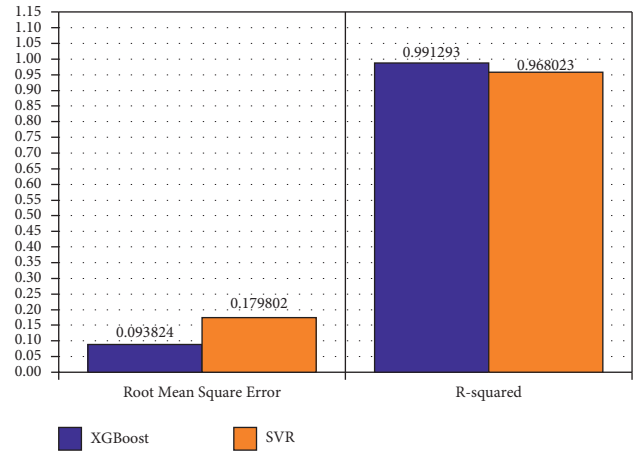

Figure 19: Comparison between the RMSE and R-squared values of XGBoost and SVR for Aman yield prediction.

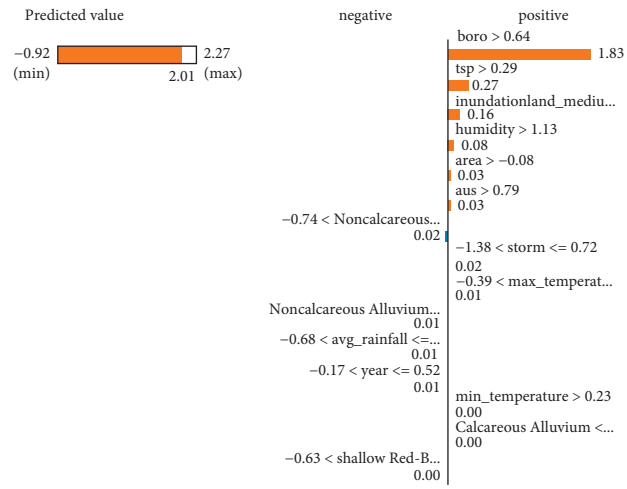

Figure 20: A plot of LIME model values for the XGBoost model and Aman yield data.

We have an r^2 value of 0.968023 in SVR. The r^2 value in XGBoost is 0.991293. As shown in Figure 19, XGBoost outperforms SVR in predicting Aman rice production.

LIME may be used to interpret a model around specific observations for both ML models for Aman yield data just like Aus yield data. The behavior of the XGBoost model for the prediction of Aman yield for a given data sample is explained in the following example (Figure 20), where the true value is 2.00548906, and the forecasted value is 2.0059824. With an interpretable linear model, the LIME model finds a local approximation. Figure 20 shows a list of explanations, each indicating the contribution of each attribute to the prediction of the above-mentioned data sample. This allows for local interpretation and identifies which changes in the features Boro, TSP, inundationland_mediumlowland, humidity, area, and Aus will have the strongest influence on the forecast.

Figure 21 shows the behavior of the SVR model for Aman yield prediction for a specific data sample where the true value is 2.00548906, and the forecasted value is

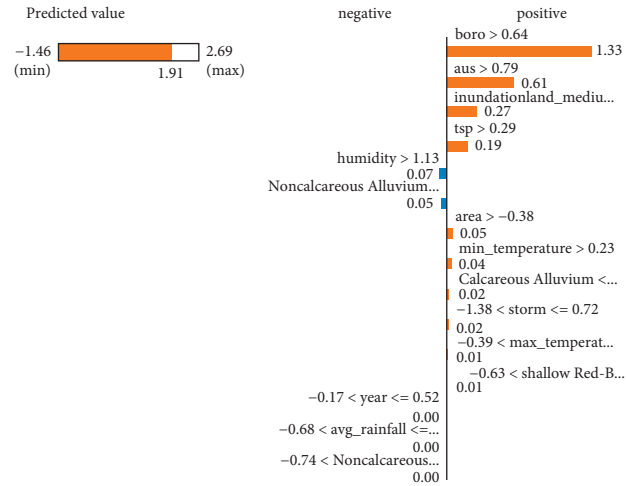

Figure 21: A plot of LIME model values for the SVR model and Aman yield data.

Table 6: The exact value and the predicted value of Boro yield in the different districts in different years.

