https://dx.doi.org/10.4314/dujopas.v10i2c.36
ISSN (Print): 2476-8316
ISSN (Online): 2635-3490
Dutse Journal of Pure and Applied Sciences (DUJOPAS), *Vol. 10 No. 2c June 2024*

Ensemble-Based Predictive Model for Crop Recommendation

Morufu Olalere[2], Gilbert I.O. Aimufua[2], Muhammad Umar Abdullahi[1], Bako Halilu Egga[2]

[1]Department of Computer Science,
Federal University of Technology,
Owerri,
Nigeria.

[2]Department of Computer Science,
Nasarawa State University,
Keffi,
Nigeria.

Email: umarfaruq54@gmail.com

Abstract

Agriculture is a vital industry that supplies food, textiles, and other basic goods to people globally. Agricultural crop production has a vital role in influencing the economy and the well-being of farmers. Nevertheless, farmers are facing substantial challenges due to the profound changes in environmental conditions. A significant challenge they have is determining the most suitable crop for their specific location that will optimize both production and profitability. Choosing suitable crop types for a certain area may be difficult due to the need for skills and experience in evaluating elements such as soil composition, climatic conditions, moisture levels, precipitation, and temperature. Multiple researchers have devised several approaches to tackle the issue of crop recommendation. Nevertheless, a significant share of these models is specifically tailored for a certain job or are amalgamations that include two or three machine-learning algorithms. These current models have restricted prediction accuracy and elevated rates of false positives, rendering them inappropriate for the intricacy of the job at hand. This study explores the field of precision agriculture with the objective of improving crop recommendation systems via the use of an ensemble-based prediction model. This paper incorporates KNN, Decision Tree, Random Forest, SVM, Naive Bayes, Logistic Regression, and XGBoost as a series of machine learning models. A stacked ensemble prediction model is created by training, evaluating, and comparing the Random Forest classifier with the stacked ensemble prediction model. In contrast to existing methods, the proposed method exhibits exceptionally high accuracy, reaching 99.8%, exceeding the performance of prior studies. Through the application of advanced predictive modeling techniques, this paper demonstrates how agricultural operations can be improved.

Keywords: Crop Recommendation, Model, Ensemble Learning, Stacking, Smart Agriculture

Author for Correspondence

INTRODUCTION

The agricultural industry is of paramount importance to our economy, and bolstering this sector may yield favorable economic and political consequences for our nation (Aliyev, Babayev, Galandarova, Gafarli & Balajayeva, 2023). Technological improvements, biological effects, and environmental circumstances have a significant impact on the health and productivity of crops (Enerijiofi, Musa, Okolafor, Igiebor, Odozi, & Ikhajiagbe, 2023). Agriculture is a vital industry that supplies food, fiber, and other necessary goods to people globally. Agricultural crop production has a pivotal role in influencing the economy and the well-being of farmers (FAO, 2020). Nevertheless, farmers are facing considerable hurdles due to the substantial changes in environmental conditions. An important challenge they have is choosing the most suitable crop for their location that will optimize production and financial gains (Fischer & Connor, 2018). Choosing the suitable crop kinds for a certain area may be difficult due to the need for skill and experience in evaluating characteristics including soil type, climate, humidity, rainfall, and temperature (Kephe, Ayisi & Petja, 2021). Furthermore, conventional techniques for suggesting crops may not consistently provide precise or current information, resulting in less-than-optimal crop production and heightened expenses for farmers (Munaweera, Jayawardana, Rajaratnam, Dissanayake, 2022).

Consequently, there is an urgent need to enhance agricultural methodologies in order to guarantee long-term viability (Pawlak & Kołodziejczak, 2020). Moreover, a lack of adequate technical proficiency and volatile weather patterns resulted in a decrease in yearly agricultural output over the majority of the globe (Gopi & Karthikeyan, 2023). Hence, it is essential to identify appropriate crops that may effectively boost productivity and production in order to fulfill the growing global food demand. Accurate forecast of agricultural output is crucial and depends on several aspects such as irrigation systems, weather conditions, and geographical location (Reddy & Kumar, 2023). An effective approach to address these difficulties is using machine learning algorithms to forecast the appropriateness and yield of crops by considering environmental variables (Durai & Shamili, 2022).

ML techniques has the capability to tackle this difficulty. Machine learning algorithms may use data-driven analysis to evaluate information and provide customized agricultural suggestions (Huang, Srivastava, Ngo, Gao, Wu, & Chiao, 2023). Utilizing machine learning algorithms for crop recommendation has great potential in improving agricultural production and sustainability. The growing availability of data and advancements in machine learning algorithms are anticipated to bolster their importance in the agricultural sector. This methodology offers farmers precise and current data regarding crop selection, optimal planting time, and allocation of resources. Consequently, farmers can make well-informed choices that can result in enhanced yields, decreased expenses, and improved sustainability (Dhanaraju, Chenniappan, Ramalingam, Pazhanivelan, & Kaliaperumal, 2022). Despite prior research on the subject, most studies have mostly examined single-task learning models and have not thoroughly investigated the capabilities of ensemble learning approaches (Rashid et al., 2021; Devan, Swetha, Sruthi & Varshini, 2023).

Ensemble learning approaches, a potent paradigm in machine learning, include amalgamating the predictions of many models to enhance overall performance and resilience (Ganaie et al., 2022). Ensemble approaches use a heterogeneous collection of many models rather than depending on a single model to overcome individual limitations and improve the precision of predictions (Guo, Wang, Xiao & Xu, 2020).

Author for Correspondence

The objective of this study is to develop a prognosis model for crop recommendation. The objective will be achieved by the use of an ensemble-based methodology that integrates seven discrete machine learning models: K-Nearest Neighbor, Decision Tree, Support Vector Machine, Random Forest, Logistic Regression, Naïve Bayes, and XGBoost. The model assists farmers in selecting suitable crops that are compatible with their specific soil and climatic conditions, hence leading to improved agricultural production. The aim of this model is to improve the precision and dependability of crop recommendations. This is achieved by using the capabilities of many algorithms and overcoming their constraints via ensemble methods. Furthermore, the model may aid farmers in adopting sustainable farming practices by reducing the reliance on harmful chemicals and promoting the use of organic fertilizers (Çakmakçı, Salık & Çakmakçı, 2023).

The finding has potential benefits that extend beyond the economic rewards for farmers. This approach has the capability to augment agricultural output and durability, resulting in significant socioeconomic and environmental benefits. The model's ability to provide accurate and up-to-date information enables farmers to make well-informed decisions, leading to enhanced crop yield, reduced dependence on harmful pesticides, and the promotion of sustainable farming practices. As a result, this might improve the farmers' and their families' standard of living, while simultaneously promoting environmental conservation.

METHODOLOGIES

The research methodology is the central framework that encompasses the techniques and procedures used to acquire, gather, and assess data, all of which are closely linked to the topic of study (Garg, 2016; Mohajan, 2018). The research methodology of the proposed model is shown in Figure 1 and consists of many sequential phases (Busetto, Wick & Gumbinger, 2020). The procedure involves collecting the dataset, examining and preparing the data, dividing the dataset, training the model, evaluating the model, measuring its performance, comparing the performance, and formulating a conclusion.

