A FIELD PROJECT REPORT

on

**GreenHarvest: Machine Learning for Sustainable Crop Selection**

**Submitted**

by

221FA04223 221FA04244

K. Sirisha M.Srujana

221FA04438 221FA04510

K. Charitha Sri N. Sai Harshitha

**Under the guidance of**

**Dr.S.Deva Kumar**



**DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING**

**VIGNAN'S FOUNDATION FOR SCIENCE, TECHNOLOGY AND RESEARCH Deemed to be UNIVERSITY**

**Vadlamudi, Guntur.**

**ANDHRA PRADESH, INDIA, PIN-522213.**



**CERTIFICATE**

This is to certify that the Field Project entitled **“**Crop Recommendation using Machine Learning**”** that is being submitted by 221FA04223 (Sirisha), 221FA04244(Srujana), 221FA04438 (Charitha Sri), 221FA04510 (Sai Harshitha)for partial fulfilment of Field Project is a bonafide work carried out under the supervision of Ms. Dr. N. Sameera., Assistant Professor, Department of CSE.

|  |  |  |
| --- | --- | --- |
| Guide name& Signature | Picture 8 | Picture 7  Dr.K.V. Krishna Kishore |
| Assistant/Associate/Professor, CSE | HOD,CSE | Dean, SoCI |



**DECLARATION**

We hereby declare that the Field Project entitled **“Crop Recommendation using Machine Learning”** is being submitted by 221FA04223 (Sirisha), 221FA04244(Srujana), 221FA04438 (Charitha Sri), 221FA04510 (Sai Harshitha) in partial fulfilment of Field Project course work. This is our original work, and this project has not formed the basis for the award of any degree. We have worked under the supervision of Ms. Dr. N. Sameera., Assistant Professor, Department of CSE

By

**221FA04223 (Sirisha)**

**221FA04244(Srujana)**

**221FA04438 (Charitha Sri)**

**221FA04510 (Sai Harshitha)**

Date:

**ABSTRACT**

This project focuses on building a machine learning-based crop recommendation system to help farmers choose the best crops based on soil and environmental factors such as nitrogen, phosphorus, potassium content, temperature, humidity, pH, and rainfall. The aim is to support precision agriculture by improving crop yields and resource management through data-driven insights. The dataset was preprocessed by handling missing values, scaling the features, and using label encoding to convert categorical crop labels into numeric data for model input. Several machine learning models were implemented, including Logistic Regression, Naive Bayes, K-Nearest Neighbors (KNN), Decision Tree, Support Vector Machine (SVM), and Gradient Boosting. Each model was trained and compared based on accuracy. The Naive Bayes model performed the best with an accuracy of 99.55%, followed by Gradient Boosting and Decision Tree at 98.18%, KNN at 97.05%, SVM at 96.14%, and Logistic Regression at 94.55%. The results show that models like Naive Bayes and Decision Tree are highly effective in recommending suitable crops, making this system a valuable tool for enhancing agricultural productivity.

**Keywords:**

Crop Recommendation, Machine Learning, Naive Bayes, KNN, Decision Tree, Gradient Boosting, SVM, Precision Agriculture, Feature Scaling, Classification Models.

**TABLE OF CONTENTS**

1. [Introduction 1](#_bookmark0)
   1. [Background and Importance of Crop Recommendation 2](#_bookmark1)
   2. [Overview of Machine Learning in Agriculture 3](#_bookmark2)
   3. [Research Objectives and Scope 4](#_bookmark3)
   4. [Challenges in Crop Recommendation 5](#_bookmark4)
   5. Applications of ML in Precision Agriculture 6
2. [Literature Survey 7](#_bookmark5)
   1. [Previous Studies on Crop Recommendation 8](#_bookmark6)
   2. [Integration of Environmental and Soil Data 9](#_bookmark7)
3. [Proposed System 10](#_bookmark8)
   1. [Input dataset 11](#_bookmark9)
      1. Detailed features of dataset 12
   2. [Data Pre-processing 13](#_bookmark10)
      1. [Handling missing values 14](#_bookmark11)
      2. [Data encoding using label encoding 15](#_bookmark13)
   3. [Model Building 16](#_bookmark14)
   4. [Methodology of the system 17](#_bookmark16)
   5. [Model Evaluation 18](#_bookmark17)
   6. [Constraints 24](#_bookmark18)
   7. Ethical Considerations 25
4. [Implementation 26](#_bookmark19)
   1. Environment Setup 27
   2. Sample code for preprocessing and MLP operations 28
5. Experimentation and Result Analysis 32
6. Conclusion 34
7. References 37

**LIST OF FIGURES**

Figure 1. Architecture of the Proposed Crop Recommendation System

Figure 2. Key Features in the Dataset

Figure 3. Data Pre-processing Overview

Figure 4. Handling Missing Values

Figure 5. Model Building Process

Figure 6. Accuracy Comparison of Different Models

Figure 7. Confusion Matrix for Best Performing Model

Figure 8. Feature Importance Visualization

Figure 9. Model Performance Metrics Overview

**LIST OF TABLES**

Table 1. Model Accuracy Comparison

Table 2. Confusion matrix for Best Performing Model

**CHAPTER-1 INTRODUCTION**

1. **INTRODUCTION**

**1.1 Background and Importance of Crop Recommendation**

Crop recommendation plays a crucial role in modern agriculture, ensuring that farmers plant the crops best suited to their local environmental and soil conditions. By aligning crop choices with real-time data such as soil, pH, temperature, and rainfall, agricultural productivity can be maximized, and resource use (like water and fertilizers) can be optimized. This approach helps in enhancing yields and promoting sustainable farming practices, especially in areas affected by climate change or with limited agricultural inputs.

**Significance of Crop Recommendation**

Optimized Crop Selection: By suggesting suitable crops based on soil conditions, climate, and market demand, crop recommendation systems help farmers choose crops that are likely to thrive in their specific environment, leading to higher yields.

Risk Mitigation: Crop recommendation systems can help farmers identify potential risks such as pests, diseases, or adverse weather conditions, allowing them to take preventive measures and reduce losses.

Biodiversity: By encouraging crop diversification, these systems can contribute to maintaining biodiversity in agricultural landscapes.

Market Access: Crop recommendation systems can help farmers identify profitable crops that are in demand in the market, ensuring better economic returns.

Enhanced Production: By optimizing crop selection and management, crop recommendation systems can contribute to increased agricultural production, ensuring food security for local communities and regions.

**1.2 Overview of Machine Learning in Agriculture**

Machine Learning (ML) has emerged as a transformative technology in agriculture, enabling the analysis of large datasets to derive insights into farming practices, crop selection, and yield optimization. From predictive models for disease detection to recommendation systems for crop selection, ML integrates environmental data, soil analysis, and crop performance metrics to deliver actionable insights that drive decision-making at both the local farm level and in large-scale agribusiness operations.

**Machine Learning Applications in Crop Recommendation**

**Soil and Climate Data Analysis:**

ML models can analyze soil parameters (such as pH levels, moisture, and nutrient content) alongside climatic factors (such as temperature, rainfall, and humidity) to recommend crops that are most likely to thrive in given conditions. By using historical data and real-time environmental data, the models provide precise recommendations for different regions and seasons, maximizing yields.

**Yield Prediction and Crop Suitability:**

By leveraging past yield data, machine learning models can predict how well certain crops will perform under similar environmental conditions. These models use data on previous crop yields, weather patterns, and soil characteristics to forecast yields and recommend crops with the highest potential for success in a specific area. This helps farmers choose crops that are more suited to local conditions, improving efficiency.

**Pest and Disease Detection:**

Some advanced crop recommendation systems integrate pest and disease prediction using image recognition and classification techniques. These systems analyze plant health and predict possible threats from diseases or pests, and based on the risk levels, recommend crops that are less susceptible to these problems. This enables farmers to avoid crops that may face heavy losses due to infestations.

**Resource Optimization:**

ML models can optimize the use of resources like water, fertilizers, and pesticides by recommending crops that are most compatible with the available resources. For instance, in areas with limited water supply, ML systems can suggest drought-resistant crops, reducing the need for irrigation. Similarly, they can recommend crops that require fewer fertilizers or are resistant to local pests.

**Real-Time Monitoring and Adaptive Recommendations:**

ML models can continuously monitor environmental data in real-time and adjust crop recommendations as conditions change. For instance, if there is an unexpected change in rainfall patterns or temperature, the system can suggest alternative crops or modifications in planting schedules. This adaptive approach helps farmers make dynamic, data-driven decisions throughout the growing season.

**1.3 Research Objectives and Scope**

Research on machine learning in crop recommendation typically aims to achieve several key goals:

**Maximizing Crop Yield:** By analyzing environmental factors such as soil type, pH levels, and weather patterns, ML models can recommend the most suitable crops for specific regions or seasons, thereby increasing yield potential and overall farm productivity.

