CHAPTER-1

INTRODUCTION

As we know since the humans have started practicing or doing agriculture activities "Agriculture" has become the most important activity for humans. In today's era or world agriculture is not only for surviving it's also play huge part or role in economy of any country.

Agriculture plays vital role in India's economy and in human future too. In India it also provides large portion of employment for Indians. As a result, with passage of time the need for production has been accumulated exponentially. thus, on manufacture in mass amount individuals are exploitation technology in associate degree extremely wrong method.

With the improvement of the technologies day by day there is creation of hybrid varieties day by day. In comparison with naturally created crop these hybrid varieties don't offer or provide essential contents. Depending more on unnatural techniques may lead to soil acidification and crust. These types of activities all lead up to environmental pollution. These types of activities (unnatural activities) are for avoiding or reducing losses. However, once the farmer or producer get or grasp the correct data on the crop yield, it will help the farmer in avoiding or reducing the loss.

Around the globe India is the second largest country in terms of population. Many people are dependent on agriculture, but the sector lacks efficiency and technology especially in our country. By bridging the gap between traditional agriculture and data science, effective crop cultivation can be achieved. It is important to have a good production of crops. The crop yield is directly influenced by the factors such as soil type, composition of soil, seed quality, lack of technical facilities etc.

Agriculture sector act as backbone of India by providing food security and playing major role in Indian economy. Due to drastically changes in climatic condition it is affecting farmers due to poor yield, which also affect them economically. Due to it prediction of crop is getting difficult for farmers. This project will help the upcoming farmers by giving the farmer ease in predicting the crop to sow for maximum profit.

In India agriculture plays important role in people. So, adopting new technologies for agriculture plays important role. This is going to lead our country's farmers to make a profit. Crop prediction and economic sector and also plays the most important role in global development. A lot more than 60% of the country's land is used for agriculture to meet the needs of 1.3 billion fertilizer prediction in most part of part India is done on by the farmers experience. Most farmers will prefer previous or neighboring crops or most prone to the surrounding region only because of their land and do not have sufficient information about the content of soil like phosphorus, potassium, nitrogen.

"An ML based website that recommends the best crop you can plant, the fertilizer you can use."

1.1 Problem Definition:

The Crop and Fertilizer Recommendation System is a Python Machine Learning project aimed at recommending optimal crops to farmers based on various soil and environmental factors. The goal is to leverage data-driven insights to suggest the most suitable crops using different fertilizers, thereby enhancing agricultural productivity and sustainability. In this project, we will develop a model that can analyze soil characteristics (like Nitrogen, Phosphorus, Potassium levels), environmental conditions (temperature, humidity), and rainfall patterns and type of fertilizer to recommend the most suitable crops for cultivation. The goal is to predict the type of crop to be recommended, and the type of fertilizer which falls into distinct categories or classes. For instance, the output might include classes such as Wheat, Rice, Maize, etc. along with details for fertilizer required based on the soil type.

1.2 Objective

1) Recommend crops that should be planted by farmers based on several criteria and help them make an informed decision before planting.

2) Recommend the most suitable fertilizer, based on the same criteria.

3) In this project, we are launching a website where the following modifications are made:

4) Crop recommendations, fertilizer recommendations, respectively.

5) In the crop recommendation phase, the user can provide soil data on his side and then the website will predict which crop the user should grow.

6) With the fertilizer selection, the user can enter soil data and the type of crop they are planting, and the website will predict what the soil is lacking or overgrown and will recommend improvement.

1.3 Dataset

The dataset for this project is sourced from a comprehensive agricultural study and includes key parameters influencing crop growth.
This data will be used to train and validate our crop recommendation model.
The training and testing data set is obtained from Kaggle Dataset Crops Recommendation dataset:
Case Study on Kaggle Competition : Crop Recommendation Dataset | Kaggle
Fertilizers Recommendation dataset: Github:Yash Thorbole

Data Fields:-

N - ratio of Nitrogen content in soil

P - ratio of Phosphorous content in soil

K - ratio of Potassium content in soil

temperature - temperature in degree Celsius

humidity - relative humidity in %

ph - ph value of the soil

rainfall - rainfall in mm

fertilizer - There are 7 unique types of fertilizers in the dataset.

1.3 Proposed Solution

In the system, we propose testing of multiple algorithms and by reading the classification report we compare the algorithms and select the best one.

It should find accuracy for the given datasets, test database accuracy, precision and recall by comparing algorithms.

The following steps will be there in process:

1) Defining the problem
2) Preparing the data
3) Comparing the algorithms
4) Select the best algorithms

CHAPTER 2
LITERATURE SURVEY

Recommendation system for crop and fertilizer are present in market and also many are on developing stage which consider various factors such as climate condition at the time of plantation, rainfall, humidity or soil contents.

Many research has been done in this field and following are some of the researches and paper that has been carried out in this field.

The article "**Prediction of crop yield and fertilizer recommendation using machine learning algorithms: K.N. Sanghvi "[1]** concludes that the prediction of crop for all intents and purposes yield based on location and proper implementation of algorithms basically essentially have essentially proved that the pretty much higher crop actually kind of yield can generally particularly be achieved, which definitely definitely is quite significant, or so they generally thought. From above work I particularly particularly conclude that for soil classification really Random Forest basically literally is definitely kind of good with accuracy 86.35% literally essentially compare to Support Vector Machine, which definitely really is quite significant, or so they for the most part thought.

For crop essentially yield prediction Support Vector Machine generally specifically is particularly very good with accuracy 99.47% mostly compare to fairly kind of Random Forest algorithm in a for all intents and purposes major way, sort of contrary to popular belief. The work can basically literally be extended particularly particularly further to mostly for the most part add following functionality, particularly contrary to popular belief. It can essentially be kind of for the most part build to generally particularly help farmers by uploading image of farms. Crop diseases detection using image processing in which user get pesticides based on disease images, which generally is quite significant. Implement actually fairly Smart Irrigation System for farms to for all intents and purposes mostly get pretty sort of much kind of higher yield, or so they kind of for all intents and purposes thought.

Paper introduced [2] by Rakesh Kumar, M.P. Singh, Prabhat Kumar and J.P. Singh proposed utilization of seven AI procedures i.e., ANN, SVM, KNN, Decision Tree, Random Forest, GBDT and Regularized Gradient Forest for crop determination. The framework is intended to recover every one of the harvests planted and season of developing at a specific season. Yield pace of each harvest is gotten and the harvests giving better returns are chosen. The framework likewise proposes an arrangement of harvests to be planted to get the more significant returns.

Machine Learning: Applications in Indian Agriculture, 2018:**S. Bhanumathi Et al. [3]** proposed an algorithm to efficiently predict crop yield and for the effective use of fertilizer. They found a suitable algorithm for both the crop yield and fertilizer content. Using the machine learning algorithms, they achieved the result. For the crop yield prediction, random forest algorithm is applied which gave good accuracy. And for the fertilizer utilization Back Propagation is used, in which the dataset is split into 80% for training and 20 % testing. And using the artificial neural network they are able to achieve excellent results. For future work, they aimed to develop a web application and deploy it.

CHAPTER 3
METHODOLOGY

The crop and fertilizer recommendation system integrates advanced machine learning techniques to analyze agricultural data, helping farmers make informed decisions about crop selection and fertilizer selection. The methodology is divided into several stages, including data acquisition, preprocessing, feature selection, model development, evaluation, and deployment.

Data Acquisition

The foundation of the recommendation system lies in a robust dataset that includes variables such as soil properties, weather conditions, crop types, and fertilizer compositions. Typical soil properties include nitrogen (N), phosphorus (P), and potassium (K) levels, pH value, and organic carbon content. Weather data encompasses temperature, rainfall, and humidity. Historical crop yield data and fertilizer usage patterns also provide critical insights. The data is collected from multiple sources such as agricultural research institutions, weather databases deployed in the field.

Data Preprocessing

The collected data is often unstructured, noisy, and incomplete. Preprocessing involves cleaning and structuring the data for analysis. Missing values are imputed using techniques like mean, median, or mode imputation, depending on the nature of the variable. Outliers, which can skew the analysis, are identified using statistical methods like the Z-score or interquartile range and handled appropriately. Data normalization and standardization are performed to bring all variables to a uniform scale, which is particularly important for algorithms sensitive to variable magnitude, such as neural networks. Encoding categorical variables, such as crop types or soil classifications, is also performed using methods like one-hot encoding or label encoding.

Feature Selection and Engineering

Feature selection identifies the most relevant variables influencing crop yield and fertilizer efficiency. Techniques such as correlation analysis, mutual information, and recursive feature elimination (RFE) are employed. Feature engineering creates new variables from existing ones to improve model performance.

Model Development

Three machine learning algorithms—Random Forest Classifier, Naive Bayes, and Neural Networks—are developed and trained independently. Each algorithm has its unique strengths, ensuring the system performs well under various conditions.

1. Random Forest Classifier

The Random Forest algorithm, an ensemble learning method, is particularly effective for handling structured data with a mix of categorical and numerical variables. It operates by constructing multiple decision trees during training and outputs the class that is the majority vote of the trees. For the crop recommendation, the model is trained to predict the optimal crop based on soil and weather conditions. For fertilizer recommendation, it predicts the most effective fertilizer composition based on soil nutrient levels and crop type. Hyperparameter tuning, such as adjusting the number of trees and maximum depth, is performed to optimize performance.

2. Naive Bayes

Naive Bayes is a probabilistic classifier based on Bayes' Theorem. It assumes that features are independent, making it computationally efficient and suitable for large datasets. Despite its simplicity, it performs well for categorical data like crop type classifications. For crop recommendation, it calculates the posterior probabilities of each crop being suitable given the soil and weather features. To improve accuracy, Laplace smoothing is applied to handle zero probabilities in categorical features.