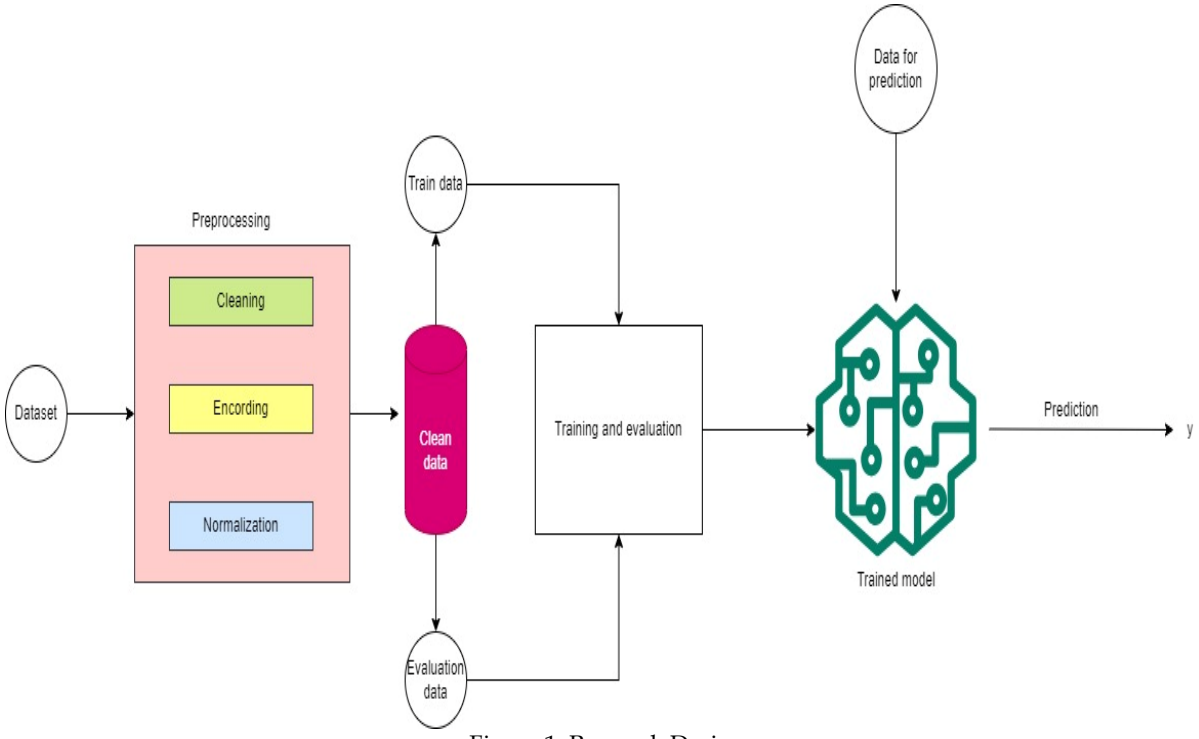

Figure 1: Research Design
Source: Author based on (Rana *et al.,* 2015)

i. **Dataset Collection:** The first stage involves gathering the dataset in CSV format and importing it into Jupiter Notebook.
ii. **Data exploration and pre-processing:** Perform correlation analysis, eliminate features with negative correlation, address missing values, eliminate duplicates, normalize the data, and scale the features.
iii. **Splitting dataset:** This refers to the process of dividing the dataset into two separate parts. One is designated for training purposes, while the other is specifically used for testing.
iv. **Training the models:** Models are trained using K-Nearest Neighbor, Decision Tree, Support Vector Machine, Random Forest, Logistic Regression, Naïve Bayes, and XGBoost algorithms.
v. **Testing the models:** The analyzed models include K-Nearest Neighbor, Decision Tree, Support Vector Machine, Random Forest, Logistic Regression, Naïve Bayes, and XGBoost.
vi. **Performance evaluation:** The evaluation of the model's performance was conducted using the accuracy score and confusion matrix given by Scikit-learn.
vii. **Models Ensembling:** this involves combining predictions from all the multiple models to improve overall performance.
viii. **Conclusion:** This is the ultimate stage when the ensemble model is presented as the most optimal model for forecasting appropriate crops for a certain area of land.

Source of Dataset

The dataset was obtained from https://www.kaggle.com/datasets/aksahaha/crop recommendation and includes data on the amounts of nitrogen, phosphorus, and potassium in the soil as well as measurements of temperature, humidity, pH, and rainfall and how they affect the growth of crops (Nti *et al.,* 2023). This dataset can be utilized to create data-based suggestions for achieving the best possible nutrient and environmental conditions to enhance crop yield (Gosai *et al*., 2021). The data size was 2200 records and seven predictors (Muhammed, Ahvar, Ahvar & Trocan, 2023). The target variable consists of twenty-two classes representing different crops (i.e., 'mungbean', 'apple', 'kidney-beans', 'banana' 'maize', 'blackgram', 'chickpea', 'mothbeans', 'coconut','coffee', 'cotton', 'grapes', 'jute', 'pigeonpeas', 'papaya', 'mango', 'lentil','muskmelon', 'orange', 'watermelon', 'pomegranate' and 'rice') each with one hundred (100) samples.

Formulation of Ensemble-Based Predictive Model for Crop Recommendation

The mathematical model aims to represent the flow of which a model for the prediction of crops and their yields can be developed.

Let the soil attributes of Nitrogen, Phosphorus, and Potassium be denoted as N, P, and K correspondingly. Thus, we may describe their sets as:

$$\boldsymbol{N} = \{\boldsymbol{n_1}, \boldsymbol{n_2}, \dots, \boldsymbol{n_3}\} \quad (1)$$

$$\boldsymbol{P} = \{\boldsymbol{p_1}, \boldsymbol{p_2}, \dots, \boldsymbol{p_3}\} \quad (2)$$

$$K = \{k_1, k_2, \dots, k_3\} \quad (3)$$

Since the combined values of $N, P\ and\ K$ provide Soil (S) properties data;

$$S = \sum_{i=1}^{3} n_i + \sum_{i=1}^{3} p_i + \sum_{i=1}^{3} k_i \quad (4)$$

whereby S can have a varying value based on the values of $N, P\ and\ K$ at any given time, therefore:

$$S = \{s_1, s_2, \dots, s_3\} \quad (5)$$

We can further have PH value and the Soil(S) properties to form the environmental factor(E):

$$E = \sum_{i=1}^{3} ph_i + \sum_{i=1}^{3} s_i \quad (6)$$

Whereby $E = \{e_1, e_2, \dots, e_3\}$ based on the value of PH and S at a given time.