Year	District/sub district	Exact value of Boro yield (in metric tons)	Exact value of Boro yield (after implementing standard scaler)	Predicted value of Boro yield in XGBoost (after implementing standard scaler)	Predicted value of Boro yield in SVR (after implementing standard scaler)
2014	Kishoregonj	707864	0.666120	0.6954153	0.46376287
2016	Mymenshing	1074189	1.688652	1.2599084	2.37932189
2013	Mymensing	936015	1.302964	1.4743396	1.6461434
2012	Gazipur	250778	−0.609756	−0.65754485	−0.57692833
2015	Gazipur	231718	−0.662959	−0.6235179	−0.72536924
2009	Narsingdi	212482	−0.716653	−0.5726305	−0.62422939
2014	Gazipur	228161	−0.672888	−0.5806098	−0.74941032
2016	Narsingdi	197940	−0.757245	−0.68641835	−0.84958871
2017	Mymenshing	1072834	1.684870	1.3707266	2.30537822

1.90912547. Using an interpretable linear model, the LIME model finds a local approximation. We may see a list of explanations that indicate the contribution of each feature to the prediction of the above-mentioned data sample. This enables for local interpretation and the determination of modifications in the features Boro, Aus, inundationland_mediumlowland, TSP, area, and min_temperature which will have the most influence on the prediction.

4.3. Result Analysis of Boro Yield Prediction. Just like we have discussed in the result section of Aus and Aman yield prediction, after using standard scaling, some of the values of the exact value and the predicted value have become negative for Boro yield values, which is shown in Table 6. As previously shown in both Aus and Aman yield, some random records are also shown for Boro yield.

In Table 6, we have put some data from 2008 to 2017 to get an overall idea of the results. Table 6 shows that the exact value of Boro yield (after applying standard scaler) in Kishoregonj district in 2014 is 0.666120, and we get 0.6954153 as the projected value using the XGBoost regression model and 0.46376287 as the forecast value using the SVR model. The XGBoost predicted value is considerably closer to the true value of the yield than the SVR model predicted value. We can also see that the exact value of Aman yield (after applying Standard Scaler) in the Gazipur district in 2015 is −0.662959, and we obtain −0.6235179 as the predicted value when using the XGBoost regression model and −0.72536924 while using the SVR model. Similar to the previous sample data, the XGBoost model's predicted value is closer to the true value of the yield than the SVR model.

Figure 22 shows a visual representation of the real and predicted values of the Boro yield for the XGBoost model, while Figure 23 shows a visual representation of the real and forecasted values of the Boro yield for the SVR model. We can see that the anticipated Boro yield levels are pretty comparable to the actual values in both figures.

For the XGBoost and SVR models, respectively, Figure 24 uses box plots to compare the actual value and projected value of the Boro yield forecast. As shown in Figure 24, the minimum scaled real value for Boro yield using XGBoost is −0.757245, and the minimum scaled forecasted value is −0.68641835. Real and Figure 25 predicted Boro yield first quartile (Q1) values are, respectively, −0.716653 and −0.5806098. The median values are −0.672888 and −0.5209967, respectively. The actual and predicted values had third quartile (Q3) values of 0.646698

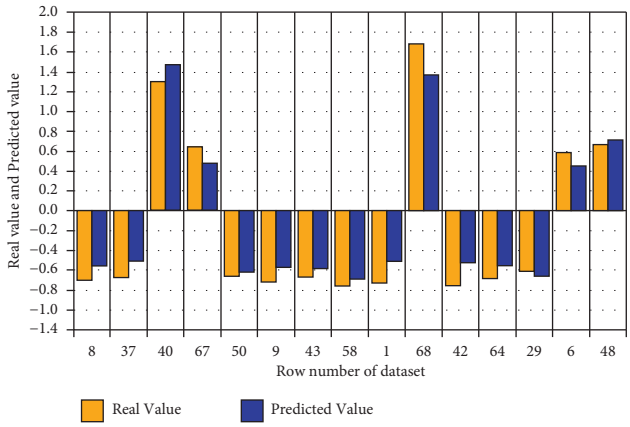

Figure 22: Comparison between real values and predicted values of Boro yield for XGBoost.