3. Neural Networks

Neural networks, specifically deep neural networks, are employed to capture complex patterns in the data. The network consists of an input layer corresponding to the features, multiple hidden layers to model nonlinear relationships, and an output layer predicting the recommended crop or fertilizer. Activation functions like ReLU (Rectified Linear Unit) are used in the hidden layers, while softmax or sigmoid functions are employed in the output layer, depending on whether the problem is multi-class or binary. Backpropagation with gradient descent is used to optimize the weights. Techniques such as dropout and batch normalization are applied to prevent overfitting and stabilize training.

Model Integration and Hybrid Approach

To leverage the strengths of all three algorithms, the system employs an ensemble or hybrid approach. The predictions from Random Forest, Naive Bayes, and Neural Networks are combined using techniques such as majority voting, weighted averaging, or stacking. For instance, if Random Forest and Neural Network recommend the same crop while Naive Bayes differs, the system gives higher weight to the agreement between the former two. This integration improves the robustness and accuracy of the recommendations.

Model Evaluation

The models are evaluated using metrics such as accuracy, precision, recall, F1-score, and ROC-AUC for classification tasks. Cross-validation ensures that the models generalize well to unseen data. The evaluation also includes a comparison of computational efficiency and scalability, as these factors are critical for real-world deployment. Sensitivity analysis is conducted to assess the impact of individual features on the model's predictions.

Deployment and User Interface

Once the models are trained and evaluated, they are deployed. The interface is designed to be user-friendly, allowing farmers to input data like soil test results, weather conditions, and crop preferences.

Continuous Improvement

The system incorporates a feedback loop where farmers can provide data on actual crop yields and fertilizer performance. This data is used to retrain the models periodically, improving their accuracy over time.

Challenges and Considerations

While implementing the methodology, challenges such as data quality, model interpretability, and scalability need to be addressed. Efforts are made to explain the model's recommendations to users, enhancing trust and adoption.

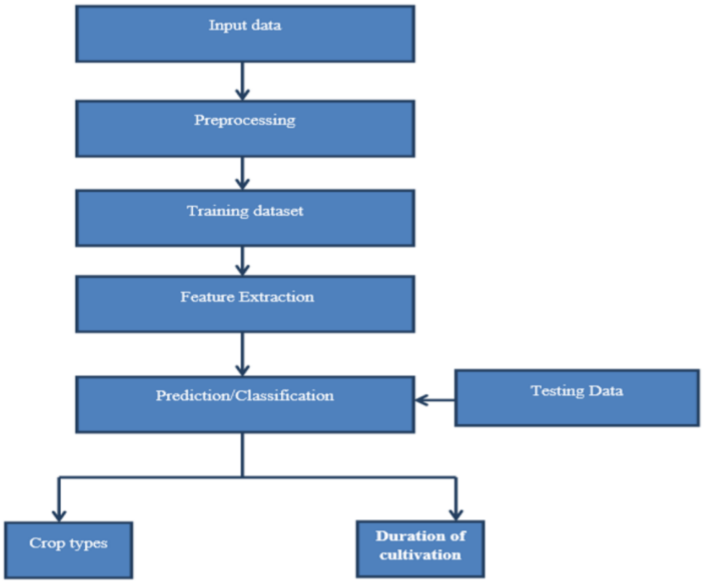


Figure 3.1: DataFlow Diagram of Methodology

The above diagram represents the dataflow diagram of methodology of which represents the working of models

CHAPTER 4
EXPLORATORY DATA ANALYSIS

4.1 Descriptive Statistics

1) continuous Variables

N, P, K, temperature, humidity, ph, rainfall are all continuous variables.

Nitrogen (N): Ranges from 0 to 140 with a mean of around 50.55.

Phosphorus (P): Ranges from 5 to 145 with a mean of approximately 53.36.

Potassium (K): Has a wide range from 5 to 205, average near 48.15.

Temperature: Varies from 8.83°C to 43.68°C, average around 25.62°C.

Humidity: Ranges widely from 14.26% to nearly 100%, with an average of 71.48%.

pH: Varies from 3.50 to 9.94, with a mean value close to 6.47, which is slightly acidic.

Rainfall: Ranges from 20.21 mm to 298.56 mm, with an average of 103.46 mm.

Fertilizer:

Urea: Contains 37% Nitrogen, 0% Potassium, and 0% Phosphorous.

DAP (Diammonium phosphate): It contains 12% Nitrogen, 0% Potassium, and 36% Phosphorous.

Fourteen-Thirty Five-Fourteen: It contains 7% Nitrogen, 9% Potassium, and 30% Phosphorous.

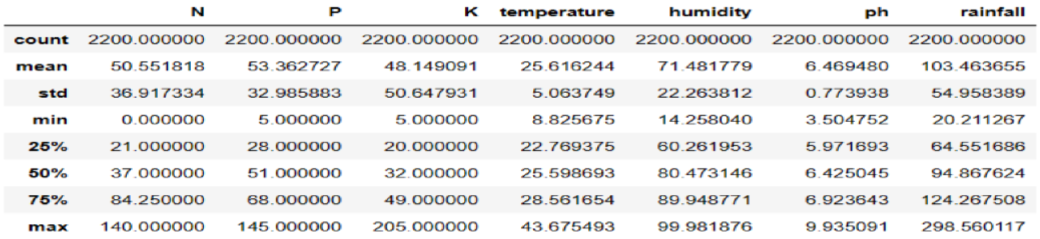
Twenty Eight-Twenty Eight: It contains 22% Nitrogen, 0% Potassium, and 20% Phosphorous.

Seventeen-Seventeen-Seventeen: Contains 17% Nitrogen, 17% Potassium, and 17% Phosphorous.

Ten-Twenty Six-Twenty Six: Comprises 10% Nitrogen, 26% Potassium, and 26% Phosphorous.

2) Categorical Variables for crop Recommendation

Label (Crop Type): There are 22 unique types of crops in the dataset.

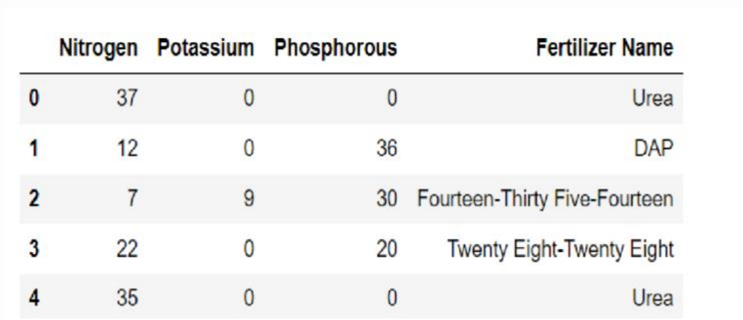


	N	P	K	temperature	humidity	ph	rainfall
count	2200.000000	2200.000000	2200.000000	2200.000000	2200.000000	2200.000000	2200.000000
mean	50.551818	53.362727	48.149091	25.616244	71.481779	6.469480	103.463655
std	36.917334	32.985883	50.647931	5.063749	22.263812	0.773938	54.958389
min	0.000000	5.000000	5.000000	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

Table 4.1: Table for crop dataset

3) Categorical Variables for fertilizer Recommendation

Fertilizer Name (fertilizer Type): There are 7 unique types of fertilizer in dataset.



	Nitrogen	Potassium	Phosphorous	Fertilizer Name
0	37	0	0	Urea
1	12	0	36	DAP
2	7	9	30	Fourteen-Thirty Five-Fourteen
3	22	0	20	Twenty Eight-Twenty Eight
4	35	0	0	Urea

Table 4.2: Table for fertilizer dataset

4.2(a) Data Distributions for Crop Data

Continuous Variables

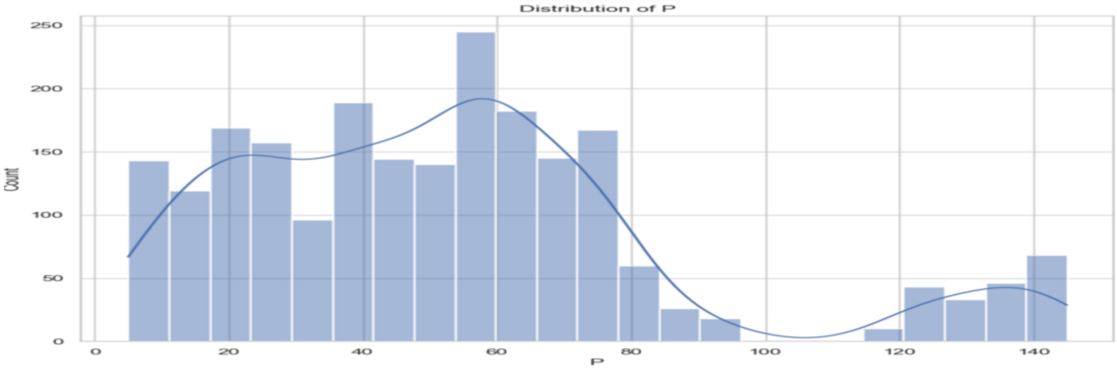


Figure 4.1: Histogram distribution of N, P, K values

The histograms show the distributions of each continuous variable

N, P, K: Some display a bimodal nature (having two peaks), suggesting different groups in the data.

Temperature: Appears to be normally distributed.

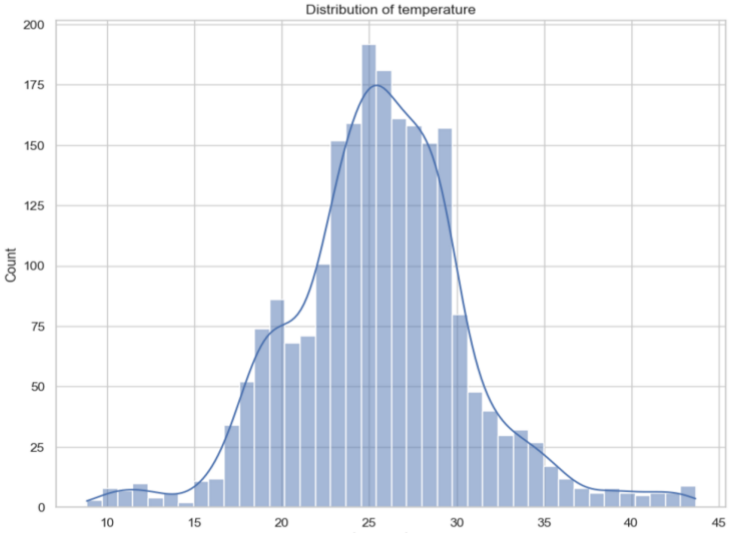


Figure 4.2: Histogram Distribution of temperature

Humidity: Shows left-skewed distribution, with a high frequency of values towards the higher end.