Let $T, R\ and\ H$ represent the set of Temperature, Rainfall, and Humidity.

$$T = \{t_1, t_2, \dots, t_3\} \quad (7)$$

$$R = \{r_1, r_2, \dots, r_3\} \quad (8)$$

$$H = \{h_1, h_2, \dots, h_3\} \quad (9)$$

The combination of $T, R\ and\ H$ will then provide the Weather(W) factor

$$W = T + R + H \quad (10)$$

We can further have a dataset(Q) that will enable the prediction of crops based on $W\ and\ E$

$$Q = W + E \quad (11)$$

$$Q = \{q_1, q_2, \dots, q_3\} \quad (12)$$

Let n represent the SK-Learn data cleaning model, the function of n on Q will therefore be $n(Q)$ producing dataset($d1$)

$$dataset(d1) = n(Q) \quad (13)$$

Base Model Predictions

For this study, a total of seven machine-learning models were chosen. Let NB, SVM, KNN, LR, DT, XGBOOST, and RF represent Naïve Bayes, Support Vector Machine, K-Nearest Neighbor, Logistic Regression, Decision Tree, and extreme Gradient Boosting, respectively. The functions nb(d1), svm(d1), knn(d1), lr(d1), dt(d1), rf(d1), and xgboost(d1) correspond to the algorithms NB, SVM, KNN, LR, DT, XGBOOST, and RF, respectively. As a result, we have:

$$Crops\ predicted\ (C_{nb}) = nb(d1) \quad (14)$$
$$Crops\ predicted\ (C_{svm}) = svm(d1) \quad (15)$$
$$Crops\ predicted\ (C_{knn}) = knn(d1) \quad (16)$$
$$Crops\ predicted\ (C_{lr}) = lr(d1) \quad (17)$$
$$Crops\ predicted\ (C_{dt}) = dt(d1) \quad (18)$$
$$Crops\ predicted\ \left(C_{xgboost}\right) = xgboost(d1) \quad (19)$$
$$Crops\ predicted\ \left(C_{rf}\right) = rf(d1) \quad (20)$$
where $C = \{c_1, c_2, \dots, c_3\}$ (21)

Stacking Predictions

Stacking predictions involves creating a new feature matrix by combining the predictions from all base models. In the context of the seven base models (KNN, DT, NB, LR, XGBoost, RF, SVM), The stacked predictions into a new feature matrix.

Let's denote the predictions from each model as

$$X_{ensemble} = [C_{nb}, C_{svm}, C_{knn}, C_{lr}, C_{dt}, C_{xgboost}] \quad (22)$$

The stacked ensemble matrix $X_{ensemble}$ would be:

$$X_{ensemble} = \begin{bmatrix} C_{nb} \\ C_{svm} \\ C_{knn} \\ C_{lr} \\ C_{dt} \\ C_{xgboost} \end{bmatrix} \quad (23)$$

This new feature matrix is then used as input for the meta-model (Random Forest in this case) to make final predictions.

Random Forest Meta-Model

The Random Forest meta-model is trained on the ensemble feature matrix using the true labels Y, represented as $X_{ensemble}$:

$$Ensemble_{model} = C_{rf}(X_{ensemble}, Y) \quad (24)$$

Equation (24) demonstrates how the Random Forest meta-model is trained by using the combined predictions from the underlying models (X_ensemble) and the actual labels (Y). Once the model is created, it is designated as the Ensemble_model and will be used to generate predictions on fresh data.

Prediction from Ensemble Model

For a new input feature X, obtain predictions from each base model and stack them as before:

$$\hat{Z}_{ensemble}\left[C_{nb}(X), C_{svm}(X), C_{knn}(X), C_{lr}(X), C_{dt}(X), C_{xgboost}(X)\right] \quad (25)$$

$$\hat{Z}_{final} = [Ensemble_{model}\ predict(\hat{Z}_{ensemble})] \quad (26)$$

The ensemble model leverages the strength of the base models of each of the base models and uses the Random Forest to combine their predictions for a more robust final prediction.

Algorithm of the Ensemble-Based Predictive Model for Crop Recommendation

The Stacked Ensemble model for crop recommendation, as described in Algorithm 1, seeks to improve predicted accuracy by using the capabilities of numerous base models.

Algorithm 1: Algorithm for Crop Ensemble-Based Model

1. Start
2. Split the training Data (TD) into $N\ fold \rightarrow D_1, D_2, \ldots D_N$
3. For each base model, $m \in SVM, KNN, LR, DT, NB\ and\ XGBOOST$ {
   Train $T_{models}, m_1, m_2, \ldots m_N$ on $N-1$ folds of the RF
   Keep the predicted outputs $\hat{Z}_{m,t}(X)$ for m_t on each test $fold\ D_n$

   }
4. Concatenate the predicted output from all base models for each test fold:
5. $X_k = [\hat{Z}SVM, 1(D_n), \hat{Z}SVM, 2(D_n), \ldots yXGBOOST, T(D_n)]$

6. Train a metal model $f(X)$ on the concatenate the predicted output X_n for each fold.
7. For each test fold D_n use the base models to predict the output Ẑm(x) and concatenate the outputs:
   X= $\{\hat{Z}SVM(D_n), \hat{Z}KNN(D_n), \hat{Z}LR(D_n), \hat{Z}DT(D_n),$
   $\hat{Z}NB(D_n), \hat{Z}XGBOOST(D_n)\}$

8. Use the trained meta-model to predict the final output: $\hat{Z}_{final}(x) = f(x)$

9. End

At first, the training data is divided into N folds for cross-validation. Afterward, several basic models such as SVM, KNN, LR, DT, NB, and XGBOOST are trained on subsets of the data using N-1 folds of a Random Forest. The outputs of these basis models are combined by concatenating them and utilized as input for a meta-model, which is trained to maximize the combination of the base model outputs. In the final prediction phase, the basic models are again used to forecast fresh data, while the meta-model combines these forecasts to provide the result.

Performance Evaluation Matrices

Diverse metrics are often used to assess the effectiveness of a categorization model. The measurements include Accuracy, Confusion Matrix, Precision, Recall, F1-score, Error Rate, and Training time.

Classification Accuracy: The term "accuracy" refers to the ratio of correct predictions produced by the model to the total number of predictions made.

$$\text{Accuracy } (\%) = \frac{TP+TN}{TP+TN+FP+FN} \times 100 \quad (27)$$

Precision: Precision is determined by the ratio of true positives to the total of true positives and false positives.

The formula for precision is:

$$\text{Precision} = \frac{TP}{TP+FP} \quad (28)$$

Sensitivity; Sensitivity is a numerical metric used to calculate the proportion of correctly identified positive situations that were incorrectly labeled as negative by the model. It is sometimes denoted as recall or true positive rate. Mathematically, it is defined as the ratio of the number of true positive (TP) occurrences to the sum of true positive and false negative (FN) cases.

Sensitivity = $\frac{TP}{TP+FN}$ (29)

Specificity

Specificity is a synonym for the actual negative rate. Theoretical definition of the term involves the calculation of the ratio between the number of true negative (TN) instances and the sum of true negative and false positive (FP) cases.

Mathematically, the expression is as follows:

Specificity = $\frac{TN}{TN+FP}$ (30)

F-Score

The F-score is a statistical metric used to evaluate the effectiveness of a binary classification model by measuring its capacity to reliably anticipate occurrences of the positive class. The computation employs the metrics of accuracy and recall.