**Optimizing Resource Utilization:** Efficient use of water, fertilizers, and pesticides is crucial in farming. Research aims to develop ML models that can recommend crops based on the availability of resources

**Enhancing Farmer Decision-Making:** Machine learning research seeks to empower farmers by providing them with data-driven, actionable insights for decision-making. These insights include recommending the best crops for a given plot of land, suggesting optimal planting and harvesting schedules, and providing alerts about potential risks like pest infestations or disease outbreaks.

**Adapting to Climate Change:** Research in ML for crop recommendation focuses on identifying climate-resilient crops that can thrive under changing conditions, such as erratic rainfall, rising temperatures, or shifting growing seasons. The goal is to provide farmers with adaptive strategies that mitigate the risks of climate variability.

**Reducing Crop Failures:** By integrating ML with weather forecasting, soil analysis, and past crop data, researchers aim to develop systems that can predict and prevent crop failures. This goal involves using predictive analytics to detect early signs of issues such as drought, pest attacks, or nutrient deficiencies, allowing for timely intervention.

Enhanced Crop Selection: Design machine learning models that consider soil data (nutrient levels, pH), climate data (temperature, rainfall), and market information (demand, pricing) to recommend the most suitable crops for specific agricultural regions.

**Resource Optimization:** Develop models that recommend irrigation schedules, fertilizer application rates, and pest control strategies based on crop type and environmental conditions, minimizing waste and maximizing productivity.

**Sustainable Practices**: Explore the integration of CRS with soil health monitoring and weather forecasting to promote sustainable agriculture. Recommend cover crops, crop rotations, and tillage practices that optimize soil health and minimize environmental impact.

**Research Scope:**

**Machine Learning Techniques**:

Investigate the use of supervised learning algorithms (e.g., Random Forests, Support Vector Machines) and unsupervised learning approaches (e.g., K-Means clustering) to identify patterns and relationships between crop performance, environmental factors, and market trends.

**Model Explainability and User Interface:**

Develop interpretable models that farmers can understand and utilize effectively. Design user-friendly interfaces for CRS that are accessible and cater to the needs of farmers with varying levels of technical expertise.

**Economic and Social Impact:**

Assess the economic benefits of CRS for farmers, including increased profitability and improved livelihoods. Explore the potential social impact of CRS on rural communities through enhanced food security and agricultural sustainability.

**Challenges and Limitations:**

Identify challenges like data quality, access to technology, and farmer adoption. Develop strategies to address these limitations and ensure the widespread adoption of CRS by farmers.

**Evaluation and Validation:**

Design robust evaluation metrics (e.g., yield improvement, resource efficiency) to assess the effectiveness of CRS in real-world agricultural settings. Implement field trials and farmer-participatory research to validate CRS recommendations.

**Integration with Farm Management Systems:**

Explore the integration of CRS with existing farm management software and agricultural extension services, offering farmers a comprehensive platform to manage their operations effectively.

* 1. **Challenges in Crop Recommendation**

**1.Data Quality and Availability:** Ensuring access to reliable and comprehensive data on soil properties, climate, market prices, and pest and disease outbreaks is crucial but often challenging.

**2.Model Complexity:** Developing accurate and interpretable models that consider the complex interactions between various factors (soil, climate, pests, diseases, etc.) is a complex task.

**3.Dynamic Environments:** Agricultural environments are dynamic and constantly changing. Crop recommendation systems need to adapt to variations in weather patterns, soil conditions, and market demands.

**4.Uncertainty and Risk:** Crop production is inherently risky due to factors like pests, diseases, and unpredictable weather. Crop recommendation systems must consider these uncertainties and provide risk mitigation strategies.

**5.Farmer Adoption:** Convincing farmers to adopt new technologies and trust the recommendations provided by these systems can be a challenge.

**6.Computational Resources:** Implementing and running complex crop recommendation models can require significant computational resources, which might be a barrier for resource-limited farmers.

**7.Data Privacy and Security:** Protecting sensitive farmer data, such as crop yields, financial information, and location data, is crucial to ensure trust and adoption.

**8.Contextualization:** Crop recommendations need to be tailored to specific farming practices, cultural preferences, and local regulations.

**9.Scalability:** The system should be scalable to accommodate a large number of farmers and diverse agricultural landscapes.

**10.Continuous Improvement:** Crop recommendation systems need to be continuously updated and improved based on feedback from farmers and evolving agricultural conditions.

**11.Economic Factors:** Crop recommendation systems need to consider economic factors such as market prices, input costs, and government policies to provide practical and profitable recommendations.

**12.Ethical Considerations:** Ensuring that crop recommendation systems are fair, equitable, and do not exacerbate existing inequalities in agriculture is crucial. This includes addressing issues like access to technology, data privacy, and the potential for unintended consequences.

* 1. **Applications of ML in Precision Agriculture**

Machine learning (ML) is revolutionizing agriculture by enabling precision farming practices. By analyzing vast amounts of data, ML algorithms can optimize resource allocation, improve crop yields, and enhance sustainability. Some key applications include:

Yield Prediction: ML models can accurately predict crop yields based on various factors like weather, soil conditions, and crop management practices.

Disease and Pest Detection: Early detection of diseases and pests through ML-powered image analysis helps farmers take timely action to prevent crop damage.

Soil Health Monitoring: ML algorithms can analyze soil data to assess its health and recommend appropriate nutrient management strategies.

Irrigation Optimization: ML-enabled systems can optimize irrigation schedules based on real-time data on soil moisture and weather conditions.

Weed Control: ML-powered image analysis can identify and locate weeds, enabling targeted herbicide application.

Harvest Optimization: ML models can predict the optimal time for harvesting based on crop maturity and market conditions.

Inventory Management: ML algorithms can forecast crop demand and help farmers manage inventory levels effectively.

**Important Uses of Machine Learning in the Identification of Crop Recommendation**

Machine learning plays a crucial role in crop recommendation by leveraging large datasets to predict optimal crops for specific regions based on a variety of factors. Here are some important uses of machine learning in crop recommendation:

**1.Soil Quality Assessment**:

Machine learning models can analyze soil characteristics such as pH, moisture content, nutrient levels, and texture. By processing this data, the models recommend crops that are most suited for the given soil type, ensuring better yields and efficient land use.

**2.Weather and Climate Analysis**:

Machine learning can integrate weather data such as temperature, rainfall patterns, and humidity levels to predict which crops will thrive in specific climatic conditions. This helps farmers make informed decisions about crop selection based on weather forecasts and historical data.

**3.Disease and Pest Prediction**:

Predictive models can detect patterns in crop diseases or pest infestations. By analyzing historical data and environmental conditions, these models can recommend crops that are resistant to certain diseases or pests in specific areas.

**4.Water Resource Management**:

Machine learning can predict the water requirements for various crops and recommend water-efficient crops for regions with limited water availability. This ensures sustainable farming practices, especially in drought-prone areas.

**5.Market Trends and Demand Forecasting**: By analyzing market trends, pricing, and demand, machine learning can suggest crops that have higher chances of profitability. This helps farmers choose crops based on market dynamics, reducing the risk of surplus or low-demand crops.

**6.Fertilizer and Input Optimization**: Machine learning can recommend the right type and amount of fertilizer or other inputs for specific crops and soil conditions. This improves crop yields and minimizes resource wastage by suggesting tailored fertilizer usage.

**7.Precision Agriculture**: Through satellite imagery, drones, and IoT sensors, machine learning models can assess field conditions in real-time. These models then suggest the best crops for each section of the field, enabling precision farming that maximizes productivity and resource efficiency.

**8.Sustainability and Environmental Impact**: Machine learning can help in choosing crops that not only optimize yield but also reduce environmental impact. For instance, models can recommend low-carbon or sustainable crops based on region-specific environmental factors.

**9.Yield Prediction**: Based on historical data, weather patterns, soil health, and other factors, machine learning models can predict crop yields. This helps in crop planning and choosing crops that are likely to produce the best yields under current conditions.

**CHAPTER-2 LITERATURE SURVEY**

**2. LITERATURE SURVEY**

**2.1** **Previous Studies on Crop Recommendation**

Shivnath and Santanu devised a machine learning approach to examine soil fertility and plant nutrient management. The backpropagation network (BPN) used is trained with inputs on crop growth characteristics[18]. More and more researchers have begun to identify this problem in Indian agriculture and are increasingly dedicating their time and efforts to help alleviate the issue[16].