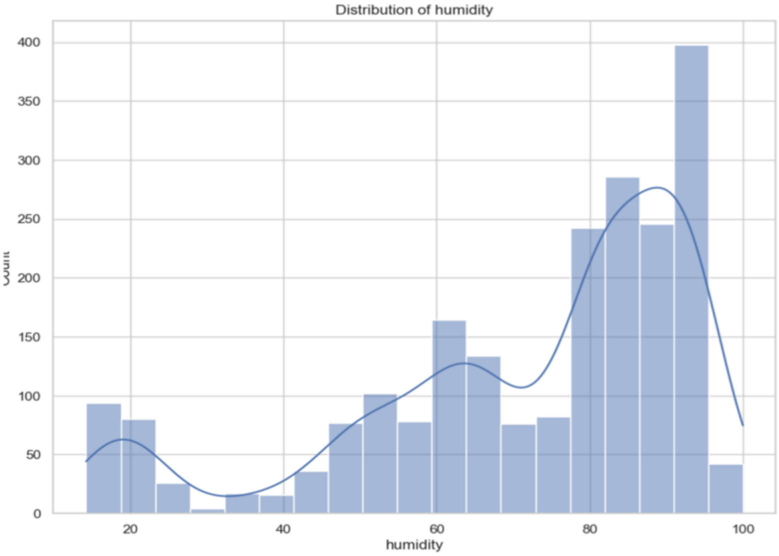


Figure 4.3: Histogram Distribution of humidity

Rainfall: Displays right-skewed distribution, indicating higher rainfall amounts are less common.

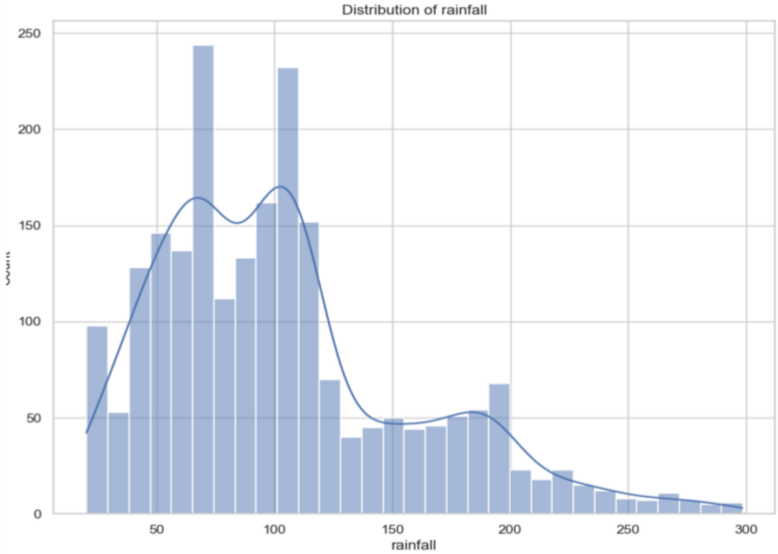


Figure 4.4: Histogram Distribution of rainfall

4.2(b) Data Distributions for Fertilizer Data

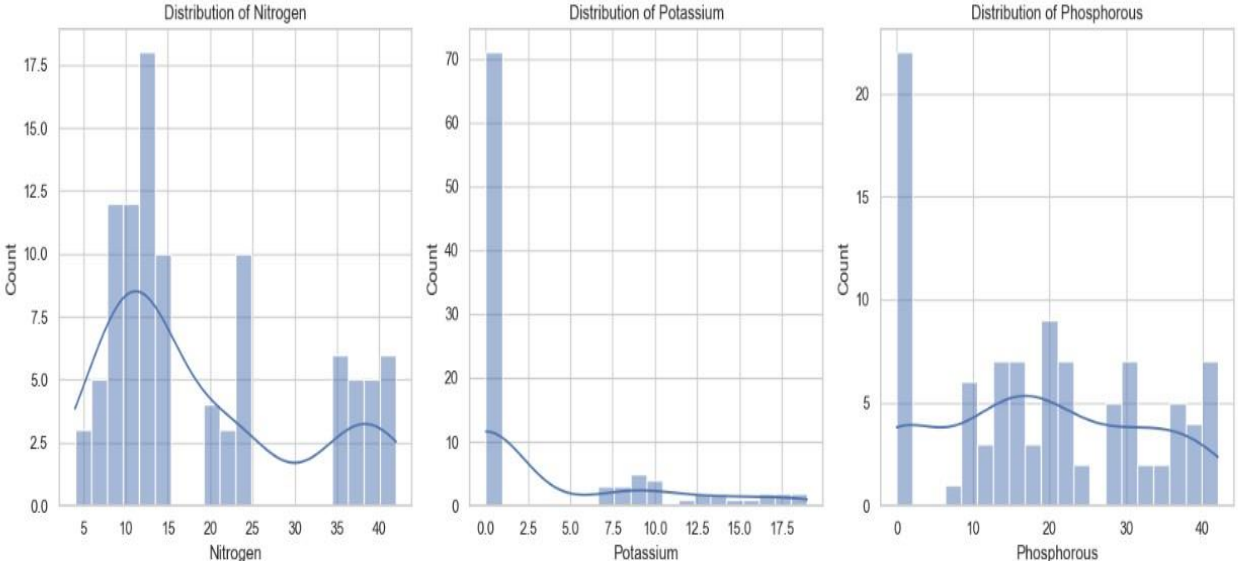


Figure 4.5: Histogram Distribution of fertilizer data

The above graph represents the histogram distribution of the fertilizer data which represents the equal distribution of N, P, K.

Observations:

1. Nitrogen:

   The histogram exhibits a multimodal distribution with several peaks, suggesting multiple common values of Nitrogen in the dataset.

   The line shows these modes as peaks in the probability density, indicating clusters of data points.

2. Potassium:

   The distribution of Potassium is highly skewed towards the lower end, with a sharp peak at the lowest bin. This skewness is evident in the curve, which has a steep drop-off as the values increase.

3. Phosphorous:

   Phosphorous levels are more evenly spread across the range, with a slight concentration at the lower end. The curve for Phosphorous is flatter than that of Potassium, suggesting less skewness.

4.3(a) Outlier Detection for Crop Recommendation

The boxplots provide insights into potential outliers in the continuous variables:

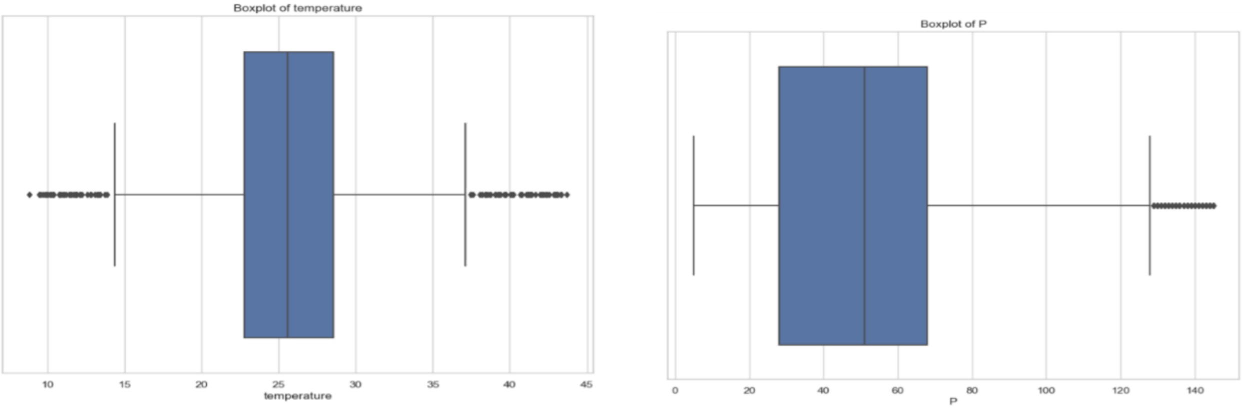


Figure4.6: Boxplot of temperature & phosphorous

Soil Nutrients (N, P, K): These features show some outliers, particularly on the higher end. This could be due to specific crops requiring significantly different nutrient levels.

Temperature: Few outliers are observed, particularly on the lower end.