It is mathematically calculated as:

F1-score = $2 * \frac{precision*recall}{precision+recall}$ (31)

Error Rate (EER)

The error rate may be computed by dividing the total count of wrong predictions made on the test set by the total count of predictions made on the test set.

Mathematically, it is expressed as:

Error Rate = $\frac{Incorrect\ Predictions}{Total\ Predictions}$ (32)

RESULTS AND DISCUSSION

Numerical Experimental Performance of the Proposed Model

This stage assesses the quantitative experimental performance of the proposed model and fine-tunes the number of samples and features. The results are well documented via the use of tables and graphs.

Importing the Libraries

The Python Libraries Pandas, Numpy, CSV, Matplotlib, Seaborn, and Joblib were loaded into Jupyter Notebook. Numpy enables fast computation and broadcasting over multi-dimensional arrays by vectorization (Stančin & Jović, 2019). Pandas is a sophisticated and intuitive open-source program designed for the analysis and manipulation of data. It is constructed using the Python programming language (Subasi, 2020). Matplotlib and Seaborn are Python libraries especially tailored for data visualization. They provide an intuitive interface for generating visually captivating and practical graphs. Seaborn is constructed upon the framework of Matplotlib and provides a slightly smaller range of functionalities in comparison to Matplotlib (Pintor et al., 2019). Figure 2 depicts the inclusion of Pandas, Numpy, CSV, Matplotlib, Seaborn, and Joblib Python libraries into the Jupyter Notebook.

```
import pandas as pd  # For data manipulation
import numpy as np  # For scientific computing
import csv
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.graph_objects as go
import joblib
```

Figure 2: Importing Python Libraries
Source: Author

Loading the Dataset

The dataset was imported into Jupyter Notebook using the pd.read_csv function. Figure 3 depicts the process of importing the dataset into Jupyter Notebook.

```
data = pd.read_csv('Crop_recommendation.csv')
data.head()
```

Figure 3: Loading the Dataset in Jupyter Notebook

Checking for Missing Values

Each attribute in the dataset was assessed for the existence of missing values, and no instances of missing values were detected. Table 1 indicates that there are no missing values (NaN) present in any of the columns. There are no missing values in any of the columns, as shown by the values in the right column (0 for each column name).

Table 1: Checking for Missing Values in the dataset
Source: Author

```
In [2]: df.isnull().sum()

Out[2]:
N              0
P              0
K              0
temperature    0
humidity       0
ph             0
rainfall       0
label          0
dtype: int64
```

Descriptive Statistics of Dataset

The *describe ()* function offers a succinct overview of a dataset including 2200 items and 7 columns, all of which consist of integer values. Table 2 presents statistical data on many parameters, such as Nitrogen, Phosphorus, Potassium, Temperature, Humidity, pH, and Rainfall. The dataset comprises statistical measurements such as the mean, standard deviation, minimum, maximum, and quartiles. These metrics provide vital insights into the mean values and variability of agricultural indicators, facilitating the analysis and detection of any anomalous data points.

Table 2: Dataset Descriptive Statistics

	Nitrogen	phosphorus	potassium	temperature	humidity	ph	rainfall
count	2200.000000	2200.000000	2200.000000	2200.000000	2200.000000	2200.000000	2200.000000
mean	50.551818	53.362727	48.149091	25.616244	71.481779	6.469480	103.463655
std	36.917334	32.985883	50.647931	5.063749	22.263812	0.773938	54.958389
min	0.000000	5.000000	5.000000	8.825675	14.258040	3.504752	20.211267
25%	21.000000	28.000000	20.000000	22.769375	60.261953	5.971693	64.551686
50%	37.000000	51.000000	32.000000	25.598693	80.473146	6.425045	94.867624
75%	84.250000	68.000000	49.000000	28.561654	89.948771	6.923643	124.267508
max	140.000000	145.000000	205.000000	43.675493	99.981876	9.935091	298.560117

Source: Author

Crops Distribution Chart

The pie chart was generated with the Matplotlib tool to visually represent the distribution of distinct values in the "label" column of the Data Frame. Figure 4 presents a pie chart that graphically represents the allocation of various crop kinds in the dataset. Every section of the visual representation corresponds to a particular crop, and the magnitude of each section shows the relative fraction of that crop in the dataset.

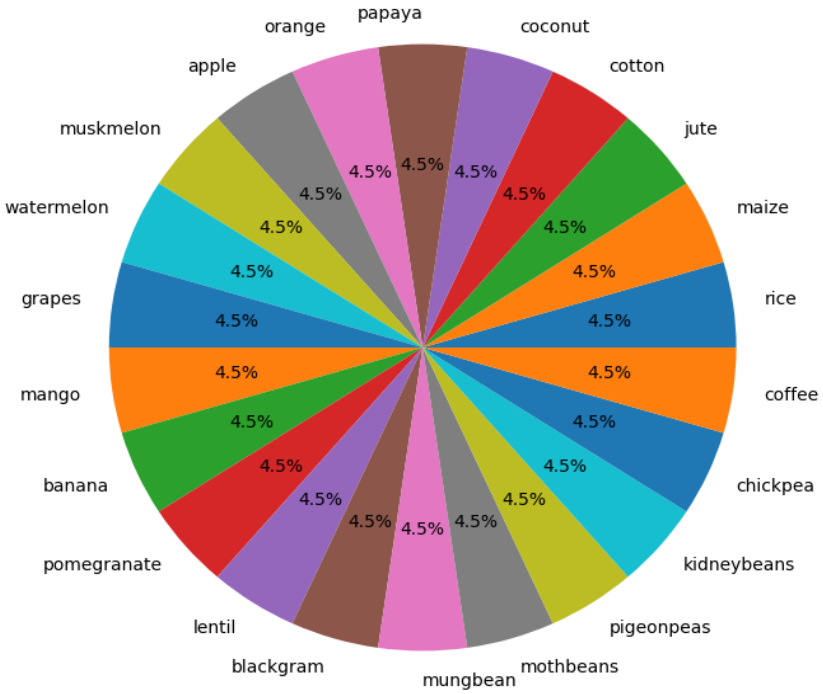

Figure 4: Crops Distribution Chart
Source: Author

RESULTS

The results extracted from the findings of the data analysis are presented here. The results are presented using figures and tables.

Encoding of Target Variable

This involves the transformation of alphabetic characters in the dataset into numerical values. Figure 5 illustrates the process of creating the Label Encoder object. This object was used to convert the category labels in the target variable y into numerical labels. The fit_transform method of the Label Encoder is used on the target variable y. This approach simultaneously adapts the encoder to the distinct labels in y and converts the labels into numerical representations. The encoded labels obtained are kept in the variable y_encoded.

```
# # Encode the target variable into numeric values
label_encoder = LabelEncoder()
y_encoded = label_encoder.fit_transform(y)
#
```

Figure 5: Encode the target variable into numeric values

Getting the Correlation

This shows the correlation of each of the features to one another. Figure 6 calculates the correlation matrix (**corr**) for the features in the dataset (**data**) excluding the 'label' column. The **drop** method is used to exclude the 'label' column from the dataset.

```
corr= data.drop(['label'],axis=1).corr()
# sns.heatmap(corr,annot=True,cbar=True,cmap='coolwarm')

plt.figure(figsize = (10,10))
sns.heatmap(data.corr(),annot=True)
plt.show()
```

Figure 6: Getting the Correlation

The obtained data frame is then used to compute the correlation matrix using the corr() function seen in Figure 7.