The authors considered few inevitable parameters including demographic features, soil characteristics, environmental parameters, and season-based features to predict the type of crop that can be sown[5]. The algorithms used by the authors were Random Forest, Naive Bayes, CHAID model, Kclosest neighbor, etc. This article [4] talks about the various applications of ANN, ML and IoT in agriculture and various models of assistance in precision farming[8]. The system recognises a user’s location before calculating similarity across upazilas utilising different agro-ecological and agro-climatic data at the upazila level using the Pearson co-relation similarity technique[13].

This paper integrates the work of several authors in a single place so it is valuable for specialists to get data of current situation of data mining systems and applications in context to farming field[19]. This dataset was created by supplementing existing rainfall, climate, and fertilizer datasets for India. ICFA( Indian Chamber of Food and Agriculture), India gathered these records[2]. The proposed system uses a Naive Bayes Classifier to recommend the crop. Parameters such as temperature, humidity, location are passed to the model to predict a suitable crop for the farm[8]. The major factors which are essential for plant growth, are processed together using various algorithms whereas the other models consider only parameters at once keeping the other factors constant[20].

Agriculture, which contributes only 15% to India's GDP, faces challenges such as substandard productivity and lack of strategic planning, leading to a high rate of suicide among marginal farmers[14]. Random forest (RF) method was used to predict crop yields in the agricultural sector. The RF method provides the [6]optimal crop production model by considering the fewest number of models possible.

Both utilized machine learning algorithms, such as Random Forest, Decision Tree, and Logistic Regression, for accurate crop predictions and real-time field insights. However, it lacked specific details on performance metrics and the overall procedure[10]. Machine learning calculations have improved the exactness of artificial intelligence machines including sensor based frameworks utilized in accuracy farming[19]. The performance of the algorithms were assessed based on their prediction of nitrogen fertilizer recommendation. Along with the adaptability of the crops to the environment, each crop species require specific soil — site conditions for optimum growth[4].

Based on the literature survey the shortcoming that we observed in these notable publications is that the authors have considered lesser [12]experimental parameters for developing the recommendation model. In our work we have considered suitability of soil properties, climatic properties.

Several machine learning methods are applied separately to recognize the soilclass of test sample to select the crop[17]. Even after suggesting the best crop type,we can also do for the system to track the plant growth and we can make the system to provides feedback if the farm is malnourished. So that the user can take necessary precautions prior[20]

**2.2 Integration of Environmental and Soil Data**

1. Environmental Factors:

Soil Properties: The soil's nutrient levels, such as nitrogen, phosphorus, and potassium (N, P, K), are important for determining suitable crops.

Soil pH: Different crops thrive in different pH ranges. For example, acidic, neutral, or alkaline soil conditions impact crop selection.

Moisture Content: Soil moisture and water availability are key indicators for determining the suitability of crops, especially in rainfed regions.

2. Climatic Conditions:

Temperature: Different crops have specific temperature ranges for optimal growth. For instance, some crops need a warm climate, while others prefer cooler conditions.

Rainfall: The amount of rainfall and its distribution over time is crucial for crops that rely on rain for irrigation.

Humidity: Humidity levels can affect crop yields, especially in areas prone to extreme dryness or wetness.

3. Agricultural Practices:

Fertilizer Use: Fertilizer recommendations based on soil nutrient content can help farmers select crops that would benefit the most from the applied nutrients.

Irrigation Practices: Crops that require more water might be recommended where irrigation systems are well-developed.

4. Machine Learning and Data-Driven Approaches:

Supervised Learning Algorithms: Models like Naive Bayes, Logistic Regression, Decision Trees, and Gradient Boosting are often used to classify and recommend crops based on features like soil health and climate conditions.

Feature Selection: Studies often emphasize selecting the most relevant features (environmental, soil, and climate data) to improve model accuracy and ensure reliable recommendations.

5. Data Used in Crop Recommendation:

Historical Crop Data: Analysis of previous crop yields, soil health reports, and climate conditions helps refine recommendations.

Sensor Data: In modern precision agriculture, real-time data from soil sensors, weather stations, and satellite imagery can be integrated to optimize crop choices.

**CHAPTER-3**

**PROPOSED SYSTEM**

1. **PROPOSED SYSTEM**

1. Dataset and Preprocessing

The project utilizes a dataset named "Crop\_recommendation.csv" containing various features potentially influencing crop selection. The code snippet explores the dataset:

Loads the data using pandas.

Prints the first few rows to understand the data structure.

Provides information about each feature using a predefined list.

Checks for missing values and replaces them with the mean for numerical features.

Uses LabelEncoder to transform categorical labels in the "label" column (representing recommended crops) into numerical values.

2. Exploratory Data Analysis (EDA)

Analyzes the distribution of each feature using histograms.

Summarizes the unique values and counts for each feature.

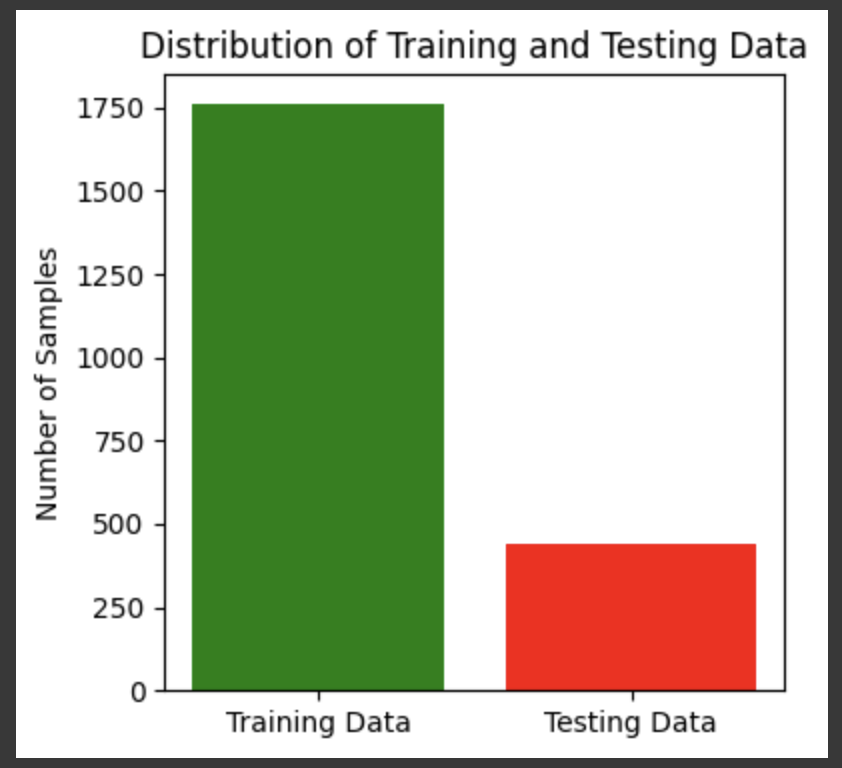
Checks for class imbalance in the target variable ("label"), which could affect model performance.

3. Data Splitting

Splits the data into training and testing sets using train\_test\_split with a 20% test size and a random state for reproducibility.

Prints the shapes of both training and testing sets to confirm proper split.

Visualizes the distribution of data points in each set.



4. Feature Selection

ANOVA (Analysis of Variance): Selects the top 5 features based on their F-value, indicating their influence on the target variable.

ANNOVA selected features:Index([‘N’, ‘P’, ‘K’, ‘humidity’ , ‘rainfall’].dtype= ‘object’)

5. Model Development and Training

Implements a loop iterating through various machine learning models:

Logistic Regression

Naive Bayes

K-Nearest Neighbors (KNN)

Decision Tree

Support Vector Machine (SVM)

Gradient Boosting

Trains each model on the training data.

6. Model Evaluation

Evaluates each model on the testing data using the following metrics:

Accuracy: Proportion of correctly predicted crop recommendations.

Precision: Ratio of true positives to all predicted positives (avoiding false positives).

Recall: Ratio of true positives to all actual positives (avoiding false negatives).

F1-score: Harmonic mean of precision and recall, combining both measures.

Prints the performance metrics for each model.

Generates confusion matrices as heatmaps using Seaborn, visualizing the distribution of correct and incorrect predictions for each class.

7. Model Selection and Testing

Analyzes the evaluation results to identify the model with the best overall performance, considering factors like accuracy, precision, recall, and F1-score.

Selects the best performing model.

Evaluates the chosen model on a new unseen dataset (if available) to assess its generalizability.

8. Deployment and Continuous Improvement

Deploys the final model as a web application, API, or mobile app, allowing farmers to access crop recommendations based on their land conditions.

Monitors the model's performance in real-world scenarios and collects feedback from farmers.

Refines the model based on new data and feedback to maintain accuracy and improve its effectiveness over time.

9. Ethical Considerations

Ensures fairness and transparency in crop recommendations, avoiding bias towards specific crops or favoring large-scale agriculture over small-scale farming.