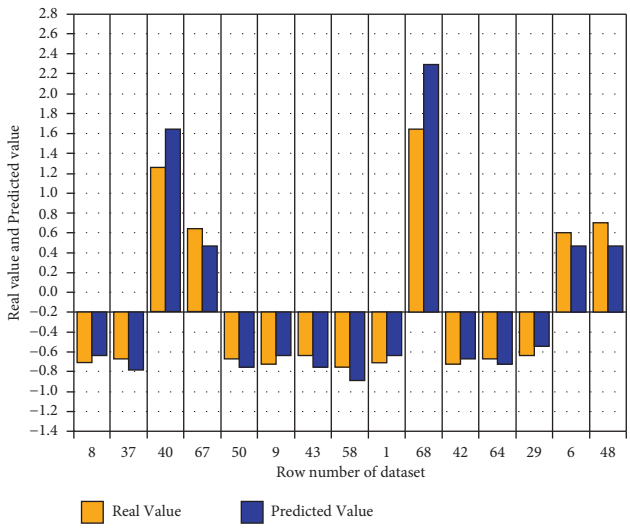

Figure 23: Comparison between real values and predicted values of Boro yield for SVR.

and 0.4803284, respectively. When employing XGBoost, the real and predicted maximum values of the Boro yield are 1.68487 and 1.4743396, respectively.

From Figure 25, we can observe the minimum, first quartile, median, third quartile, and maximum values of the actual and anticipated values of Boro yield when using the SVR model. The true values' minimum, first quartile (Q1), median, third quartile (Q3), and maximum values, respectively, are −0.757245, −0.716653, −0.672888, 0.646698, and 1.68487. While the minimum, first quartile (Q1), median, third quartile (Q3), and maximum values for the projected values using SVR are, respectively, −0.84958871, −0.74941032, −0.62422939, 0.48604702, and 2.30537822.

We have found better results in root mean square error (RMSE) while using SVR compared to XGBoost for Boro yield prediction. From Table 7, we can observe that the root mean square error is 0.227616 while using XGBoost. But in SVR, the error is 0.232760, which is slightly more than

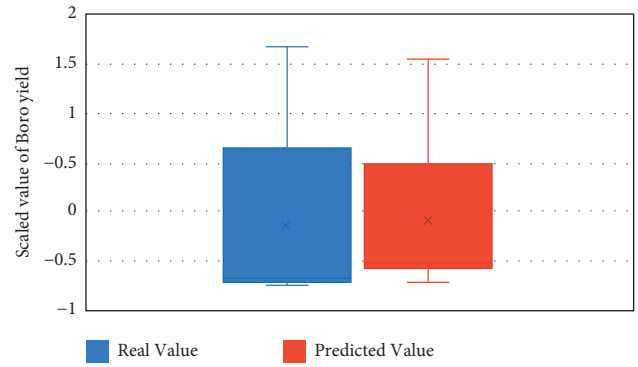

Figure 24: Comparison between real values and predicted values of Boro yield for XGBoost using box plot.

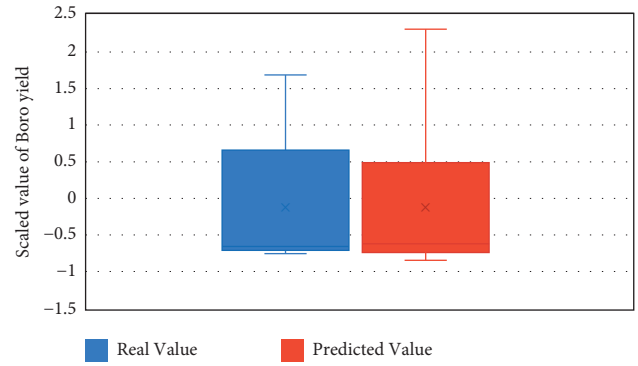

Figure 25: Comparison between real values and predicted values of Boro yield for SVR using box plot.

Table 7: RMSE and r^2 value after Boro yield prediction.

	Root mean square error	r^2 value
XGBoost	0.227616	0.934234
Support vector machine	0.232760	0.931228

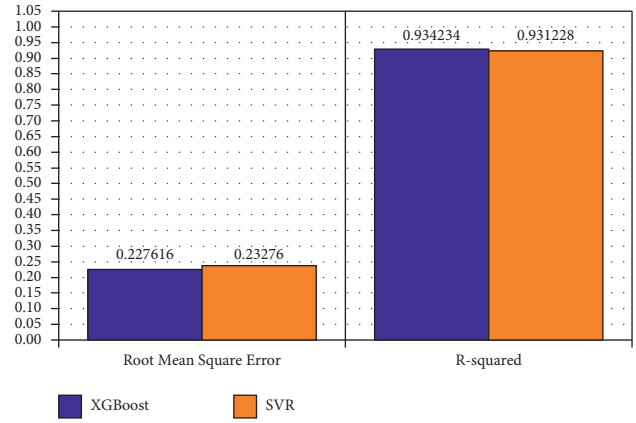

Figure 26: Comparison between the RMSE and R-squared values of XGBoost and SVR for Boro yield Prediction.