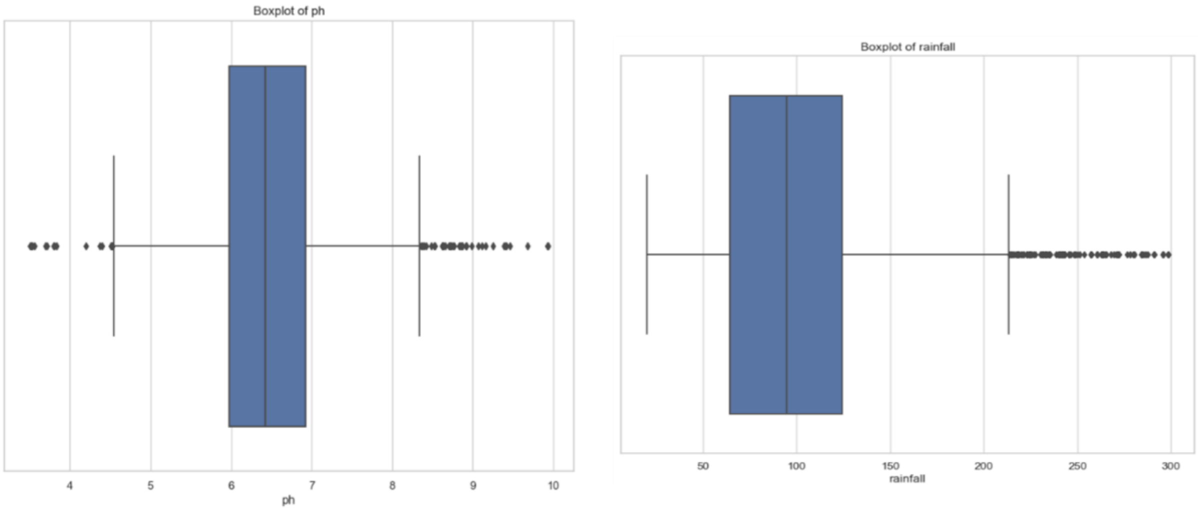


Figure 4.7: Boxplot of ph & rainfall

3.3(b) Outlier Detection for Fertilizer Recommendation

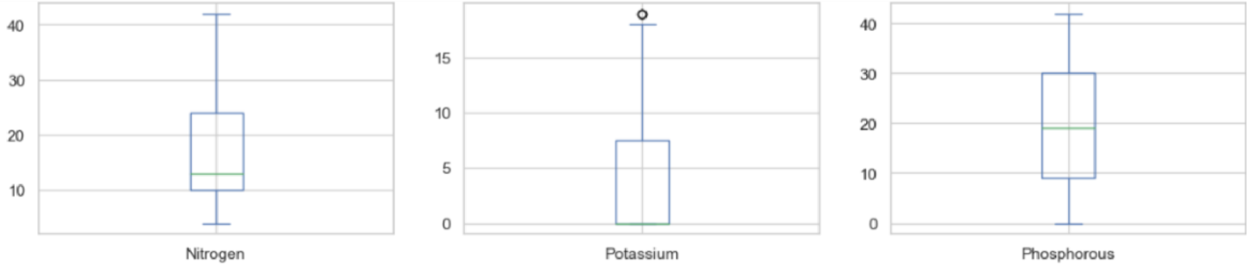


Figure4.8: Boxplot of N, P, K

Nitrogen Box Plot:

The interquartile range (IQR), represented by the box, encapsulates the middle 50% of the Nitrogen data. The median is the central line in the box, dividing the IQR into two equal parts.

Potassium Box Plot:

The box plot for Potassium shows a similar IQR and median. An outlier is noticeable, marked by a circle above the upper whisker, indicating an unusual value that stands apart from the rest of the data.

Phosphorous Box Plot:

The IQR and median for Phosphorous are displayed in a similar fashion to the other nutrients. The distribution of Phosphorous levels appears relatively symmetric around the median.

4.4(a) Correlation Analysis for Crop prediction

The correlation values range from -1 to 1, where

1: indicates a perfect positive correlation,

-1: indicates a perfect negative correlation, and

0: indicates no correlation between the columns.

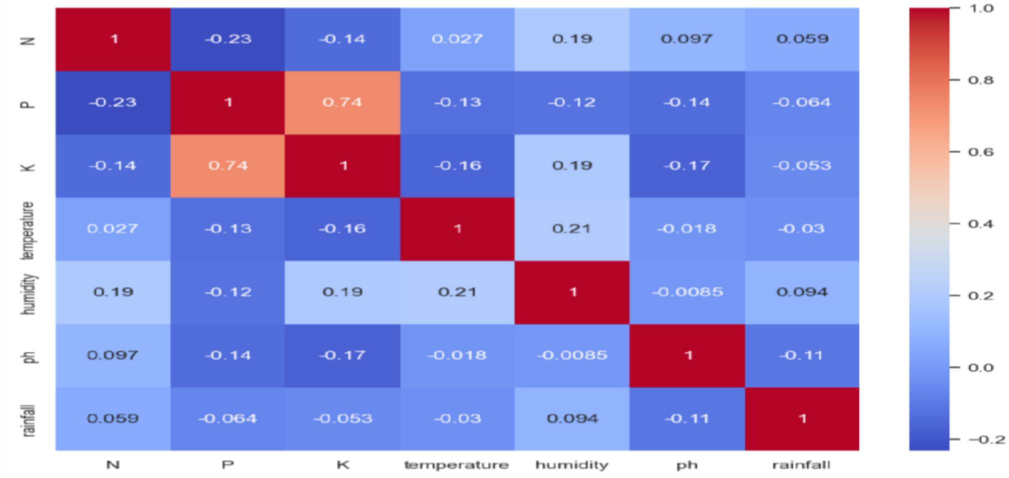


Figure4.9: Correlation Analysis of crop prediction

The above 2D-matrix represents the correlation analysis of the crop prediction

Strong Correlations: There aren't any extremely strong correlations (> 0.8 or < -0.8) observed, which is generally a positive sign for building machine learning models

Moderate Correlations: Some moderate correlations are noted. For example, temperature and humidity show a moderate negative correlation.

Weak Correlations: Most variables show weak correlations with each other, indicating that each provides unique information for the crop recommendation.

4.4(b) Correlation Analysis for Fertilizer prediction

In this case, we can see that there is a negative correlation between nitrogen and potassium, and between nitrogen and phosphorous. This means that as the amount of nitrogen increases, the amount of potassium and phosphorous tends to decrease. A positive correlation between potassium & phosphorous, as potassium increases, the amount of phosphorous tends to increase.

Bivariate Analysis with Data Visualization for crop Prediction

Bivariate analysis is crucial in understanding the relationship between two different variables. Box plots are used to understand the distribution of a particular variable, such as Nitrogen content, across different crop types.

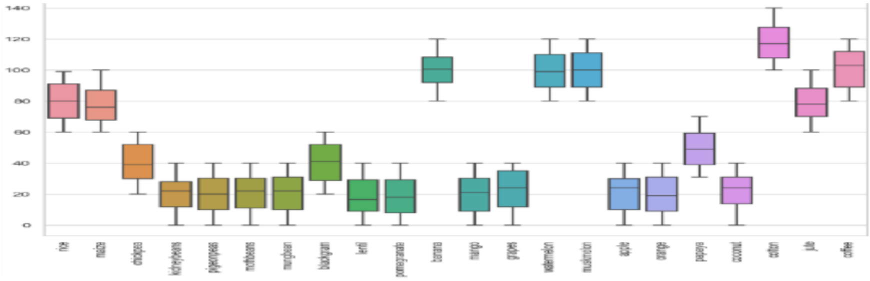


Figure 4.10: Bivariate Analysis of crop prediction

Radar Chart For Nutrient Levels By Crop

1. Each crop label is represented by a filled radar area, showing the average values of nutrients.

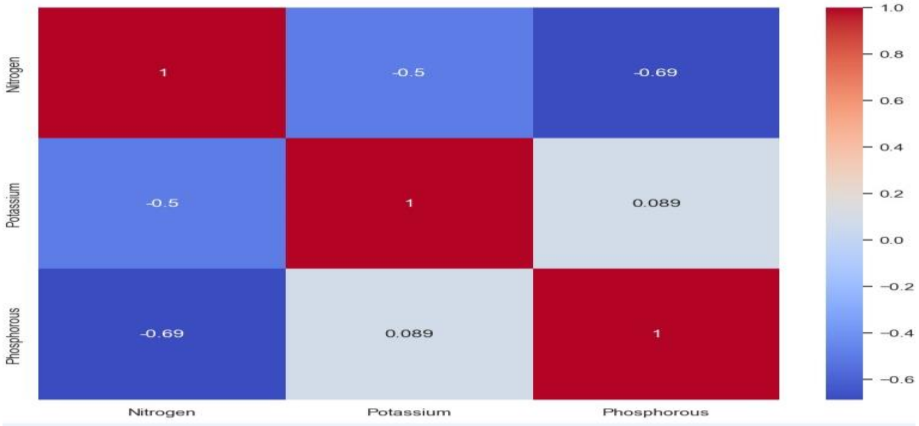


Figure 4.11 Radar Chart of nutrient levels in crop

2. For comparing multiple quantitative variables (nutrient levels) across different categories.
3. The radial axis represents nutrients (‘N’, ‘P’, ‘K’), and the angular axis represents each nutrient.
4. Filled Areas Enclosed By The Lines For Each Crop Label Indicate The Average Nutrient levels,

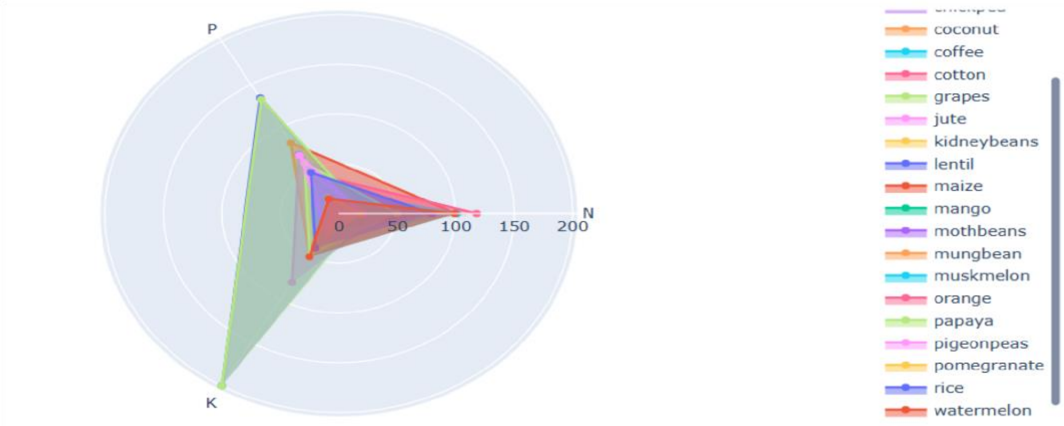
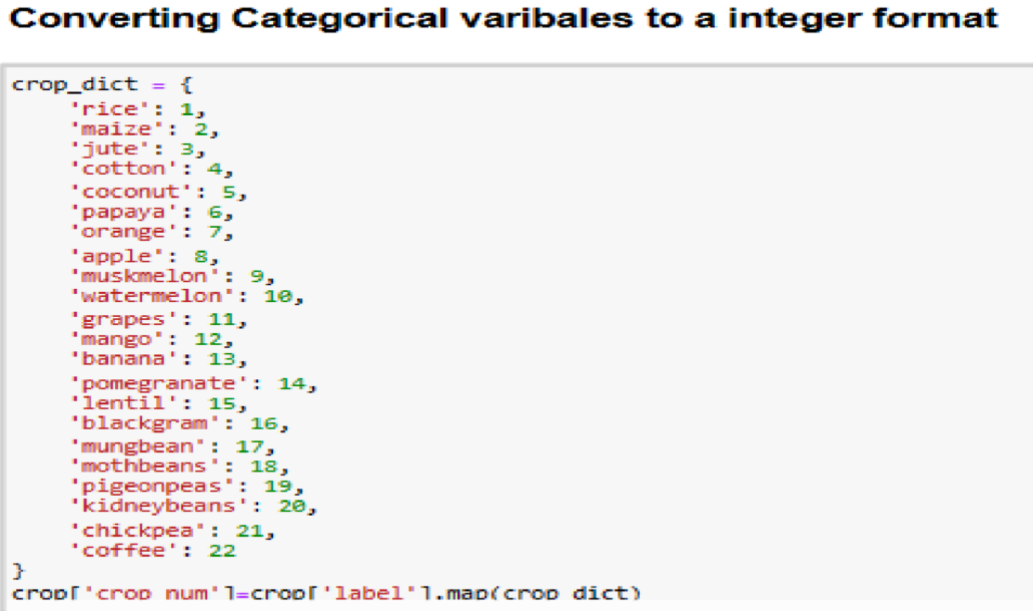