Protects user privacy by implementing appropriate data security measures.

Promotes responsible use of the system, avoiding over-reliance on technology and encouraging farmers to maintain good agricultural practices.

**3.1 input Dataset**

For a crop recommendation system, the dataset typically contains 8 features that capture various factors influencing crop growth and yield. Below are the features and their descriptions:

**3.1.1 Detailed Features of the Dataset**

N (Nitrogen Content): Amount of nitrogen present in the soil.

P (Phosphorus Content): Amount of phosphorus in the soil.

K (Potassium Content): Potassium content in the soil.

Temperature (°C): The average temperature of the region.

Humidity (%): The amount of moisture in the air, which affects transpiration and crop water needs.

pH Level: pH of the soil, indicating its acidity or alkalinity.

Rainfall (mm): Amount of rainfall received, affecting water availability for crops.

Crop Type (Target Variable): The type of crop recommended based on the given input features, such as wheat, rice, maize, etc.

* 1. **Data Pre-processing**

Pre-processing the data is a crucial step in ensuring that the dataset is ready for building a crop recommendation model. It involves cleaning the data, handling missing values, and transforming categorical data into numerical formats, among other steps.

**Handling Missing Values**:  
Missing values in features like soil nutrients (N, P, K) were filled using mean imputation to maintain data integrity. If a feature had no missing values, no imputation was applied.

**Dropping Unnecessary Columns**:  
No columns were dropped, as all features contribute significantly to crop recommendation.

**Encoding Categorical Features**:  
Label Encoder was used to convert categorical variable like crop type into numerical form, ensuring compatibility with machine learning models.

**Feature Scaling**:  
Min-Max scaling was applied to continuous features such as temperature, rainfall, and pH, bringing them into the range [0,1] to ensure uniformity across the dataset and improve model performance.

**Feature Selection**:  
Correlation analysis was used to identify redundant or highly correlated features, while Recursive Feature Elimination (RFE) was tested to ensure that the most relevant features were selected.

**Data Splitting**:  
The dataset was split into 80% training and 20% testing sets, ensuring model evaluation on unseen data.

**Outlier Detection and Handling**:  
Outliers in continuous features such as temperature were detected using z-scores and retained if deemed agronomically significant.

**3.3 Model Building**

The model-building process in the project involves several key functions. First, train\_test\_split() is used to divide the dataset into training and testing sets. StandardScaler () is applied to normalize the data for consistent model performance.

Preparing Data

The dataset is divided into two parts: features (X) and the target variable (y).

X: Represents various factors influencing crop growth, such as soil characteristics (pH, nitrogen content, phosphorus levels, potassium levels), weather conditions (temperature, humidity, rainfall).

y: Represents the recommended crop based on the feature inputs.

Data Division

The dataset was split into a training set (80%) and a testing set (20%).

This division allows the model to learn from the training data while providing an unbiased evaluation of its performance on unseen data.

Training of Models

Multiple models were trained and evaluated, including:

Logistic Regression:

A simple and interpretable model that estimates the probability of a particular crop being suitable for the given environmental conditions and soil characteristics.

The logistic regression model is trained to predict the probability of recommending a specific crop based on factors like soil pH, rainfall, temperature, and nutrient content.

Naive Bayes:

The Gaussian Naive Bayes model was employed due to its effectiveness with independent features.

This method is effective when the feature independence assumption holds, allowing the model to calculate the probability of each crop being suitable given the current conditions.

K-Nearest Neighbors (KNN):

The KNN classifier is trained to recommend the most suitable crop based on the proximity of feature values in the training dataset.

This model uses distance metrics to classify new instances based on their similarity to historical data on successful crop growth.

Support Vector Machine (SVM):

The SVM model can be trained to find the optimal hyperplane that separates different crops based on environmental and soil factors.

This model is particularly useful when the data is high-dimensional and works well if the crop recommendations can be linearly separable in the feature space.

Decision Tree:

A decision tree classifier can be built to recommend crops by recursively partitioning the data based on input features.

This model is highly interpretable, making it easy to visualize the decision paths and understand why a particular crop is being recommended.

Gradient Boosting:

The Gradient Boosting model was applied to improve classification accuracy by combining weak learners (decision trees) in an iterative boosting process. This model captures complex patterns in the data for better crop recommendations.

Performance Metrics:

Accuracy: Measures the overall correctness of the crop recommendations.

Precision: Indicates the proportion of correctly recommended crops out of all crops predicted as suitable.

Recall: Represents how effectively the model identified all appropriate crops for a given set of environmental conditions.

F1-Score: A balance between precision and recall, particularly valuable for datasets with class imbalance.

Confusion Matrix: The confusion matrix highlighted areas where the model might struggle, such as recommending crops for borderline or ambiguous growing conditions.

Evaluation Insights: Different models may perform variably depending on the complexity of the data

The evaluation showed that some models excelled in balancing precision and recall, while others faced challenges in misclassifying crops under specific conditions.

**3.4 Methodology of the system**

**A. Architecture of the System**

The proposed system architecture for predicting suitable crops based on soil and climatic conditions consists of several key steps: data collection, preprocessing, feature selection, model training, and classification. The architecture includes the following components:

**Input Layer:** The input layer gathers essential environmental and soil-related data, such as soil nutrients (N, P, K), temperature, humidity, pH, and rainfall. This information is crucial for recommending appropriate crops based on the growing conditions.

**Preprocessing Layer:** The input data undergoes preprocessing to ensure it is clean and formatted for machine learning models. This process involves handling missing values, encoding categorical variables if necessary, and scaling numerical features such as N, P, K, temperature, and other environmental factors.

**Feature Selection Layer:** Relevant features are selected and retained based on their importance in the crop recommendation task. This step eliminates less significant features and keeps only those that are directly related to the outcome, such as soil and weather conditions.

ANOVA (Analysis of Variance): Selects the top 5 features based on their F-value, indicating their influence on the target variable.

**Classifier Layer**: Several machine learning algorithms are employed to classify the data and recommend suitable crops. The models implemented in the system include:

Logistic Regression

Naive Bayes

K-Nearest Neighbors (KNN)

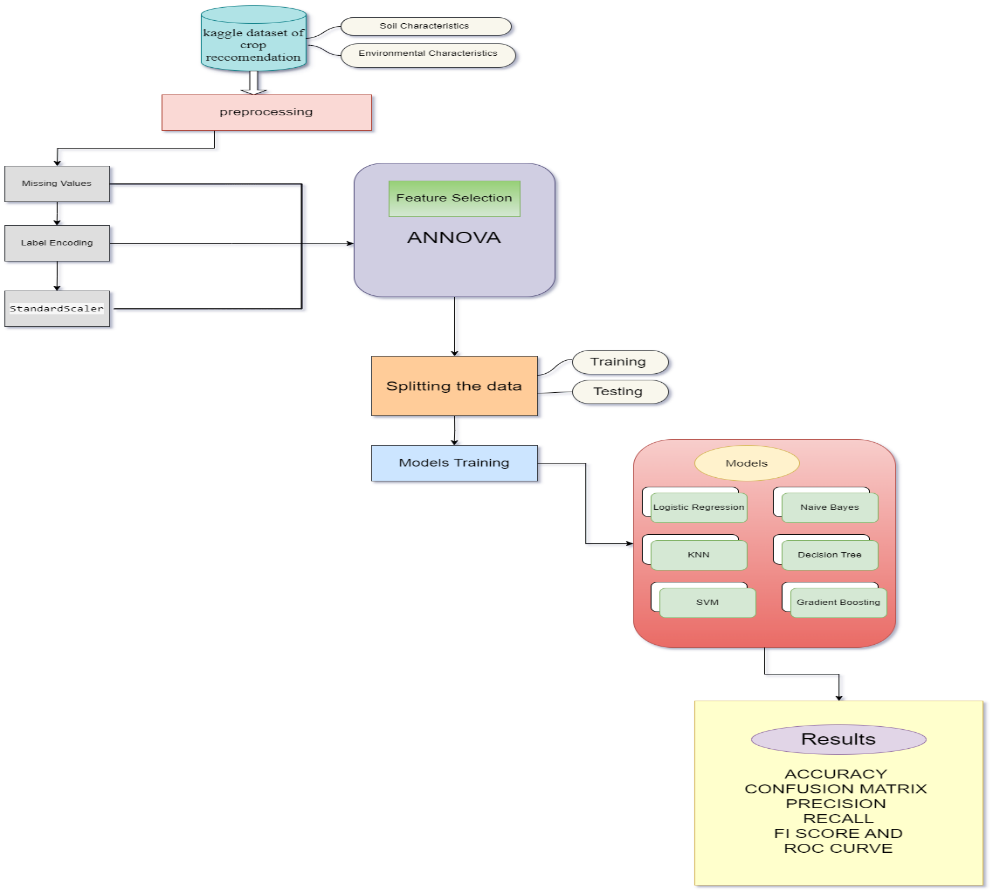
Support Vector Machine (SVM)

Decision Tree

Gradient Boosting

Each model is trained using the selected features, and their performance is evaluated to determine the best one for crop recommendation.