XGBoost. In XGBoost, the r^2 value we have got is 0.934234. In SVR, the r^2 value becomes 0.931228. So, XGBoost gives slightly better results for Boro rice yield prediction

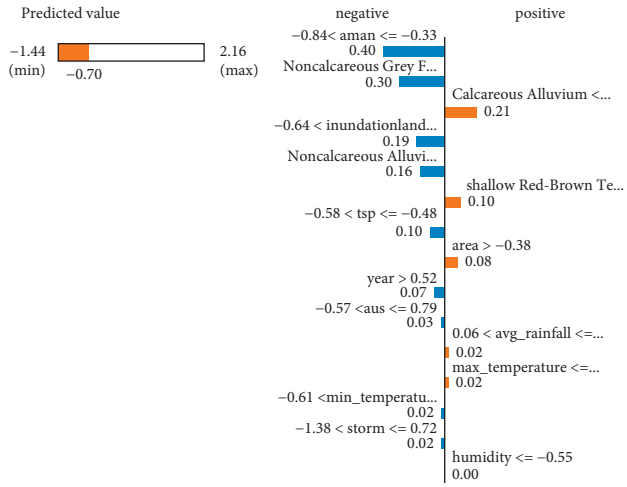

Figure 27: A plot of LIME model values for the XGBoost model and Boro yield data.

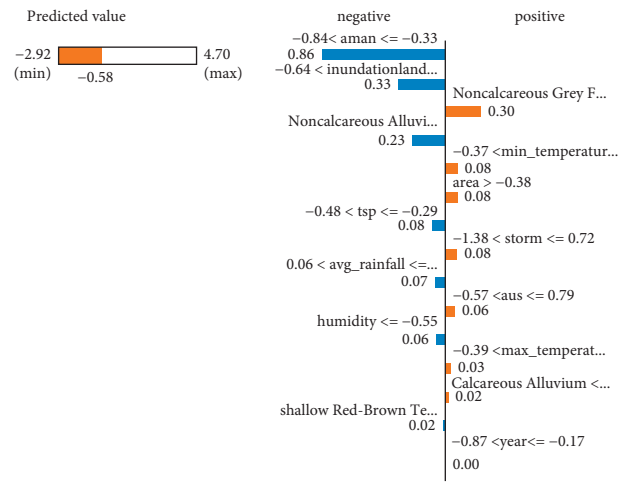

Figure 28: A plot of LIME model values for the SVR model and Boro yield data.

compared to SVR, as we can clearly observe from the bar chart representation in Figure 26.

For the interpretability of both the ML models for Boro yield data, LIME can be used to interpret a model around specific observations. In the following example (Figure 27), we explain the behavior of the XGBoost model for the prediction of Boro yield for a particular data sample where the real value is −0.66295908 and the predicted value is −0.69751798. The LIME model finds a local approximation with an interpretable linear model. From Figure 27, we can observe a list of explanations, reflecting the contribution of each feature to the prediction of the particular data sample mentioned above. This provides local interpretability, and it also allows us to determine that the changes in the features aman, noncalcareous grey floodplain soil, inundationland_mediumlowland, noncalcareous alluvium, TSP, and Aus will have the most impact on the prediction.

In Figure 28, we can observe the behavior of the SVR model for the prediction of Boro yield for a particular data sample where the real value is −0.60975640, which gives a predicted value of −0.57692833. The LIME model finds a local approximation with an interpretable linear model. We can observe a list of explanations, reflecting the contribution of each feature to the prediction of the particular data sample mentioned above. This provides local interpretability, and it also allows us to determine that the changes in the features aman, inundationland_mediumlowland, noncalcareous alluvium, TSP, and humidity will have the greatest impact on the prediction.