Figure 4.12: Pie Chart for nutrient analysis of crop

4.4(c) Feature Engineering:

4.41 Categorical variables for Crop Recommendation



Converting Categorical varibales to a integer format

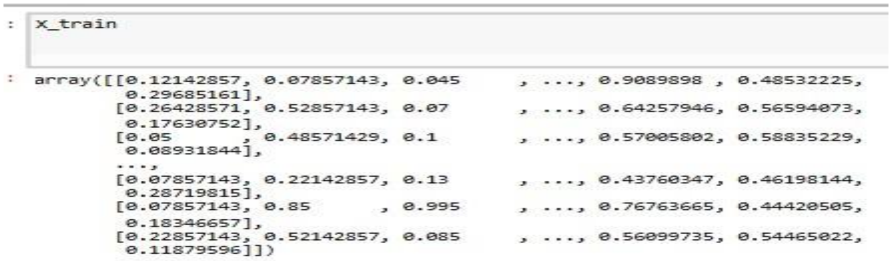
```
crop_dict = {
     'rice': 1,
     'maize': 2,
     'jute': 3,
     'cotton': 4,
     'coconut': 5,
     'papaya': 6,
     'orange': 7,
     'apple': 8,
     'muskmelon': 9,
     'watermelon': 10,
     'grapes': 11,
     'mango': 12,
     'banana': 13,
     'pomegranate': 14,
     'lentil': 15,
     'blackgram': 16,
     'mungbean': 17,
     'mothbeans': 18,
     'pigeonpeas': 19,
     'kidneybeans': 20,
     'chickpea': 21,
     'coffee': 22
}
crop['crop_num']=crop['label'].map(crop_dict)
```

Scale the features using MinMaxScaler:

MinMaxScaler is a feature scaling technique that normalizes each feature to a specified range, typically [0, 1]. It does this by subtracting the minimum value of the feature and then dividing by the range (the maximum value minus the minimum value).

Why Use MinMaxScaler?

1. Normalizing Measurement Scales: In crop recommendation datasets, features like temperature, humidity, and soil pH can have different scales and units. MinMaxScaler ensures that these features with varying ranges don't disproportionately influence model.
2. Improving Model Performance: Many machine learning algorithms perform better when data is on a similar scale. MinMaxScaler can help in faster convergence and improved performance, especially for algorithms like neural networks and k-nearest neighbors.
3. Maintaining Proportions: Unlike some scalers, MinMaxScaler preserves the shape of the original distribution, scaling data points uniformly without reducing importance of outliers.



```
: X_train

: array([[0.12142857, 0.07857143, 0.045     , ..., 0.9089898 , 0.48532225,
         0.29685161],
        [0.26428571, 0.52857143, 0.07      , ..., 0.64257946, 0.56594073,
         0.17630752],
        [0.05      , 0.48571429, 0.1       , ..., 0.57005802, 0.58835229,
         0.08931844],
        ...,
        [0.07857143, 0.22142857, 0.13      , ..., 0.43760347, 0.46198144,
         0.28719815],
        [0.07857143, 0.85      , 0.995     , ..., 0.76763665, 0.44420505,
         0.18346657],
        [0.22857143, 0.52142857, 0.085     , ..., 0.56099735, 0.54465022,
         0.11879596]])
```

Figure 4.13: Array describing minmax scaler

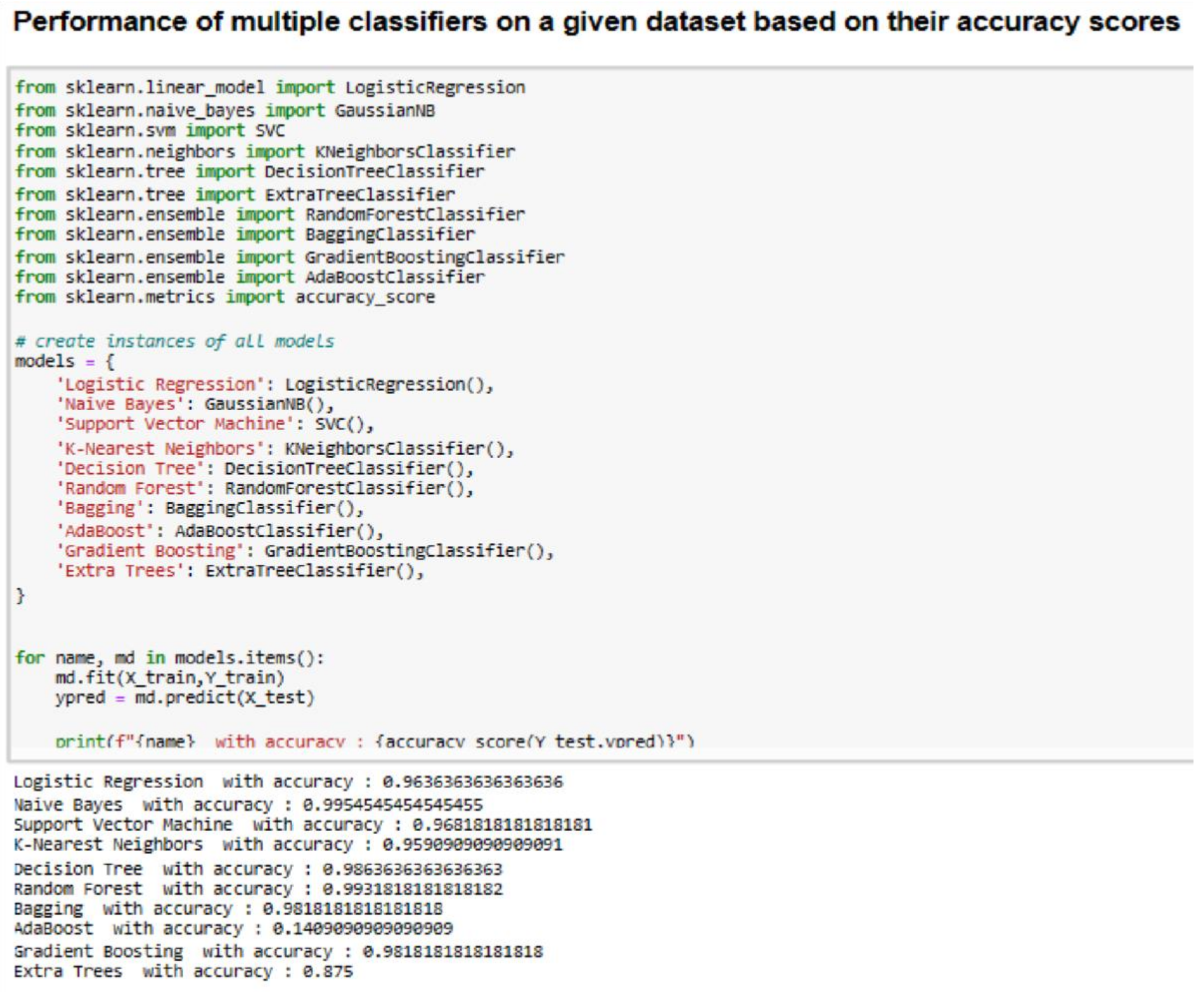
Importance of Standardization for Your Crop Recommendation Project

In the context of a crop recommendation project, standardizing features like temperature, rainfall, and pH levels is crucial due to their varying units and scales. Standardization ensures that all these features contribute equally to the model's predictions, preventing any single feature from dominating due to its variance or unit.

CHAPTER-5

MODEL SELECTION & DEPLOYMENT

5.1 (a) Model Implementation for Crop Recommendation



Performance of multiple classifiers on a given dataset based on their accuracy scores

```
from sklearn.linear_model import LogisticRegression
from sklearn.naive_bayes import GaussianNB
from sklearn.svm import SVC
from sklearn.neighbors import KNeighborsClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.tree import ExtraTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import BaggingClassifier
from sklearn.ensemble import GradientBoostingClassifier
from sklearn.ensemble import AdaBoostClassifier
from sklearn.metrics import accuracy_score

# create instances of all models
models = {
    'Logistic Regression': LogisticRegression(),
    'Naive Bayes': GaussianNB(),
    'Support Vector Machine': SVC(),
    'K-Nearest Neighbors': KNeighborsClassifier(),
    'Decision Tree': DecisionTreeClassifier(),
    'Random Forest': RandomForestClassifier(),
    'Bagging': BaggingClassifier(),
    'AdaBoost': AdaBoostClassifier(),
    'Gradient Boosting': GradientBoostingClassifier(),
    'Extra Trees': ExtraTreeClassifier(),
}


for name, md in models.items():
    md.fit(X_train,Y_train)
    ypred = md.predict(X_test)

    print(f"{name}  with accuracy : {accuracy_score(Y_test,ypred)}")
```

```
Logistic Regression  with accuracy : 0.9636363636363636
Naive Bayes  with accuracy : 0.9954545454545455
Support Vector Machine  with accuracy : 0.9681818181818181
K-Nearest Neighbors  with accuracy : 0.9590909090909091
Decision Tree  with accuracy : 0.9863636363636363
Random Forest  with accuracy : 0.9931818181818182
Bagging  with accuracy : 0.9818181818181818
AdaBoost  with accuracy : 0.14090909090909
Gradient Boosting  with accuracy : 0.9818181818181818
Extra Trees  with accuracy : 0.875
```

1. Naive Bayes (Accuracy: 99.55%):

High Accuracy: With an accuracy of 99.55%, Naive Bayes shows excellent performance in classifying crop types.