**Output Layer:** The system presents the final recommendation, indicating the most suitable crop based on the input data and model predictions.

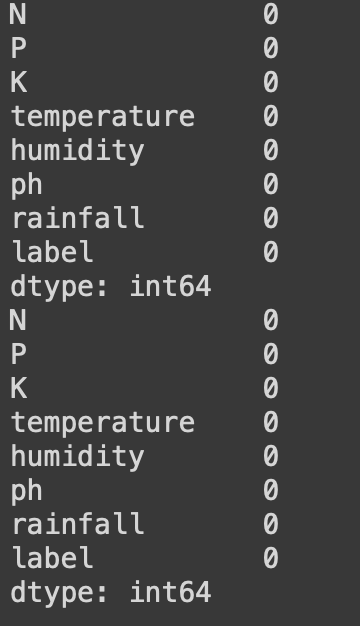


**B. Training and Preprocessing of Data**

Data preprocessing is an essential step to ensure the dataset is clean and structured for effective model training. In this project, the following preprocessing techniques were employed:

**Data Cleaning:** The dataset was reviewed for any missing values or inconsistent entries. Columns that were irrelevant or redundant were removed to focus on key features that contribute to accurate crop recommendations.

**Handling Missing Data**: Missing values in the dataset were addressed by using imputation techniques such as mean or median imputation for numerical fields (e.g., nitrogen, phosphorus, potassium, etc.). There are no missing values so the features are same.



**Label Encoding:** For the categorical target variable (the crop to be recommended), label encoding was applied to convert crop names into numerical labels that can be processed by the machine learning algorithms.

**Feature Scaling:** Standardization or normalization techniques were applied to ensure uniform scaling of the numerical features such as nitrogen, phosphorus, and potassium levels, temperature, and rainfall. This ensures that no single feature dominates the model training process due to larger magnitudes.

**Data Splitting:** The dataset was divided into training (80%) and testing (20%) sets to evaluate model performance. This split helps ensure the model is tested on unseen data, providing a reliable assessment of its generalization ability.

**C. Feature Selection**

Feature selection plays a key role in improving the model’s performance by identifying the most relevant features for crop recommendation. In this project, embedded feature selection methods such as those provided by Decision Trees and Logistic Regression (L1 regularization) were used to automatically select important features during the training process. Additionally, filter methods, such as correlation-based feature selection, were employed to measure the strength of relationships between input features and the target variable (crop recommendation).

ANOVA (Analysis of Variance): Selects the top 5 features based on their F-value, indicating their influence on the target variable.

The following features were retained as key predictors for recommending suitable crops:

N (Nitrogen content in the soil)

P (Phosphorus content in the soil)

K (Potassium content in the soil)

Humidity (Percentage)

Rainfall (mm of rainfall)

These features were found to have the most significant impact on determining the appropriate crop for a given set of environmental conditions. By focusing on these features, the system improves its ability to make accurate crop recommendations.

**D. Model Training**

Several machine learning models were trained to predict the most suitable crop for cultivation based on the given environmental and soil data. The following models were used:

**Logistic Regression:** A simple and interpretable model that estimates the probability of each crop being suitable based on the input features.

**Naive Bayes:** A probabilistic classifier that computes the likelihood of each crop based on the assumption of conditional independence between features. This model is particularly useful for handling both categorical and continuous data.

**K-Nearest Neighbors (KNN):** This model classifies crops by looking at the nearest training data points in the feature space, effectively recommending a crop based on similar historical data points.

**Support Vector Machine (SVM):** A powerful model that finds the optimal hyperplane for separating crops into distinct classes based on input features. SVM works well in high-dimensional spaces.

**Decision Tree:** This model was employed for its interpretability and its ability to split the data into distinct groups based on the most significant features. Decision Trees are also useful for ranking features by importance.

**Gradient Boosting:** The Gradient Boosting model was applied to improve classification accuracy by combining weak learners (decision trees) in an iterative boosting process. This model captures complex patterns in the data for better crop recommendations.

**E. Output Layer**

The Output Layer of the system provides the final crop recommendation based on the input data and the trained machine learning models. The system predicts the most suitable crop for cultivation under the given environmental and soil conditions. The output label represents the recommended crop, such as Rice, Wheat, Cotton, or other crops, depending on the conditions.

For each input instance, the system predicts the appropriate crop category, providing actionable insights for farmers or agricultural experts. The model outputs can also include the probability or confidence level of the recommended crop, giving further context to the decision.

**F. Results**

The system's output provides a clear classification of the most suitable crop for cultivation based on the input features. Each model's performance is evaluated using metrics such as accuracy, precision, recall, and F1-score. A confusion matrix is used to analyze the true positives, false positives, true negatives, and false negatives, allowing for a detailed evaluation of each model's prediction capabilities.

The overall system helps in recommending crops based on real-time environmental and soil conditions, offering valuable decision-making insights for optimizing agricultural productivity.

**3.5 Model Evaluation**

In machine learning, several metrics are used to assess the performance of a model depending on the type of problem (classification, regression, clustering, etc.). For classification tasks, common metrics include:

Accuracy: The ratio of correct predictions to the total number of predictions. While commonly used, it can be misleading for imbalanced datasets.

Precision: Measures the proportion of true positive predictions out of all positive predictions. It is useful when the cost of false positives is high.

Recall (Sensitivity): The proportion of true positives identified out of all actual positives. High recall is critical in scenarios where false negatives are costly.

F1-Score: The harmonic mean of precision and recall, providing a single score that balances both. It is useful when precision and recall need to be optimized together.

Confusion Matrix: The confusion matrix highlighted areas where the model might struggle, such as recommending crops for borderline or ambiguous growing conditions.

ROC-AUC: The area under the Receiver Operating Characteristic curve provides an aggregate measure of performance across all classification thresholds. It is especially useful for binary classification problems.

When comparing models, accuracy is often the first metric considered, especially for classification tasks. However, comparing models based solely on accuracy can be misleading, particularly when dealing with imbalanced datasets. For instance, a model that predicts the majority class in an imbalanced dataset may higher accuracy, another may have better precision and recall, making it more suitable for certain applications. Additionally, cross-validation methods were employed to ensure that the accuracy reported for each model is generalized across different subsets of data, rather than being specific to the training set.

Individual Model Performance

Logistic Regression:

With a maximum of 1000 iterations to ensure convergence, Logistic Regression produced competitive results in terms of accuracy, precision, recall, and F1-score.

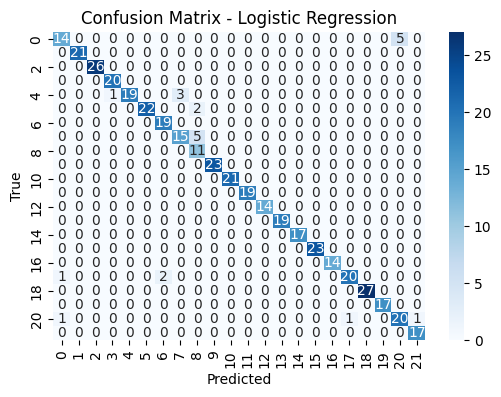
\*\*Logistic Regression\*\*

Accuracy: 0.9500

Precision: 0.9559

Recall: 0.9500

F1-score: 0.9506



Naive Bayes:

The Naive Bayes classifier performed well, particularly in high-dimensional data, yielding decent accuracy despite some assumptions about feature independence.

\*\*Naive Bayes\*\*

Accuracy: 0.9909

Precision: 0.9919

Recall: 0.9909

F1-score: 0.9909

A graph of a number

Description automatically generated

K-Nearest Neighbors (KNN):

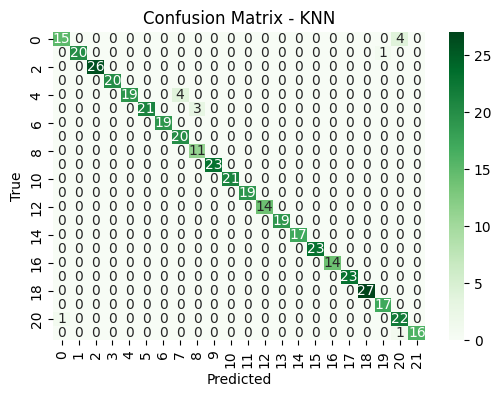
The KNN classifier provided a good balance between simplicity and performance.