5. Discussion

The presented results show the performance of the proposed model for the yield prediction of three rice ecotypes (Aus, Aman, and Boro) using a dataset that included meteorological, agro-chemical, and soil physiographic parameters. We have observed that all three parameters have a role in the yield forecast while utilizing LIME to demonstrate explainability. In terms of prediction, the ML models we deployed produced fairly decent results.

In the case of the comparison between the two ML models, XGBoost predicts more accurate yield values for the three main rice ecotypes (Aus, Aman, and Boro) according to the results. It has a better fit to the data than SVR. One of the reasons for XGBoost doing well is that it is an ensemble method. It makes decisions using multiple trees; therefore, it builds strength by repeating itself. We also have the base learner splitting the solution space into a collection of hyperrectangles for every tree ensemble. The ensemble will perform well if this is close to the underlying truth, or even an adequate approximation. Additionally, it makes use of parallel processing. It also supports regularization, which helps to reduce the adjusted loss function and avoid overfitting and underfitting.

6. Conclusion

This study successfully developed a system that can predict the yield for all three different rice ecotypes of Southeast Asia and can also produce the explainability of the system. Along with the conventional data preprocessing, the main focus was on feature selection. Two different approaches were applied to finally reduce the feature number to 10. Initially, heat map was used to reduce the number of features from 43 to 16, and then finally, RFE was applied to further reduce the feature count to 10. The sets of features selected for each kind of rice ecotype were different for both the ML models, and there were a total of 6 sets of features for three rice ecotypes and two ML models. However, there were some common features among the six sets of features, and those were the principal features for the yield prediction. The RMSE and R-squared values for each type of rice ecotype for both the ML models were satisfactory. Additionally, an XAI model called LIME was applied to show the explainability of the two ML models. The most impactful features for yield prediction of different rice types were found while using these two different ML models. The most impactful features are a combination of meteorological, agro-chemical, and soil

physiographic factors and do not contain only one type of feature. Therefore, the proposed system can help in yield prediction for three different rice ecotypes not only in a precise manner but also provides an explanation behind those prediction models. Apart from the model's positive aspects, it also has a certain drawback. The study focused on a few distinct regions in Bangladesh. It would be preferable to include each district of Bangladesh. There are some possible future directions along with this study. There is an opportunity to do further research by adding more sections to develop the system. For instance, water management, seed management, and soil management can be added to demonstrate "precision agriculture" in terms of paddy more precisely. Furthermore, other factors that may contribute to produce better yield are disease detection and pesticide management. A large amount of out-turn is lost every year to the attack of different diseases and the usage of improper pesticides. In the future, these two fields can be explored. Moreover, other ML techniques can be applied in the future for better accuracy, and we can also apply other explainable AI models to show the interpretability of the models used. Furthermore, in the future, this concept of yield prediction and explainability of the model used can be applied to other crops as well.

Data Availability

Data used to support the findings of the study are available in the manuscript.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by King Saud University, Riyadh, Saudi Arabia, under researchers supporting project number RSP-2022/18.

References

[1] Ricepedia". International Rice Research Institute, "AfricaRice, and the International Center for Tropical Agriculture," https://ricepedia.org/rice-as-food/the-global-staple-rice-consumers.

[2] X. Yang, L. Shu, J. Chen et al., "A survey on smart agriculture: development modes, technologies, and security and privacy challenges," *IEEE/CAA Journal of Automatica Sinica*, vol. 8, no. 2, pp. 273–302, 2021.

[3] R. Sobayo, H. H. Wu, R. Ray, and L. Qian, "Integration of convolutional neural network and thermal images into soil moisture estimation," in *Proceedings of the 2018 1st International Conference on Data Intelligence and Security (ICDIS)*, pp. 207–210, IEEE, South Padre Island, TX, USA, May 2018.

[4] S. k. A. l. Z. Rahman, K. C. Mitra, and S. M. M. Islam, "Soil classification using machine learning methods and crop suggestion based on soil series," in *Proceedings of the 2018 21st International Conference of Computer and Information Technology (ICCIT)*, pp. 1–4, IEEE, Dhaka, Bangladesh, December 2018.

[5] K. Kaur and K. Singh Attwal, "Effect of temperature and rainfall on paddy yield using data mining," in *Proceedings of the 2017 7th International Conference on Cloud Computing, Data Science & Engineering-Confluence*, pp. 506–511, IEEE, Noida, India, January 2017.