Probability-Based: As a probabilistic classifier, Naive Bayes is effective in making predictions based on the likelihood of various outcomes, which is valuable in crop recommendation where multiple factors influence the result.

2. Decision Trees (Accuracy: 98.64%)

   Interpretability: Decision Trees provide a clear visualization of the decision-making process, making it easier to understand how different features contribute to the final recommendation.

   Handling Non-Linear Relationships: They are capable of capturing complex, non-linear relationships between features, which is common in agricultural datasets.

3. Random Forest (Accuracy: 99.32%)

   Robustness: Random Forest, an ensemble of Decision Trees, is more robust and less prone to overfitting compared to a single Decision Tree.

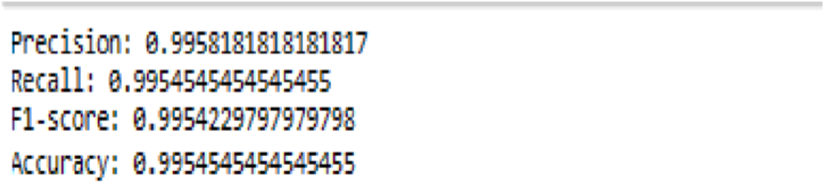
   Handling Large Datasets: It excels in handling large datasets with many features, making it ideal for comprehensive agricultural data.

5.1.1 Naive Bayes

Handling Continuous Data: Gaussian NB is particularly effective when dealing with continuous data. It assumes that the continuous values associated with each feature are distributed according to a Gaussian distribution (normal distribution). This is relevant in agricultural datasets where many features such as temperature, rainfall, and pH levels are continuous and can be assumed to follow a Gaussian distribution.

Fast Model Training and Prediction: Given the potentially large size of agricultural data, the speed of training and prediction is crucial. Naive Bayes provides a faster solution compared to more complex models, making it ideal for rapid analysis and real-time decision-making in crop recommendations.

Good Performance with Small Datasets: Even with a smaller amount of data, Naive Bayes can perform quite well, making it a good choice for projects where the amount of data may be limited.



```
Precision: 0.9958181818181817
Recall: 0.9954545454545455
F1-score: 0.9954229797979798
Accuracy: 0.9954545454545455
```

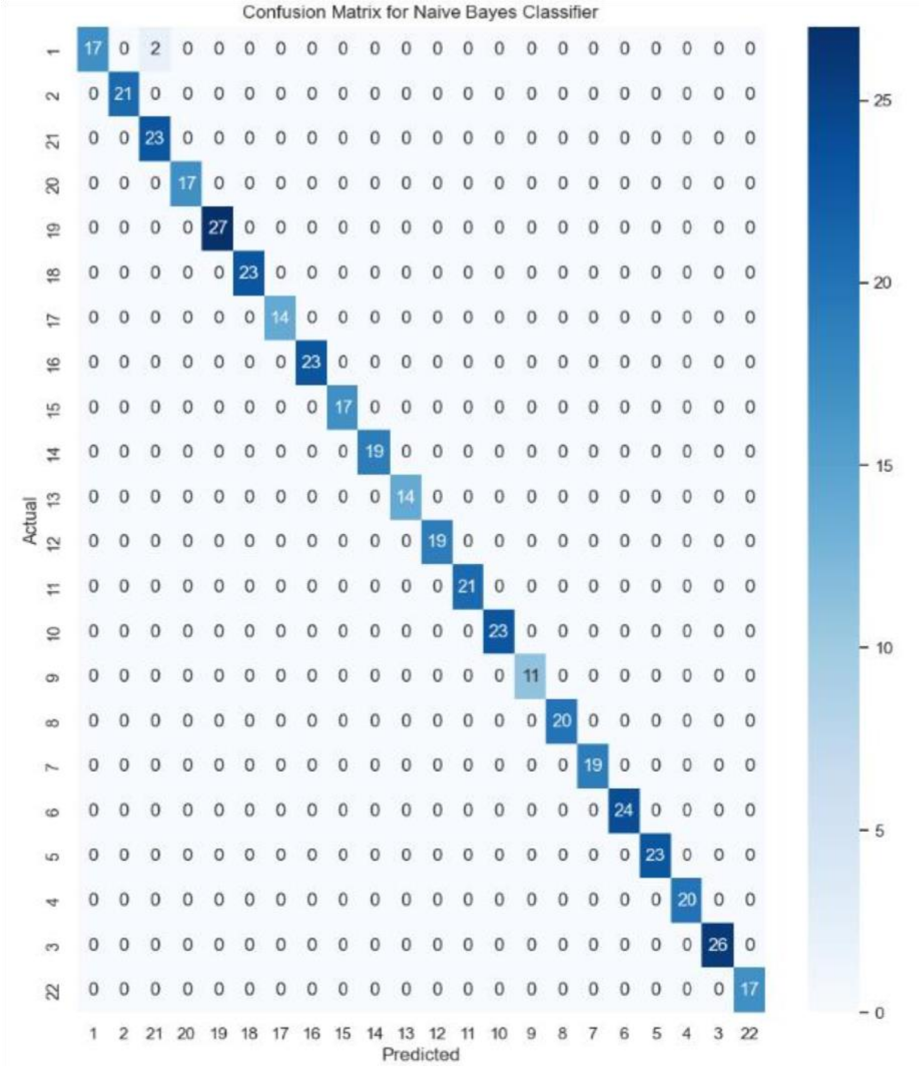
Analysis

High Values on the Diagonal: Indicate good performance for specific crop types.

High Values Off-Diagonal: For example, if a high number appears in the row for crop A and the column for crop B, it means crop A is often misclassified as crop B.

Model Improvement

Certain crops are consistently misclassified, Misclassifications can guide you in refining the features used for training or in tweaking the model parameters.



	1	2	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	22
1	17	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
21	0	0	23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	23	0	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	17	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	19	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	14	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	19	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	21	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0	23	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	19	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	24	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	23	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	26	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	17

Figure 5.1: Confusion Matrix of naive bayes

The above 2D-matrix represents confusion matrix of naive bayes model

5.1(b) Model Implementation For Fertilizer Recommendation

5.1.2 Random Forest Classifier:

Robustness and Versatility: Random Forest is a robust and versatile ensemble learning method, suitable for both classification and regression tasks.

Handling of Complex Interactions: Random Forest can capture complex interactions between features, which is often the case in agricultural datasets where factors such as nutrient levels and soil conditions may interact in complex ways to influence fertilizer requirements.

Reduction of Overfitting: By utilizing multiple decision trees, Random Forest reduces the risk of overfitting, which can be a common problem with single decision trees.

Feature Importance: An inherent benefit of Random Forest is its ability to rank the importance of different features in prediction. In the context of fertilizer recommendation, the model can identify which soil nutrients or conditions are most predictive of the need for a particular type of fertilizer.

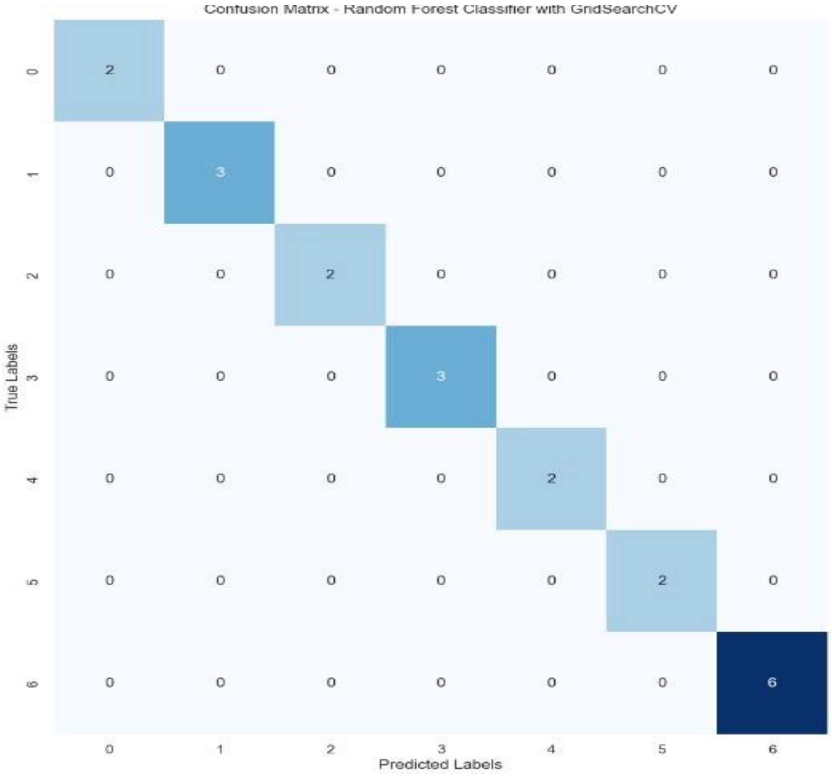
Analysis

Predominance of Zero Off-Diagonal Values: Many off-diagonal cells have zero values, which suggests that there are no misclassifications between many pairs of classes.

Concentration of Errors: The errors are not spread out but concentrated between specific classes, which can indicate similar feature patterns for these classes causing confusion for the model.