Accuracy: 0.9682

Precision: 0.9725

Recall: 0.9682

F1-score: 0.9684



Decision Tree:

The model exhibited robust performance with balanced accuracy, and it excelled in precision and recall for specific crop types.

\*\*Decision Tree\*\*

Accuracy: 0.9886

Precision: 0.9890

Recall: 0.9886

F1-score: 0.9886

A graph of a number

Description automatically generated with medium confidence

Support Vector Machine (SVM):

Probability estimates were enabled during training, which facilitated detailed performance assessments. SVM showed strong performance, especially in precision and recall metrics.

\*\*SVM\*\*

Accuracy: 0.9636

Precision: 0.9687

Recall: 0.9636

F1-score: 0.9634

A diagram of a graph

Description automatically generated

Gradient Boosting:

\*\*Gradient Boosting\*\*

Accuracy: 0.9864

Precision: 0.9876

Recall: 0.9864

F1-score: 0.9865

A graph of a number

Description automatically generated

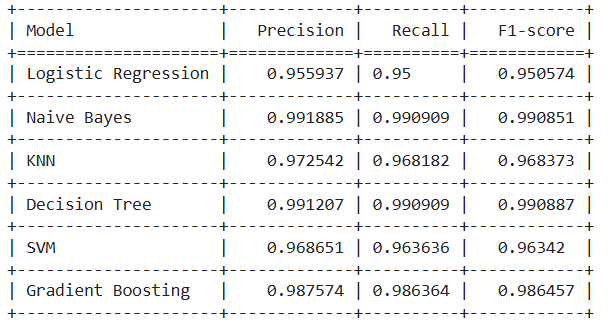


Table 1. Recorded Results for each Classifier

1. **Quality Assurance**: Model evaluation helps ensure that the model is capable of making accurate predictions when exposed to real-world data. It acts as a quality control mechanism to validate the model's generalization ability.
2. **Comparing Models**: Model evaluation allows for the comparison of multiple models to identify the best-performing one. It helps data scientists and stakeholders make informed decisions about which model to deploy.
3. **Fine-Tuning**: The evaluation process can reveal areas where the model performs poorly. This information is valuable for refining the model, making it more robust, and addressing its limitations.
4. **Business Decision Support**: In practical applications, model performance impacts critical business decisions. A well-evaluated model provides confidence to stakeholders, leading to better decision-making.
5. **Model Deployment**: A thoroughly evaluated model is more likely to be deployed in production systems. It instils trust in the model's predictions, which is essential in real- world applications.

**3.6 Constraints**

Data Quality and Availability:

Data Accuracy: Ensure that the dataset used is accurate, representative, and up to date.

Imbalanced Datasets: If the dataset is imbalanced (like in medical or binary classification problems), appropriate techniques like SMOTE should be applied to avoid biased predictions.

Computational Constraints:

Memory and Processing Power: Large datasets or complex models (like Gradient Boosting or SVM) require significant processing power and memory, which can be a limitation on certain hardware.

Training Time: Models like Gradient Boosting can take considerable time to train compared to simpler models like Logistic Regression or Naive Bayes.

Model Interpretability:

Complex models like Gradient Boosting and SVM might be less interpretable compared to simpler models such as Decision Trees or Logistic Regression. If explainability is crucial (e.g., in healthcare or legal domains), this can be a significant constraint.

Accuracy vs. Generalization:

Overfitting can occur if the model is too complex, which may result in high training accuracy but poor performance on unseen data.

Regulatory Compliance:

Depending on the industry, there could be regulatory constraints, such as GDPR (General Data Protection Regulation) for user privacy, or specific health-related data protections like HIPAA in the medical field.

**3.7 Ethical Considerations**

Bias and Fairness:

Machine learning models may inadvertently reinforce biases present in the data. Ethical consideration must be taken to ensure fairness, especially in sensitive domains like healthcare, recruitment, and criminal justice.

Discrimination: The model must not discriminate against any group based on race, gender, socioeconomic status, etc.

Privacy and Security:

Ensure that personal or sensitive data (such as in medical datasets) is anonymized and secure.

Data should be handled according to ethical standards and legal regulations for privacy, particularly if the data contains personally identifiable information (PII).

Transparency and Accountability:

The decision-making process of the model should be as transparent as possible. Especially in high-stakes areas like healthcare or legal decisions, users should be able to understand how predictions are made. Clear documentation and explanation of the machine learning models used should be provided to stakeholders.

Responsible Use of AI:

Misinformation: Ensure that the results from the model are not used to spread misinformation or harm individuals.

Automation of Decision Making: Care should be taken when automating decisions that significantly affect people's lives, such as healthcare treatment recommendations or financial approvals. Human oversight is often necessary.

Environmental Impact:

Training large models can consume significant energy, contributing to environmental impact. Consider energy-efficient models and hardware.

Informed Consent:

If personal data is used, it is essential to obtain consent from users or subjects and ensure they are aware of how their data will be used.

**CHAPTER-4**

**IMPLEMENTATION**

**4.Implementation**

# **4.1 Environment Setup**

To ensure the smooth execution of our crop recommendation system, we utilized a robust environment tailored for data analysis and machine learning tasks. The project was primarily executed in Python, supported by a variety of libraries that facilitated data manipulation, model training, and result visualization. Key libraries included NumPy for numerical computations, pandas for data processing, and matplotlib and seaborn for visualizing data insights. Machine learning algorithms, such as logistic regression, decision trees, and support vector machines, were implemented using the scikit-learn library.

For this project, Google Colab was used as the primary environment, allowing easy access to powerful cloud-based computing resources, eliminating the need for local hardware constraints, and simplifying package management. The dataset was uploaded into Colab either directly or via Google Drive. Data preprocessing included encoding categorical variables, addressing missing values, and applying feature scaling to prepare the data for modeling.

Google Colab’s virtual environment provided sufficient computational power for efficient data processing and model training, ensuring seamless execution of the crop recommendation system.

# **4.2 Sample Code for Preprocessing and Model Training and Testing**

**1. Data Preprocessing**

This part of the code involves handling missing values, encoding categorical features, and scaling numerical features.

# Import necessary libraries

import pandas as pd

from sklearn.model\_selection import train\_test\_split

from sklearn.preprocessing import StandardScaler, LabelEncoder

from sklearn.impute import SimpleImputer

# Load the dataset

data = pd.read\_csv('Crop\_recommendation.csv')

# Checking for missing values and handling them

imputer = SimpleImputer(strategy='mean')

data['feature\_with\_missing\_values'] =imputer.fit\_transform(data[['feature\_with\_missing\_values']])

# Encoding categorical variables

label\_encoder = LabelEncoder()

data['categorical\_feature'] = label\_encoder.fit\_transform(data['categorical\_feature'])

# Splitting data into features (X) and target (y)

X = data.drop(columns=['target\_column']) # replace 'target\_column' with your target variable name

y = data['target\_column']

# Split the dataset into training and testing sets (80% training, 20% testing)

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Feature scaling

scaler = StandardScaler()

X\_train = scaler.fit\_transform(X\_train)

X\_test = scaler.transform(X\_test)

**2. Model Training**

This part demonstrates training several machine learning models like Logistic Regression, Decision Tree, Naive Bayes, SVM, Gradient Boosting and KNN.

# Import machine learning models

from sklearn.linear\_model import LogisticRegression

from sklearn.tree import DecisionTreeClassifier

from sklearn.naive\_bayes import GaussianNB

from sklearn.svm import SVC

from sklearn.ensemble import GradientBoostingClassifier

from sklearn.neighbors import KNeighborsClassifier

# Instantiate the models

logistic\_regression = LogisticRegression(random\_state=42)

decision\_tree = DecisionTreeClassifier(random\_state=42)

naive\_bayes = GaussianNB()

svm\_model = SVC(random\_state=42)

gradient\_boosting = GradientBoostingClassifier(random\_state=42)

knn = KNeighborsClassifier()

# Train the models

logistic\_regression.fit(X\_train, y\_train)

decision\_tree.fit(X\_train, y\_train)

naive\_bayes.fit(X\_train, y\_train)

svm\_model.fit(X\_train, y\_train)

gradient\_boosting.fit(X\_train, y\_train)

knn.fit(X\_train, y\_train)

**3. Model Testing and Evaluation**

After training, evaluate the models' performance using metrics such as accuracy, precision, recall, and F1-score.