[6] M. B. Hossain, D. Roy, M. d. Maniruzzaman et al., "Response of crop water requirement and yield of irrigated rice to elevated temperature in Bangladesh," *International Journal of Agronomy*, vol. 2021, pp. 2021–11.

[7] L. Wickramasinghe, R. Weliwatta, P. Ekanayake, and J. Jayasinghe, "Modeling the relationship between rice yield and climate variables using statistical and machine learning techniques," *Journal of Mathematics*, vol. 2021, pp. 2021–9.

[8] V. Amaratunga, L. Wickramasinghe, A. Perera, J. Jayasinghe, and U. Rathnayake, "Artificial neural network to estimate the paddy yield prediction using climatic data," *Mathematical Problems in Engineering*, vol. 2020, pp. 1–11.

[9] J. Shook, T. Gangopadhyay, L. Wu, B. Ganapathysubramanian, S. Sarkar, and A. K. Singh, "Crop yield prediction integrating genotype and weather variables using deep learning," *PLoS One*, vol. 16, no. 6, Article ID e0252402, 2021.

[10] R. B. Guruprasad, S. Kumar, and S. Randhawa, "Machine learning methodologies for paddy yield estimation in India: a case study," in *Proceedings of the IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*, pp. 7254–7257, IEEE, Yokohama, Japan, Augest 2019.

[11] T. Takayama, A. Uchida, H. Sekine et al., "Validation of bipls for improving yield estimation of rice paddy from hyperspectral data in west java, Indonesia," in *Proceedings of the 2012 IEEE International Geoscience and Remote Sensing Symposium*, pp. 6581–6584, IEEE, Munich, Germany, July 2012.

[12] M. R. S. Muthusinghe, S. T. Palliyaguru, W. Wand, A. M. H. Saranga, and W. H. Rankothge, "Towards smart farming: accurate prediction of paddy harvest and rice demand," in *In Proceedings of the 2018 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)*, pp. 1–6, IEEE, Malambe, Sri Lanka, December 2018.

[13] N. Liundi, A. W. Darma, R. Gunarso, and H. L. H. S. Warnars, "Improving rice productivity in Indonesia with artificial intelligence," in *In Proceedings of the 2019 7th International Conference on Cyber and IT Service Management (CITSM)*, pp. 1–5, IEEE, Jakarta, Indonesia, November 2019.

[14] X. Han, F. Liu, X. He, and F. Ling, "Research on rice yield prediction model based on deep learning," *Computational Intelligence and Neuroscience*, vol. 2022, pp. 1–9.

[15] P. W. Koh and P. Liang, "Understanding black-box predictions via influence functions," in *Proceedings of the International Conference on Machine Learning*, pp. 1885–1894, PMLR, 2017.

[16] M. Bojarski, P. Yeres, A. Choromanska et al., "Explaining how a deep neural network trained with end-to-end learning steers a car," 2017, http://arxiv.org/abs/1704.07911.

[17] M. T. Ribeiro, S. Singh, and C. Guestrin, ""Model-agnostic interpretability of machine learning," 2016, http://arxiv.org/abs/1606.05386?context=cs.LG.

[18] D. Doran, S. Schulz, and T. R. Besold, "What does explainable ai really mean? a new conceptualization of perspectives," 2017, http://arxiv.org/abs/1710.00794.

[19] A. Adadi and M. Berrada, "Peeking inside the black-box: a survey on explainable artificial intelligence (xai)," *IEEE Access*, vol. 6, pp. 52138–52160, 2018.

[20] W. m. r. K. Wanninayaka, R. M. K. T. Rathnayaka, and E. P. N. Udayakumara, "Artificial neural network to estimate

the paddy yield prediction using remote sensing, weather and non weather variable in ampara district, Sri Lanka," in *Proceedings of the 2020 5th International Conference on Information Technology Research (ICITR)*, pp. 1–6, IEEE, Moratuwa, Sri Lanka, December 2020.

[21] Md S. Alam, K. Kalpoma, Md S. Karim, A. Al Sefat, and J.-ichi Kudoh, "Boro rice yield estimation model using modis ndvi data for Bangladesh," in *Proceedings of the IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*, pp. 7330–7333, IEEE, Yokohama, Japan, Augest 2019.

[22] Y. Alebele, W. Wang, W. Yu et al., "Estimation of crop yield from combined optical and sar imagery using Gaussian kernel regression," *Ieee Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 14, pp. 10520–10534, 2021.