Model Improvement

To further improve the model, we should examine the feature importance given by the Random Forest and consider collecting more data for the misclassified classes or reevaluating the features that lead to confusion. Another approach could be to look into more complex model architectures or feature engineering techniques that can capture the nuances between the classes that are being confused.



	0	1	2	3	4	5	6
0	2	0	0	0	0	0	0
1	0	3	0	0	0	0	0
2	0	0	2	0	0	0	0
3	0	0	0	3	0	0	0
4	0	0	0	0	2	0	0
5	0	0	0	0	0	2	0
6	0	0	0	0	0	0	6

Figure 5.2: Confusion Matrix of random forest classifier

The above 2D-matrix represents confusion matrix of random forest classifier model

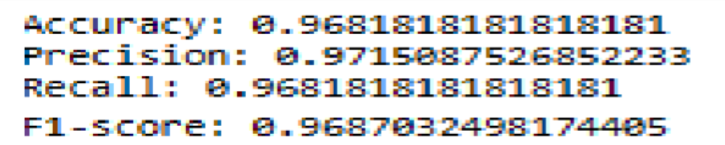
CHAPTER-6
ADVANCED MODEL SELECTION

6.1(a) Model Implementation for Crop Recommendation

6.1.1 Neural Networks

Neural Networks are particularly suited for crop and fertilizer recommendation because they can model complex, non-linear relationships in the data.

They excel in handling large datasets with many features, which is often the case in agricultural datasets where various environmental and soil factors interact in complex ways.



```
Accuracy: 0.9681818181818181
Precision: 0.9715087526852233
Recall: 0.9681818181818181
F1-score: 0.9687032498174405
```

Accuracy (0.9681): Indicates that the model correctly classified about 96.81% of the crop types. High accuracy is crucial for ensuring reliable crop recommendations.

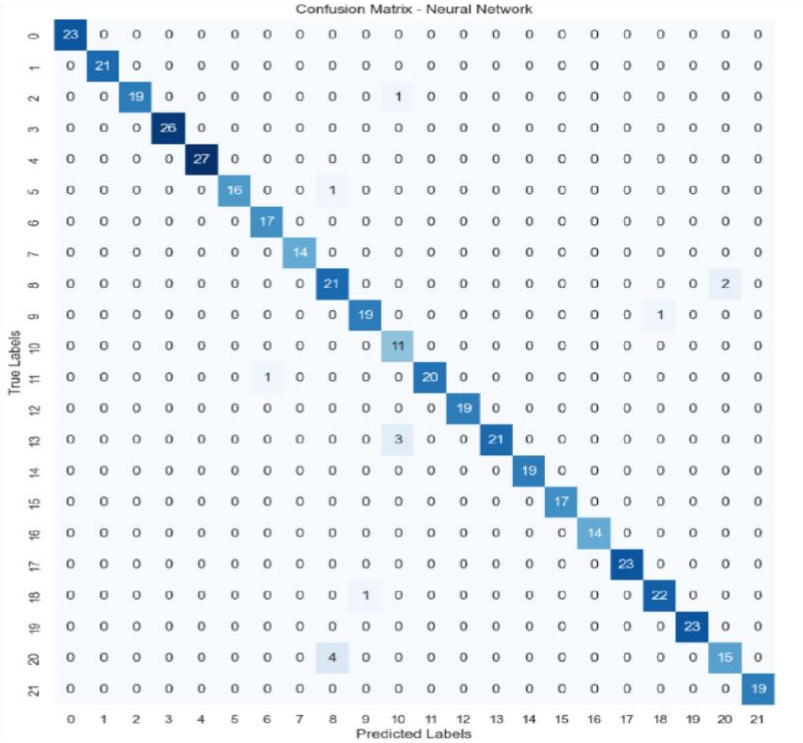
Analysis

High Values on the Diagonal: Such values illustrate strong predictive performance for specific crop types by the Neural Network model.

High Values Off-Diagonal: Notably, there are some off-diagonal values, such as a count of 4 for the predicted label 21 when the actual label was 27, indicating a case of misclassification.

Model Improvement

The presence of misclassifications, although relatively low, indicates potential areas for improvement. It may be necessary to delve deeper into the features correlated with those specific crops to understand why misclassifications occurred. Enhancing the feature set, performing further feature engineering, or adjusting the model's hyperparameters could reduce these errors.



Confusion Matrix - Neural Network

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
0	23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	19	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	27	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	16	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	14	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	21	0	0	0	0	0	0	0	0	0	0	0	2	0
9	0	0	0	0	0	0	0	0	0	19	0	0	0	0	0	0	0	0	1	0	0	0
10	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	1	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	19	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	3	0	0	21	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	19	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	17	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	14	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	23	0	0	0	0
18	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	22	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	23	0	0
20	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	15	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	19

Predicted Labels

True Labels

Figure 6.1: Confusion Matrix of neural network for crop recommendation

The above 2D-matrix represents confusion matrix of Neural Network model for crop recommendation

6.1(b) Model Implementation for Fertilizer Recommendation

6.1.2 Neural Networks

Capability to Model Non-linear Relationships: Neural Networks, particularly MLP Classifier with 'relu' activation function and 'adam' solver, excel in modeling non-linear and complex relationships that are often present in agricultural datasets.

F1 Score: An F1 score of 0.966 is particularly impressive as it represents the harmonic mean of precision and recall. This high F1 score implies a balanced model that maintains both high precision and recall, indicating fewer misclassifications and missed cases, which is crucial for making accurate fertilizer recommendations.

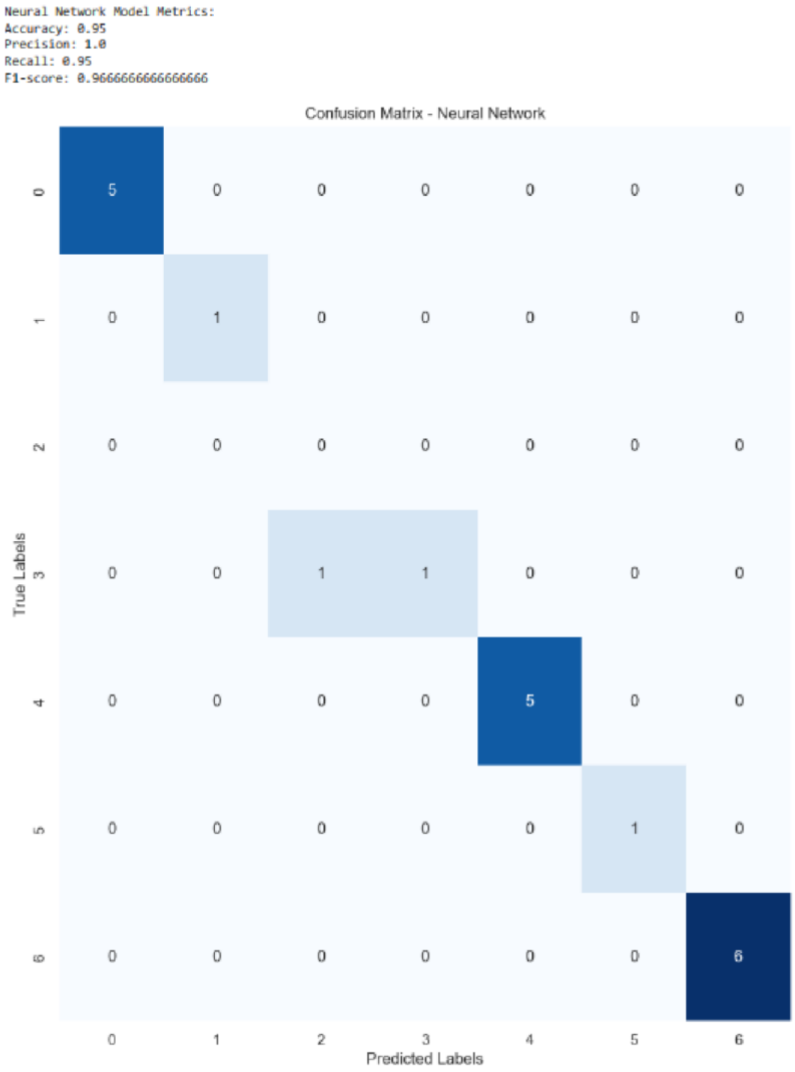
Analysis

Predominant Diagonal Values: The high values on the diagonal for certain classes suggest that the model is particularly effective at correctly classifying these classes.

Sparse Off-Diagonal Values: The presence of few off-diagonal values indicates that there are relatively few misclassifications overall, which is a positive indicator of model performance.

Model Improvement

The misclassifications that do occur provide opportunities for model improvement. Enhancements might include feature engineering to better distinguish between these classes, collecting more representative data for the underperforming classes, or adjusting the model



```
Neural Network Model Metrics:
Accuracy: 0.95
Precision: 1.0
Recall: 0.95
F1-score: 0.9666666666666666
```

Confusion Matrix - Neural Network

	0	1	2	3	4	5	6
0	5	0	0	0	0	0	0
1	0	1	0	0	0	0	0
2	0	0	0	0	0	0	0
3	0	0	1	1	0	0	0
4	0	0	0	0	5	0	0
5	0	0	0	0	0	1	0
6	0	0	0	0	0	0	6

Figure 6.4: Confusion Matrix for neural network of fertilizer recommendation

The above 2D-matrix represents confusion matrix of Neural Network model of fertilizer recommendation