# Import metrics for evaluation

from sklearn.metrics import accuracy\_score, precision\_score, recall\_score, f1\_score, confusion\_matrix

# Function to evaluate models

def evaluate\_model(y\_test, y\_pred, model\_name):

print(f"Results for {model\_name}:")

print(f"Accuracy: {accuracy\_score(y\_test, y\_pred):.4f}")

print(f"Precision: {precision\_score(y\_test, y\_pred, average='macro'):.4f}")

print(f"Recall: {recall\_score(y\_test, y\_pred, average='macro'):.4f}")

print(f"F1-Score: {f1\_score(y\_test, y\_pred, average=‘macro'):.4f}")

# List of models and their names

models = [

(logistic\_regression, "Logistic Regression"),

(decision\_tree, "Decision Tree"),

(naive\_bayes, "Naive Bayes"),

(svm\_model, "SVM"),

(gradient\_boosting, "Gradient Boosting"),

(knn, "K-Nearest Neighbors")

]

# Loop through the models, make predictions, and evaluate

for model, model\_name in models:

y\_pred = model.predict(X\_test)

evaluate\_model(y\_test, y\_pred, model\_name)

**Explanation:**

Data Preprocessing:

Missing values are handled using SimpleImputer.

Categorical variables are encoded using LabelEncoder.

Features are scaled using StandardScaler.

The dataset is split into training and testing sets.

Model Training:

Several models are trained using logistic regression, decision tree,Naive Bayes, SVM, Gradient Boosting and KNN .

Model Evaluation:

The models are tested on the test set, and performance metrics like accuracy, precision, recall, F1-score are computed.

**CHAPTER-5**

**Experimentation and Result Analysis**

**5. Experimentation and Result Analysis**

The experimentation followed a structured approach, starting with data preprocessing that involved cleaning the dataset, handling missing values, label encoding, and scaling features like N, P, K, temperature, and rainfall. The data was split into training (80%) and testing (20%) sets. Key features were selected using embedded methods like Decision Trees and Logistic Regression, along with correlation-based techniques. Several machine learning models, including Logistic Regression, KNN, SVM, Decision Trees, and Gradient Boosting, were trained on the preprocessed data.

**Result Analysis**

**1. Model Performance Metrics**

The performance of each model was evaluated using common metrics such as:

Accuracy: Percentage of correctly classified crops out of total predictions.

Precision: Ratio of correctly predicted positive observations to the total predicted positives.

Recall: Ability of the model to find all the relevant cases within a dataset.

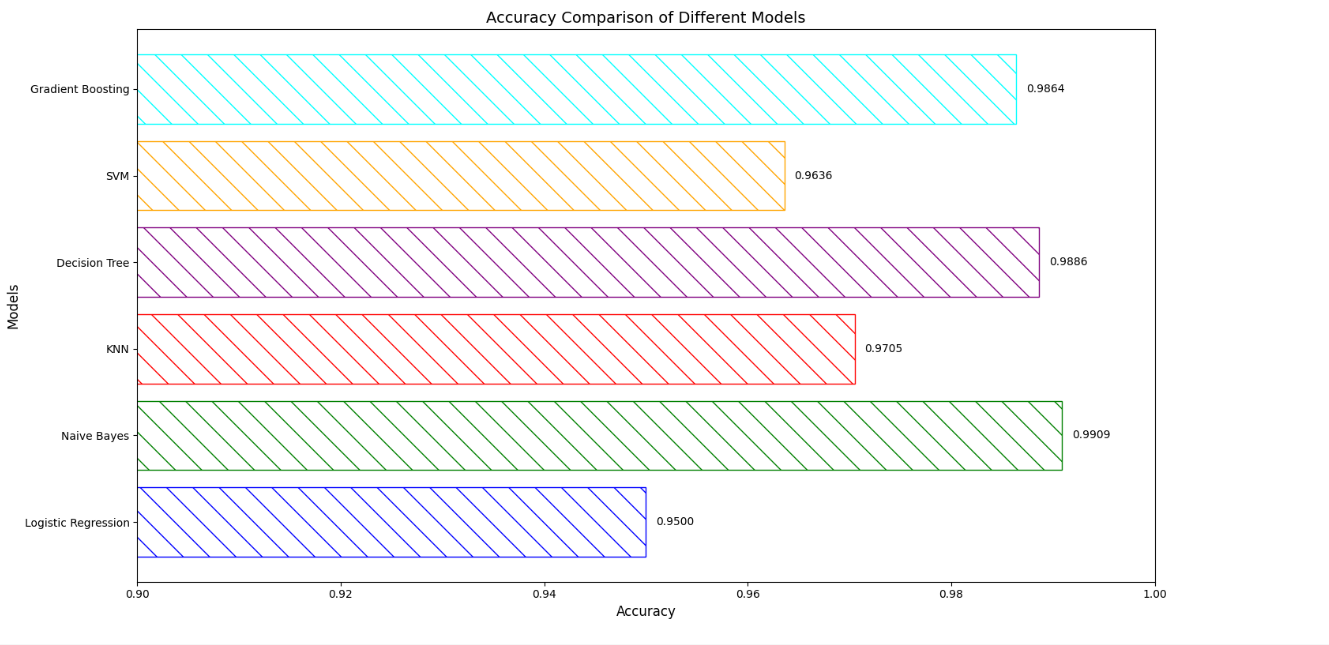
F1-Score: Harmonic mean of precision and recall, providing a balance between the two.

Confusion Matrix: For analyzing the performance in terms of true positives, true negatives, false positives, and false negatives.

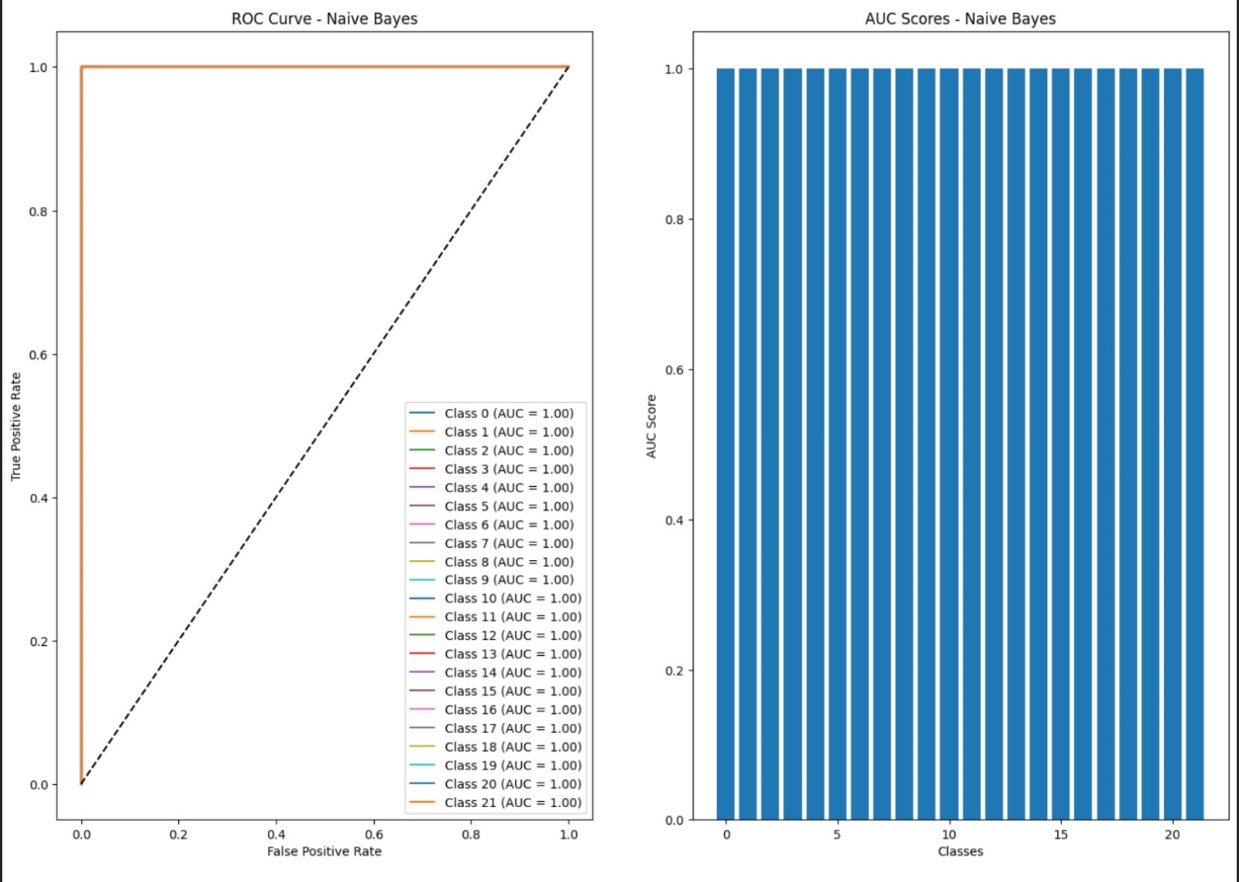
**2. Comparative Analysis of Models**

The models were compared to identify which one performs best for crop recommendation. The

following is an example of how the results might be analyzed:



Naive Bayes emerged as the best-performing model with an accuracy of 99.09%, along with high precision, recall, and F1-score. This indicates that Naive Bayes is robust for this crop recommendation task due to its effectiveness in handling the underlying probabilistic relationships in the data.This is the ROC curve for naïve bayes.