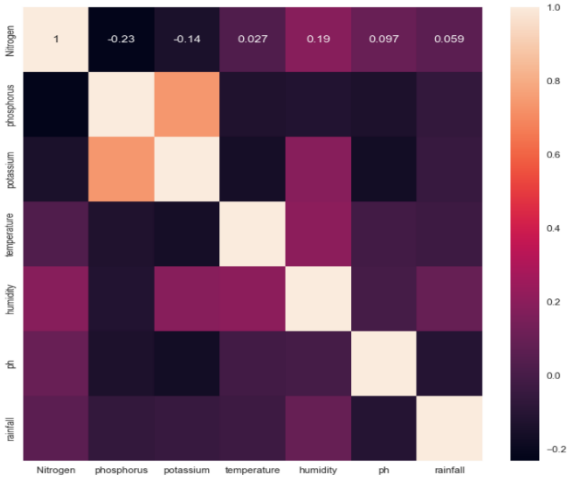

Figure 7: Correlation Matrix
Source: Author

Splitting the Dataset into two

The dataset is divided into input variables (x) and the output variable (y). The symbol (x) represents the characteristics (independent variables) associated with each data point. The variable shown by (y), representing the goal label or dependent variable, is associated with each data point depicted in Figure 8.

```
In [ ]: from collections import Counter
        # Split the data into features (X) and the target label (y)
        X = data[['Nitrogen', 'phosphorus', 'potassium', 'temperature', 'humidity', 'ph', 'rainfall']]
        y = data['label']
```

Figure 8: Splitting the Dataset into two

Train and Test Split

Data partitioning refers to the process of dividing the data into several sets. The training set and the test set. Figure 9 depicts the division of the data into distinct training and testing sets.

```
# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y_encoded, test_size=0.2, random_state=1)
```

Figure 9: Train and Test Split

Feature Scaling

The dataset was standardized using the Standard Scaler method to enhance the performance of the models. Figure 4.10 depicts the process of scaling and standardizing the training, validation, and test data using a standard scaler.

```
In [ ]: from sklearn.preprocessing import StandardScaler
        # scaler
        scaler = StandardScaler()
        X_train = scaler.fit_transform(X_train)
        X_test = scaler.fit_transform(X_test)
```

Figure 10: Feature Scaling

Modeling with the Selected Algorithm

Dataset fitting involves integrating the dataset into the various phases of model building, including training, validation, and testing. Figure 11 depicts the use of several machine learning techniques, such as K-Nearest Neighbor, Decision Tree, Support Vector Machine, Random Forest, Logistic Regression, and Naïve Bayes, in the training and testing of data. Moreover, it demonstrates the suitability of the training and validation data.

```
In [ ]: from sklearn.linear_model import LogisticRegression
        from sklearn.naive_bayes import GaussianNB
        from sklearn.exceptions import ConvergenceWarning
        from sklearn.utils._testing import ignore_warnings

        allModels = {
            'KNN': KNeighborsClassifier(n_neighbors=5),
            'DecisionTree': DecisionTreeClassifier(criterion='gini', random_state=42),
            'SVM': SVC(kernel='linear', C=1, gamma="scale"),
            'RandomForestClassifier': RandomForestClassifier(random_state=42),
            'LogisticRegression': LogisticRegression(solver="newton-cholesky", max_iter=200),
            'NaiveBayes': GaussianNB()
        }
```

Figure 11: Modeling with the Selected Algorithm

Ensemble Model

The prediction model was created by using a crop ensemble approach and leveraging stacking strategies. Stacking is an ensemble learning technique that involves training many diverse machine-learning models and combining their outputs into a meta-model to get final predictions (Kim et al., 2019). Moreover, the use of stacking efficiently mitigates overfitting by enabling the meta-model to assimilate information acquired from the mistakes produced by the underlying models. Figure 12 demonstrates the use of the stacking ensemble methodology.

```
from sklearn.multiclass import OneVsRestClassifier


stack = StackingClassifier(estimators=base_estimators, final_estimator= best_model, cv=
t_start = time()
stack.fit(X_train, y_train)
t_stop = time()
#  predict
p_start = time()
pred = stack.predict(X_test)
p_stop = time()
print(f'Training time: {t_stop - t_start}s')
print(f'Prediction time: {p_stop - p_start}s')
print(f'Score {stack.score(X_test, y_test)}')
```

Figure 12: Implementation of Stacking Ensemble Method

Classification Report of the Ensemble-Based Predictive Model Using Stacking

The classification report for the crop recommendation ensemble-based prediction model is shown in Table 3. The crop selection prediction model, using an ensemble-based (stacking) strategy, demonstrates robust and consistent performance across several criteria, as seen by the classification report. The model exhibits outstanding accuracy, recall, and F1-score metrics for each crop class (1 to 7), suggesting its capacity to provide precise predictions with few occurrences of both false positives and false negatives.

Table 3: Classification Report of the Ensemble-Based Predictive Model

	precision	recall	f1-score	support
1	1.00	1.00	1.00	18
2	1.00	1.00	1.00	22
3	1.00	1.00	1.00	15
4	1.00	1.00	1.00	18
5	0.77	1.00	0.87	17
6	1.00	0.95	0.98	22
7	0.94	1.00	0.97	29
micro avg	0.95	0.99	0.97	141
macro avg	0.96	0.99	0.97	141
weighted avg	0.96	0.99	0.97	141

Classes 1, 2, 3, and 4 exhibit impeccable performance across all criteria, underscoring the model's remarkable accuracy in these particular domains. Class 5 has a little reduced degree of precision, although compensates for it with a notable level of retrieval, suggesting that the model excels in recognizing occurrences of this class, while there could be some erroneous recognitions. Class 6 has a harmonious performance, distinguished by a notable degree of precision and retrieval. The model has outstanding accuracy and recall in predicting instances of Class 7, highlighting its proficiency in this particular category. The F1-score, computed using the micro-average technique, is 0.97, suggesting a substantial degree of overall competence. The weighted average provides further evidence of the model's consistent and high performance across several classes.

Confusion Matric of the Ensemble-Based Predictive Model Using Stacking

The stacking ensemble prediction model for crop selection had exceptional performance in all categories shown in Figure 13. The accuracy, recall, and F1 scores achieved favorable results for classes 1, 2, 3, and 4. Class 5 had a relatively lower level of accuracy, but showed a high level of recall and an outstanding F1-score, suggesting few misclassifications. Class 6 demonstrated exceptional accuracy and achieved a high F1 score, although with a somewhat lower recall rate. Class 7 demonstrated outstanding performance in terms of both accuracy and comprehensiveness, leading to a high F1 score.