[23] Md A. Hossain, M. N. Uddin, M. Arif Hossain, and Y. M. Jang, "Predicting rice yield for Bangladesh by exploiting weather conditions," in *Proceedings of the 2017 International Conference on Information and Communication Technology Convergence (ICTC)*, pp. 589–594, IEEE, Jeju Island, South Korea, October 2017.

[24] Z. Zhuo, C. Gao, and Y. Liu, "Regional grain yield response to climate change in China: a statistic modeling approach," *Ieee Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 7, no. 11, pp. 4472–4479, 2014.

[25] N. Gandhi, O. Petkar, and L. J. Armstrong, "Rice crop yield prediction using artificial neural networks," in *Proceedings of the 2016 IEEE Technological Innovations in ICT for Agriculture and Rural Development (TIAR)*, pp. 105–110, IEEE, Chennai, Tamil Nadu, India, July 2016.

[26] A. Chakrabarty, N. Mansoor, M. I. Uddin, M. H. Al-adaileh, N. Alsharif, and F. W. Alsaade, "Prediction approaches for smart cultivation: a comparative study," *Complexity*, vol. 2021, pp. 2021–16.

[27] N. Bali and A. Singla, "Emerging trends in machine learning to predict crop yield and study its influential factors: a survey," *Archives of Computational Methods in Engineering*, vol. 29, no. 1, pp. 95–112, 2022.

[28] S. A. Shetty, T. Padmashree, B. M. Sagar, and N. K. Cauvery, "Performance analysis on machine learning algorithms with deep learning model for crop yield prediction," in *Data Intelligence and Cognitive Informatics*, pp. 739–750, Springer, 2021.

[29] J. Cao, Z. Zhang, F. Tao et al., "Integrating multi-source data for rice yield prediction across China using machine learning and deep learning approaches," *Agricultural and Forest Meteorology*, vol. 297, Article ID 108275, 2021.

[30] Md M. Islam, S. Matsushita, R. Noguchi, and T. Ahamed, "Development of remote sensing-based yield prediction models at the maturity stage of boro rice using parametric and nonparametric approaches," *Remote Sensing Applications: Society and Environment*, vol. 22, Article ID 100494, 2021.

[31] S. Ju, H. Lim, J. W. Ma et al., "Optimal county-level crop yield prediction using modis-based variables and weather data: a comparative study on machine learning models," *Agricultural and Forest Meteorology*, vol. 307, Article ID 108530, 2021.

[32] Agricultural Dataset Bangladesh(44 Parameters): https://www.kaggle.com/tanhim/agricultural-dataset-bangladesh-44-parameters.

[33] T. Chen and C. Guestrin, "Xgboost: a scalable tree boosting system," in *Proceedings of the 22nd Acm Sigkdd International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, New York, NY ACM, November 2016.

[34] C. Bentéjac, A. Csörgő, and G. Martínez-Muñoz, "A comparative analysis of gradient boosting algorithms," *Artificial Intelligence Review*, vol. 54, no. 3, pp. 1937–1967, 2021.

[35] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial," *Frontiers in Neurorobotics*, vol. 7, no. 21, 2013.

[36] J. H. Friedman, "Stochastic gradient boosting," *Computational Statistics & Data Analysis*, vol. 38, no. 4, pp. 367–378, 2002.

[37] R. E. Schapire, *Explaining adaboost Empirical inference*, pp. 37–52, Springer, Berlin, Germany, 2013.

[38] T. Chen, T. He, M. Benesty et al., "Xgboost: extreme gradient boosting," *R package version 0*, vol. 1, no. 4, 2015.

[39] M. Awad and R. Khanna, *Support vector regression". Efficient Learning Machines*, pp. 67–80, Springer, Berlin, Germany, 2015.

[40] Max Welling, "Support vector regression". Department of computer science," *University of Toronto, Toronto (Kanada)*, 2004.

[41] A. Messalas, C. Aridas, and Y. Kanellopoulos, "Evaluating mashap as a faster alternative to lime for model-agnostic machine learning interpretability," in *Proceedings of the 2020 IEEE International Conference on Big Data (Big Data)*, pp. 5777–5779, IEEE, Atlanta, GA, USA, December 2020.

[42] Jürgen Dieber and Sabrina Kirrane, "Why model why? assessing the strengths and limitations of lime"," 2020, http://arxiv.org/abs/2012.00093.