**CHAPTER-6**

**CONCLUSION**

**6.Conclusion**

This project demonstrates the potential of machine learning approaches to optimize crop recommendations for farmers based on various environmental and soil conditions. By applying algorithms such as Logistic Regression, Decision Tree, Naive Bayes, SVM, Gradient Boosting, and K-Nearest Neighbors (KNN), in optimizing crop recommendations based on soil and environmental conditions. These models, trained on complex agricultural datasets, provide actionable insights that can improve decision-making for farmers. However, the success of these algorithms is heavily influenced by the quality of input data, making robust data management practices essential.

Moving forward, addressing challenges related to data quality and the interpretability of machine learning models will be crucial for their widespread adoption. Enhancing transparency in model predictions will allow agricultural professionals to better trust and apply the insights generated. Integrating additional data sources, such as satellite imagery and real-time climate data, will further improve model accuracy, helping to refine crop recommendations and better support farmers in diverse regions.

Ultimately, this study showcases the potential of machine learning to revolutionize precision agriculture by improving crop yield predictions and tailoring farming strategies to local conditions. The continued refinement of these models, along with collaborative efforts between agronomists, data scientists, and farmers, will be key to overcoming existing limitations and fully realizing the benefits of machine learning in agricultural management.

In summary, this study highlights the significant potential of machine learning in the agricultural sector. As these technologies advance, they could play a transformative role in improving crop yield, optimizing land use, and ensuring food security. Continued collaboration between data scientists, agronomists, and farmers will be crucial in unlocking the full potential of machine learning in crop management.

**CHAPTER-7**

**REFRENCES**

### **REFERENCES**

[1] S. Sanjana, K. N. Kumar, P. Satyanarayana, K. N. V. Suresh Varma, I. Anitha, and V. G. Krishnan, “Implementation of Optimal Crop Recommendation System using Machine Learning Algorithms,” *Proc. - 2024 Int. Conf. Expert Clouds Appl. ICOECA 2024*, pp. 678–683, 2024, doi: 10.1109/ICOECA62351.2024.00123.

[2] P. Parameswari, N. Rajathi, and K. J. Harshanaa, “Machine Learning Approaches for Crop Recommendation,” *2021 Int. Conf. Adv. Electr. Electron. Commun. Comput. Autom. ICAECA 2021*, pp. 1–5, 2021, doi: 10.1109/ICAECA52838.2021.9675480.

[3] B. Dey, J. Ferdous, and R. Ahmed, “Machine learning based recommendation of agricultural and horticultural crop farming in India under the regime of NPK, soil pH and three climatic variables,” *Heliyon*, vol. 10, no. 3, p. e25112, 2024, doi: 10.1016/j.heliyon.2024.e25112.

[4] J. Madhuri and M. Indiramma, “Artificial Neural Networks Based Integrated Crop Recommendation System Using Soil and Climatic Parameters,” *Indian J. Sci. Technol.*, vol. 14, no. 19, pp. 1587–1597, 2021, doi: 10.17485/ijst/v14i19.64.

[5] R. K. Ray, S. K. Das, and S. Chakravarty, “Smart Crop Recommender System-A Machine Learning Approach,” *Proc. Conflu. 2022 - 12th Int. Conf. Cloud Comput. Data Sci. Eng.*, pp. 494–499, 2022, doi: 10.1109/Confluence52989.2022.9734173.

[6] D. Garg and M. Alam, “An effective crop recommendation method using machine learning techniques,” *Int. J. Adv. Technol. Eng. Explor.*, vol. 10, no. 102, pp. 498–514, 2023, doi: 10.19101/IJATEE.2022.10100456.

[7] A. Priyadharshini, S. Chakraborty, A. Kumar, and O. R. Pooniwala, “Intelligent Crop Recommendation System using Machine Learning,” *Proc. - 5th Int. Conf. Comput. Methodol. Commun. ICCMC 2021*, no. Iccmc, pp. 843–848, 2021, doi: 10.1109/ICCMC51019.2021.9418375.

[8] M. Choudhary, R. Sartandel, A. Arun, and L. ladge, “Crop Recommendation System and Plant Disease Classification using Machine Learning for Precision Agriculture,” *Artif. Intell. Commun. Technol.*, pp. 39–49, 2022, doi: 10.52458/978-81-955020-5-9-4.

[9] G. Gayatri, K. N. V. S. Praharsha, K. Hemanth, and M. Owk, “Crop Recommendation System Using Machine Learning,” *Springer Proc. Math. Stat.*, vol. 421, pp. 735–742, 2024, doi: 10.1007/978-3-031-51167-7\_70.

[10] S. Hossain, N. N. I. Prova, M. R. Sadik, and A. Al Maruf, “Enhancing Crop Management: Ensemble Machine Learning for Real-Time Crop Recommendation System from Sensor Data,” *Int. Conf. Smart Syst. Appl. Electr. Sci. ICSSES 2024*, pp. 1–6, 2024, doi: 10.1109/ICSSES62373.2024.10561406.

[11] S. M. Pande, P. K. Ramesh, A. Anmol, B. R. Aishwarya, K. Rohilla, and K. Shaurya, “Crop Recommender System Using Machine Learning Approach,” *Proc. - 5th Int. Conf. Comput. Methodol. Commun. ICCMC 2021*, no. Iccmc, pp. 1066–1071, 2021, doi: 10.1109/ICCMC51019.2021.9418351.

[12] D. Modi, A. V. Sutagundar, V. Yalavigi, and A. Aravatagimath, “Crop Recommendation Using Machine Learning Algorithm,” *2021 5th Int. Conf. Inf. Syst. Comput. Networks, ISCON 2021*, pp. 1–5, 2021, doi: 10.1109/ISCON52037.2021.9702392.

[13] G. Chauhan and A. Chaudhary, “Crop Recommendation System using Machine Learning Algorithms,” *Proc. 2021 10th Int. Conf. Syst. Model. Adv. Res. Trends, SMART 2021*, pp. 109–112, 2021, doi: 10.1109/SMART52563.2021.9676210.

[14] M. Y. Raghavendra and H. S. Annapurna, “A Systematic Literature Review: Crop Recommendation Using Machine Learning Algorithms,” *2024 Int. Conf. Knowl. Eng. Commun. Syst. ICKECS 2024*, vol. 1, pp. 1–6, 2024, doi: 10.1109/ICKECS61492.2024.10617047.

[15] D. Dahiphale, P. Shinde, K. Patil, and ..., “Smart farming: Crop recommendation using machine learning with challenges and future ideas,” *UMBC Student Collect.*, pp. 0–11, 2023, [Online]. Available: https://mdsoar.org/handle/11603/28455%0Ahttps://mdsoar.org/bitstream/handle/11603/28455/crop\_recommendation\_using\_machine\_leanring\_preprint.pdf?sequence=1&isAllowed=y

[16] S. Desai, M. Joshi, and P. Mane, “Intelligent Crop Recommendation System Using Machine Learning Algorithms,” *Commun. Comput. Inf. Sci.*, vol. 1781 CCIS, pp. 277–289, 2023, doi: 10.1007/978-3-031-35641-4\_23.

[17] S. A. Z. Rahman, K. C. Mitra, and S. M. M. Islam, “Soil Classification Using Machine Learning Methods and Crop Suggestion Based on Soil Series,” *2018 21st Int. Conf. Comput. Inf. Technol. ICCIT 2018*, pp. 1–4, 2018, doi: 10.1109/ICCITECHN.2018.8631943.

[18] S. P. Raja, B. Sawicka, Z. Stamenkovic, and G. Mariammal, “Crop Prediction Based on Characteristics of the Agricultural Environment Using Various Feature Selection Techniques and Classifiers,” *IEEE Access*, vol. 10, pp. 23625–23641, 2022, doi: 10.1109/ACCESS.2022.3154350.

[19] L. K. Seru and S. A. I. M. Gandham, “Crop Recommendation System Using Machine Department of Computer Science and Engineering School of Computing Sathyabama Institute of Science and Technology Jeppiaar Nagar , Rajiv Gandhi Salai,” no. April, pp. 1–51, 2022.

[20] I. Journal, “Crop Recommendation System,” *Interantional J. Sci. Res. Eng. Manag.*, vol. 07, no. 10, pp. 1–11, 2023, doi: 10.55041/ijsrem26385.