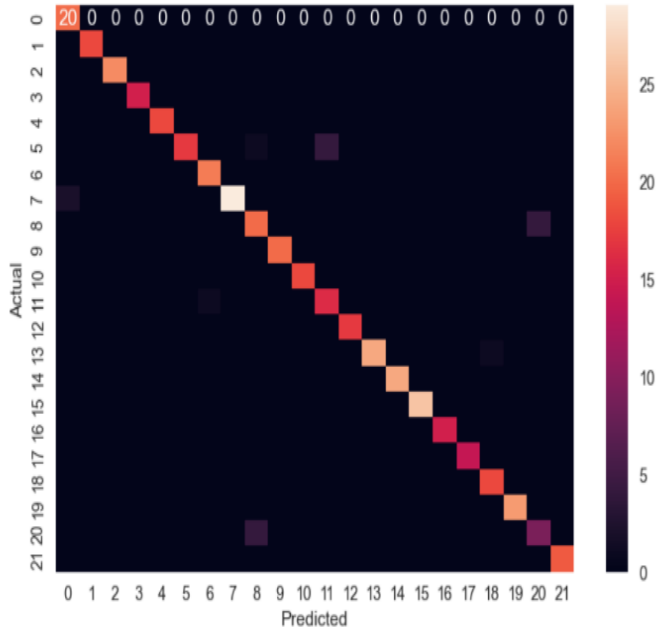

Figure 13: Confusion Matric of the Ensemble-Based Predictive Model Using Stacking

Learning Curve for the Ensemble-Based Predictive Model Using Stacking

The Figure 14 illustrates the Learning Curve for the Ensemble-Based Predictive Model. The x-axis of the learning curve reflects either the amount of training data or the number of training iterations in the training set. The y-axis of the learning curve represents the numerical value of the chosen performance indicator. The training curve illustrates the model's performance on the training data over time, while the validation curve showcases the model's performance on a separate dataset that was not used during training. In this context, the training score remains consistently high at 1.0, suggesting that the model performs well on the training data. The cross-validation score starts high at 1.0 when the training set is around 200 samples, and increases gradually as the size of the training set increases. This implies that the issue of underfitting and overfitting in the model has been resolved.

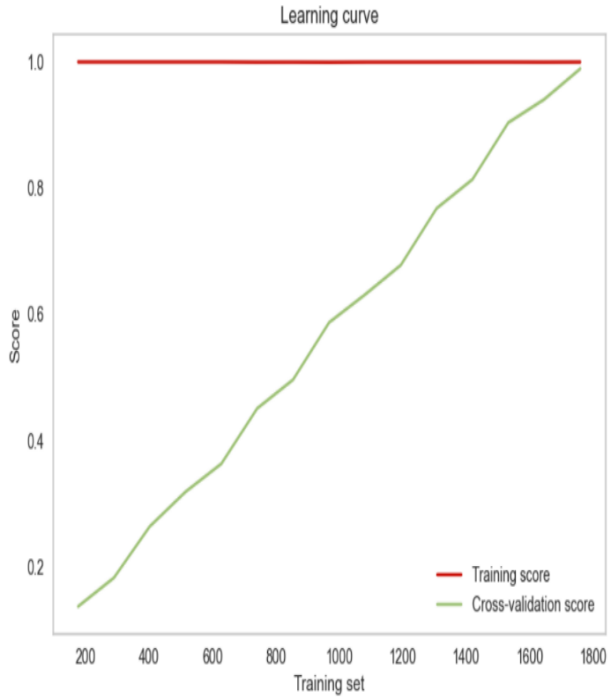

Figure 14: Learning Curve for the Ensemble-Based Predictive Model

ROC Curve for the Ensemble-Based Predictive Model

In Figure 15, the Area Under the Curve (AUC) values of the Receiver Operating Characteristic (ROC) curve for the Ensemble-Based Predictive Model are presented. The dataset consists of twenty-two (22) RUC classes of crops. The model successfully predicted fifteen (15) RUC classes, encompassing Banana, Chickpea, Coconut, Grapes, Kidneypeas, Lentil, Mango, Mothbeans, Mungbean, Muskmelon, Orange, Papaya, Pomegranate, and Watermelon, achieving an impressive AUC value of 1.00. For six other RUC classes, including Apple, Coffee, Cotton, Jute, Maize, and Pigeonbeans, the model achieved AUC values ranging from 0.90 to 0.99. However, for one RUC class, specifically Rice, the AUC value was slightly lower at 0.88.

The findings indicate that the ensemble-based model excelled in predicting all crops, exhibiting AUC values ranging from 0.88 to 1.00. This suggests a remarkable performance across various crop types. Such high AUC values imply strong predictive capability and accuracy of the model in discerning between different crop classes.

Furthermore, the True Positive Rate (TPR) values, ranging from 0.0 to 0.8, underscore the model's ability to correctly identify instances of the positive class across different crop classes. This indicates a high level of sensitivity in detecting the presence of specific crops, further validating the effectiveness of the ensemble-based predictive model.

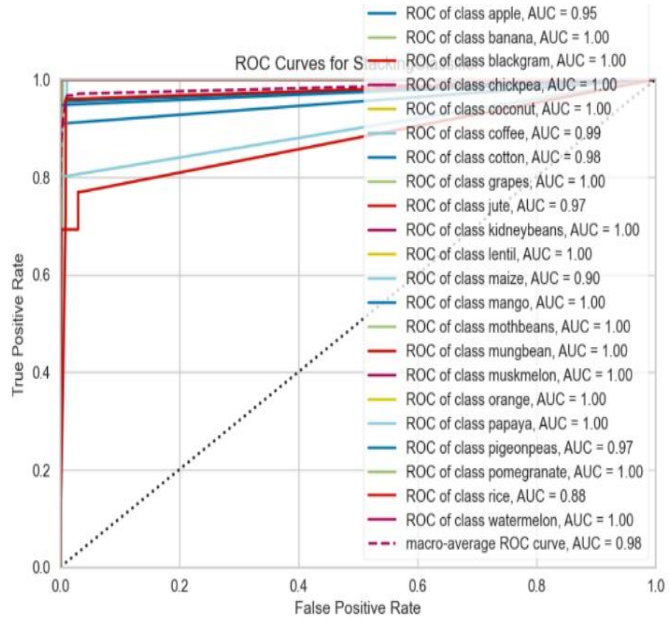

Figure 15: ROC Curve for the Ensemble-Based Predictive Model

DISCUSSION

Performance Evaluation of Selected Models

Table 4 offers a comprehensive evaluation of the performance of all models, facilitating comparison. This study used a total of eight machine-learning models, namely KNN, Decision Tree, Random Forest, Support Vector Machine, Naive Bayes, Logistic Regression, and XGBoost. The models were trained and evaluated as distinct classifiers with the objective of predicting crop outcomes. The results of the models were analyzed and computed to build the stacked ensemble prediction model.

Table 4: Models Performance

S/N	Model	Accuracy	Macro Avg.			
			Precision	Recall	F Score	Support
1.	KNN	98.4	0.98	0.99	0.99	141
2.	DT	99.5	0.96	0.99	0.97	141
3.	SVM	98.9	0.96	0.99	0.99	141
4.	RF	99.8	0.99	0.99	0.99	141
5.	LR	96.1	0.97	0.98	0.98	141
6.	NB	99.6	0.99	0.99	0.99	141
7.	XGBoost	89.1	0.89	0.88	0.88	141
8.	Stacked Model	99.8	0.96	0.99	0.97	141

The Random Forest (RF) model is regarded as the most superior model because to its exceptional accuracy of 99.8%. Furthermore, it demonstrates exceptional accuracy, sensitivity, and overall effectiveness, as all metrics get a score of 0.99, highlighting its strong predictive skills. The Naive Bayes (NB) algorithm has an impressive accuracy rate of 99.6% and regularly demonstrates exceptional precision, recall, and F1-score metrics. The Decision Tree (DT) approach demonstrates exceptional performance in all areas, with an impressive accuracy rate of 99.5%. The accuracies of the Support Vector Machine (SVM) and K-Nearest Neighbors (KNN) are comparable, with SVM obtaining an accuracy of 98.9% and KNN achieving an accuracy of 98.4%. Logistic Regression (LR) has a remarkable accuracy rate of 96.1%. By comparison, XGBoost has an impressive accuracy rate of 87.1%. Nevertheless, it exhibits

deficiencies in terms of precision, recall, and F1-score. To summarize, the findings emphasize the effectiveness of Random Forest and Naive Bayes algorithms in crop selection, highlighting their appropriateness for practical use. Figure 16 displays the data in a bar graph.

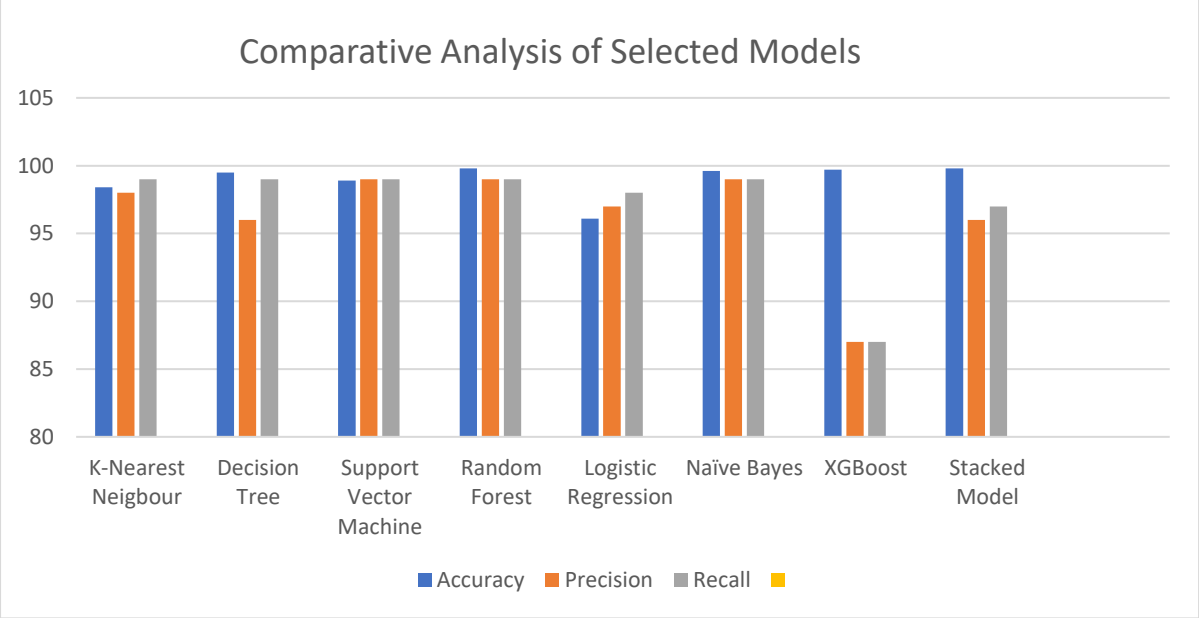

Figure 16: Eight ML Models Accuracy Plot

Comparison with Literature

Table 5 illustrates the efficacy of our suggested research in comparison to previous studies conducted in this domain.

Table 5: Comparison with the Literature

S/N	Author/Year	Machine Learning Model	Accuracy (%)
1.	Khaki & Wang (2019)	Deep Reinforcement Learning	93.5
2.	Palanivel & Surianarayana (2019)	KNN, NB, MLR, ANN and RF	72.33–94.13
3.	Kalimuthu, Vaishnavi & Kishore (2020)	Linear regression, LASSO, Light GBM, Random Forest, and XGBoost	RMSE from 0.07 to 0.2
4.	Nti *et al.* (2023)	AdaBoost GB, Light-GBM, RF, XGBoost, and Stacked TBEL	87.95–99.32
5.	This Study (2024)	KNN, Decision Tree, Random Forest, Support Vector Machine, Naive Bayes, Logistic Regression, XGBoost and Stacked Model	89.1-99.8

The table illustrates the evolution of models over time, with each subsequent study introducing a diverse set of machine learning algorithms to enhance predictive accuracy. Our study, denoted as the "Stacked Model," stands out by achieving an accuracy score approximately 5.4% higher than the most recent research (Nti et al., 2023).

Khaki & Wang (2019) leveraged Deep Reinforcement Learning, obtaining an accuracy of 93.5%. Palanivel & Surianarayana (2019) employed a combination of KNN, NB, MLR, ANN, and RF, yielding accuracy ranging from 72.33% to 94.13%. Kalimuthu, Vaishnavi & Kishore (2020) focused on regression models, reporting RMSE values between 0.07 and 0.2.

Nti et al. (2023) expanded the range of models with AdaBoost GB, Light-GBM, RF, XGBoost, and Stacked TBEL, achieving an accuracy range of 87.95% to 99.32%. In comparison, our study (2024) integrated KNN, Decision Tree, Random Forest, Support Vector Machine, Naive Bayes, Logistic Regression, XGBoost, and a Stacked Model, surpassing previous benchmarks with an accuracy range of 89.1% to 99.8%.

The notable improvement in accuracy by the Stacked Model in our study highlights the effectiveness of combining multiple algorithms for enhanced predictive performance in the given field. This comparative analysis reinforces the significance of our proposed model as a leading approach in the domain.

CONCLUSION

The aim of this study is to develop a prognostic model using an ensemble-based methodology for the purpose of suggesting appropriate crops. In order to achieve this objective, it is crucial to have an extensive dataset that encompasses a diverse array of meteorological and environmental factors, such as temperature, precipitation, humidity, and pH level. Furthermore, it is necessary to include soil properties like as nitrogen, phosphorus, and potassium levels. The dataset underwent data cleaning techniques in Python to rectify missing values, as well as detect and manage anomalous characters and relationships. The technique included using a standard feature scaler to extract relevant attributes while removing unnecessary columns. A prediction model was developed by combining the mathematical equations of ensemble machine learning models, including K-Nearest Neighbors (KNN), Decision Tree, Random Forest, Support Vector Machine (SVM), Naive Bayes, Logistic Regression, and XGBoost. The cleaned dataset was used for the purposes of training, evaluating, and validating the prognostic model. The model achieved a precision level of 99.8%. Furthermore, the model was subjected to a thorough evaluation alongside other pertinent studies, and it shown a higher level of precision in comparison to past inquiries. The model greatly improves its performance by efficiently lowering the occurrences of false positive and false negative discoveries, while increasing the frequency of actual positive outcomes. Implementing this improvement is essential for enhancing the precision of forecasts, reducing mistakes, and bolstering the dependability of the predictive model. This study sets a standard for sophisticated predictive modeling methods, offering useful insights for future investigations in the domain of precision agriculture and crop recommendation systems. Nevertheless, the study recognizes constraints in designing user interfaces that are appropriate for practical usage by farmers.

Conflict of Interest: The corresponding author, representing all authors, confirms the absence of any conflict of interest.

REFERENCES

Aliyev, S., Babayev, F., Galandarova, U., Gafarli, G., & Balajayeva, T. (2023). Economic security of regions: A prerequisite for diversifying the Azerbaijan economy. *Journal of Eastern European and Central Asian Research (JEECAR)*, 10(5), 827-840. https://doi.org/10.15549/jeecar.v10i5.1480

Busetto, L., Wick, W., & Gumbinger, C. (2020). How to use and assess qualitative research methods. *Neurological Research and practice*, 2, 1-10. https://doi.org/10.1186/s42466-020-00059-z

Çakmakçı, R., Salık, M. A., & Çakmakçı, S. (2023). Assessment and Principles of Environmentally Sustainable Food and Agriculture Systems. *Agriculture*, 13(5), 1073.https://doi.org/10.3390/agriculture13051073

Dhanaraju, M., Chenniappan, P., Ramalingam, K., Pazhanivelan, S., & Kaliaperumal, R. (2022). Smart Farming: Internet of Things (IoT)-Based Sustainable Agriculture. *Agriculture* 2022, 12, 1745. https://doi.org/10.3390/agriculture12101745

Durai, S. K. S., & Shamili, M. D. (2022). Smart farming using machine learning and deep learning techniques. *Decision Analytics Journal*, 3, 100041. https://doi.org/10.1016/j.dajour.2022.100041

Enerijiofi, K. E., Musa, S. I., Okolafor, F. I., Igiebor, F. A., Odozi, E. B., & Ikhajiagbe, B. (2023). Sustainable Approaches for the Remediation of Agrochemicals in the Environment. In One Health Implications of Agrochemicals and their Sustainable Alternatives (pp. 511-543). Singapore: Springer Nature Singapore. https://doi.org/10.1007/978-981-99-3439-3_19

Fischer, R. A., & Connor, D. J. (2018). Issues for cropping and agricultural science in the next 20 years. *Field Crops Research,* 222, 121-142. https://doi.org/10.1016/j.fcr.2018.03.008

Food and Agriculture Organization of the United Nations. (2020). FAO at a glance. http://www.fao.org/3/ca9692en/CA9692EN.pdf

Ganaie, M. A., Hu, M., Malik, A. K., Tanveer, M., & Suganthan, P. N. (2022). Ensemble deep learning: A review. Engineering Applications of Artificial Intelligence, 115, 105151. https://doi.org/10.1016/j.engappai.2022.105151

Garg R. (2016). Methodology for research I. *Indian journal of anaesthesia*, 60(9), 640–645. https://doi.org/10.4103/0019-5049.190619

Gopi, P. S. S., & Karthikeyan, M. (2023). Red fox optimization with ensemble recurrent neural network for crop recommendation and yield prediction model. *Multimedia Tools and Applications*, 1-21. http://dx.doi.org/10.1007/s11042-023-16113-2

Guo, Y., Wang, X., Xiao, P., & Xu, X. (2020). An ensemble learning framework for convolutional neural network based on multiple classifiers. *Soft Computing*, 24, 3727-3735. https://doi.org/10.1007/s00500-019-04141-w

Huang, Y., Srivastava, R., Ngo, C., Gao, J., Wu, J., & Chiao, S. (2023). Data-Driven Soil Analysis and Evaluation for Smart Farming Using Machine Learning Approaches. *Agriculture*, 13(9), 1777. https://doi.org/10.3390/agriculture13091777

Kalimuthu, M., Vaishnavi, P., & Kishore, M. (2020, August). Crop prediction using machine learning. In 2020 third international conference on smart systems and inventive technology (ICSSIT) (pp. 926-932). IEEE. https://doi.org/10.1109/ICSSIT48917.2020.9214190

Kephe, P. N., Ayisi, K. K., & Petja, B. M. (2021). Challenges and opportunities in crop simulation modelling under seasonal and projected climate change scenarios for crop production in South Africa. *Agriculture & Food Security*, 10(1), 1-24.https://doi.org/10.1186/s40066-020-00283-5

Khaki, S., & Wang, L. (2019). Crop yield prediction using deep neural networks. *Frontiers in plant science*, 10, 621. https://doi.org/10.3389/fpls.2019.00621

Kim, N., Ha, K. J., Park, N. W., Cho, J., Hong, S., & Lee, Y. W. (2019). A comparison between major artificial intelligence models for crop yield prediction: Case study of the midwestern United States, 2006–2015. ISPRS *International Journal of Geo-Information*, 8(5), 240. https://doi.org/10.3390/ijgi8050240

Mohajan, H. K. (2018). Qualitative research methodology in social sciences and related subjects. *Journal of economic development, environment and people*, 7(1), 23-48. https://www.ceeol.com/search/article-detail?id=640546

Munaweera, T. I. K., Jayawardana, N. U., Rajaratnam, R., & Dissanayake, N. (2022). Modern plant biotechnology as a strategy in addressing climate change and attaining food security. *Agriculture & Food Security*, 11(1), 1-28. https://doi.org/10.1186/s40066-022-00369-2

Nti, I. K., Zaman, A., Nyarko-Boateng, O., Adekoya, A. F., & Keyeremeh, F. (2023). A predictive analytics model for crop suitability and productivity with tree-based ensemble learning. *Decision Analytics Journal*, 8, 100311. https://doi.org/10.1016/j.dajour.2023.100311

Palanivel, K., & Surianarayanan, C. (2019). An approach for prediction of crop yield using machine learning and big data techniques. *International Journal of Computer Engineering and Technology*, 10(3), 110-118. https://ssrn.com/abstract=3555087

Pawlak, K., & Kołodziejczak, M. (2020). The role of agriculture in ensuring food security in developing countries: Considerations in the context of the problem of sustainable food production. *Sustainability*, 12(13), 5488. https://doi.org/10.3390/su12135488

Rashid, M., Bari, B. S., Yusup, Y., Kamaruddin, M. A., & Khan, N. (2021). A comprehensive review of crop yield prediction using machine learning approaches with special emphasis on palm oil yield prediction. IEEE access, 9, 63406-63439. https://doi.org/10.1109/ACCESS.2021.3075159

Reddy, K. M., Kumar, R., & Kiran, S. B. (2023). Impact of Climate Change on Tuber Crops Production and Mitigation Strategies. In Advances in Research on Vegetable Production Under a Changing Climate Vol. 2 (pp. 167-184). Cham: Springer International Publishing. https://doi.org/10.1007/978-3-031-20840-9_8