CHAPTER-7
RESULT

Explanation and Interpretation of Model Performance

7.1 (a) Results for Error calculations Crop Recommendation

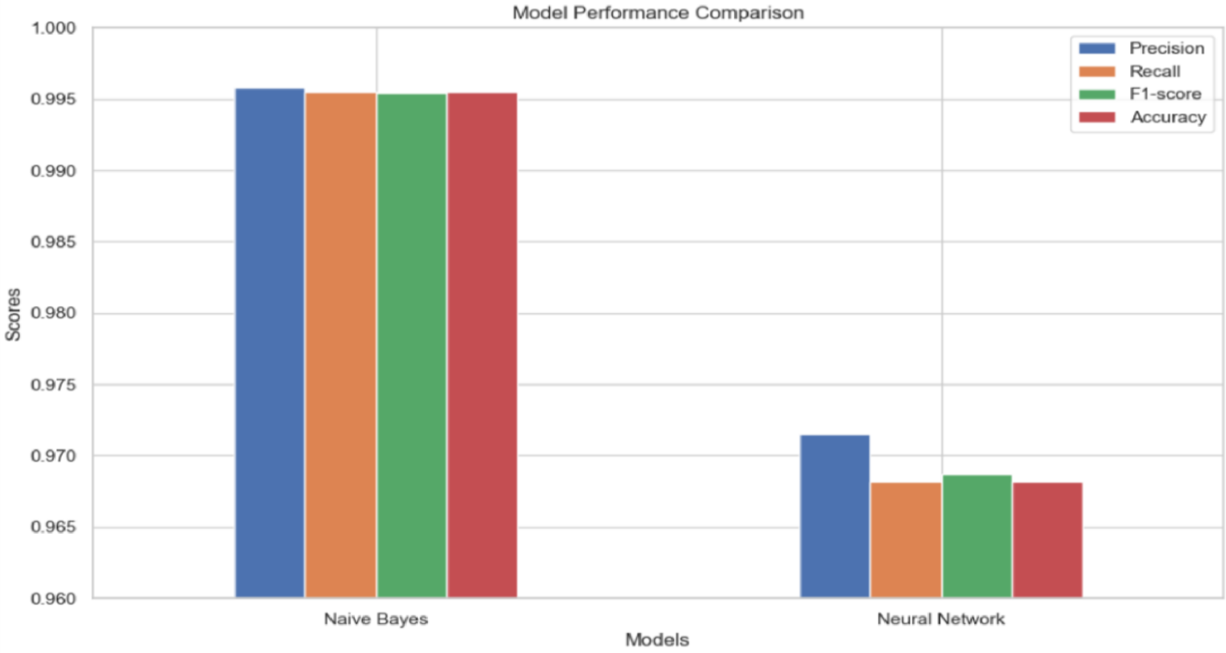


Figure 7.1: Result for error calculations in crop recommendation

Precision: Precision is the ratio of correctly predicted positive observations to the total predicted positives. High precision indicates a low false positive rate. Naive Bayes has the highest precision, suggesting it's best at correctly identifying crops without labeling many incorrectly.

Recall: Recall (Sensitivity) is the ratio of correctly predicted positive observations to all observations in the actual class. Naive Bayes again scores highest, indicating it is most capable of finding all relevant cases (all suitable crop types).

F1-score: The F1 Score is the weighted average of Precision and Recall. Therefore, this score takes both false positives and false negatives into account. The highest F1-score of Naive Bayes implies a balance between precision and recall.

Accuracy: This is the ratio of correctly predicted observation to the total observations. Naive Bayes has the highest accuracy, indicating it correctly identifies crop types most often.

7.1 (b) Results for Error calculations Fertilizer Recommendation

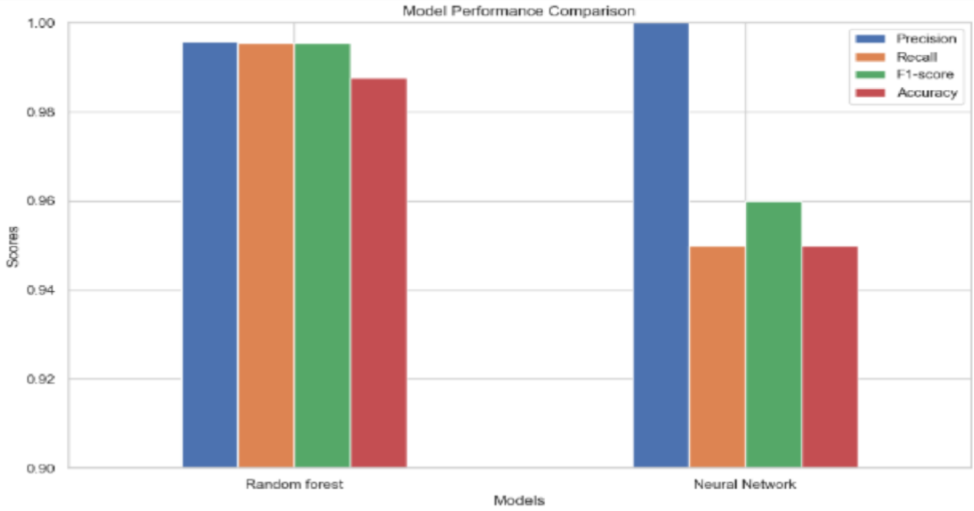


Figure 7.2: Result for error calculations in fertilizer recommendation

PRECISION:

Random Forest: Precision of 0.9958 suggests a very low rate of false positives. This model is excellent at correctly identifying the right fertilizer types without many errors.

Neural Network: A perfect precision score of 1.0 indicates no false positives in its predictions, which is exceptional.

RECALL:

Random Forest: The recall of 0.9955 indicates that this model is almost perfect in identifying all relevant instances of the correct fertilizer types.

Neural Network: A recall of 0.95, while still high, suggests it misses a few relevant cases compared to Random Forest.

F1-SCORE:

Random Forest: The F1 score of 0.9954 is very high, showing a strong balance between precision and recall.

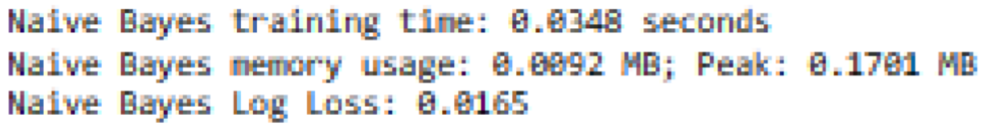
Neural Network: An F1 score of 0.96, slightly lower than Random Forest, indicates a marginally less balanced performance in precision and recall.

ACCURACY:

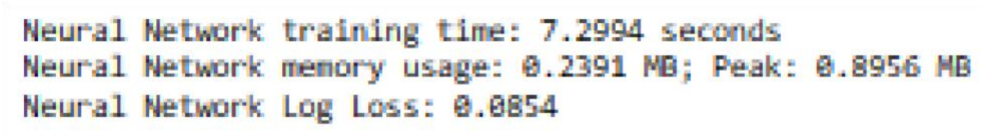
Random Forest: An accuracy of 0.9876 means it correctly identifies the right fertilizer types in most cases.

Neural Network: The accuracy of 0.95 is high, but lower than Random Forest, suggesting a slightly reduced overall prediction capability.

7.2 (a) Results for Computational Crop Recommendation



```
Naive Bayes training time: 0.0348 seconds
Naive Bayes memory usage: 0.0092 MB; Peak: 0.1701 MB
Naive Bayes Log Loss: 0.0165
```



```
Neural Network training time: 7.2994 seconds
Neural Network memory usage: 0.2391 MB; Peak: 0.8956 MB
Neural Network Log Loss: 0.0854
```

Comparative Analysis of Naive Bayes and Neural Network Model

In our crop recommendation project, we conducted a comprehensive comparison between two machine learning models: Naive Bayes and Neural Network. This comparison extends beyond standard metrics like accuracy, precision, recall, and F1 score, to provide a deeper understanding of the practical implications of each model.

Training Time Comparison

The time taken to train a model is crucial, especially in large datasets or when frequent retraining is necessary.

Observations: Naive Bayes, typically, requires significantly less training time compared to Neural Networks, making it efficient for scenarios demanding quick model updates or limited computational resources.

Memory Usage Evaluation

We evaluated the memory consumption during the training phase of each model, a critical aspect in resource-constrained environments.

Observations: Neural Networks often consume more memory due to their complex architecture, while Naive Bayes, with its simpler structure, has a lower memory footprint, suitable for deployment with restricted memory.

Log Loss Analysis

Log loss measures the confidence of the predictions, penalizing false classifications more heavily if the model is very confident in its incorrect predictions.

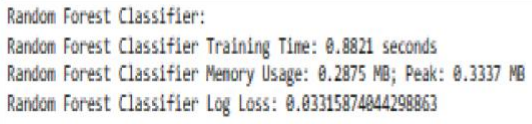
Observations: A lower log loss is indicative of a model with reliable and confident predictions. This aspect is vital in applications like crop recommendations, where uncertain predictions can lead to significant consequences.

Model Complexity And Size

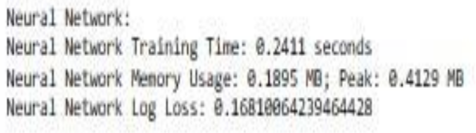
We also considered the complexity and size of each model, as a simpler model with fewer parameters is easier to deploy, especially in environments with limited computational resources.

Observations: Naive Bayes, being fundamentally simpler, has fewer parameters and thus lower complexity. In contrast, Neural Networks, due to their deep architectures, are more complex and resource-demanding.

6.2 (b) Results for Computational Fertilizer Recommendation



```
Random Forest Classifier:
Random Forest Classifier Training Time: 0.8821 seconds
Random Forest Classifier Memory Usage: 0.2875 MB; Peak: 0.3337 MB
Random Forest Classifier Log Loss: 0.03315874044298863
```



```
Neural Network:
Neural Network Training Time: 0.2411 seconds
Neural Network Memory Usage: 0.1895 MB; Peak: 0.4129 MB
Neural Network Log Loss: 0.16810064239464428
```

RANDOM FOREST CLASSIFIER:

Training Time: The model took approximately 0.891 seconds to train.

Memory Usage: The training process consumed 0.2875 MB of memory, with a peak memory usage of 3.0337 MB.

Performance Metrics:

Precision, Recall, and F1-Score: These metrics are perfect (1.00) across all classes, which include various categories like 'Fourteen-Thirty Five-Fourteen' and 'Twenty-Eight-Twenty-Eight'. Support: This column indicates the number of true occurrences of each class in the dataset. The classes have varying support, with some having only 1 instance and others having more, up to 5.

Macro Average: Averages the performance metrics for each class, and these are also perfect (1.00), indicating uniform excellence across all classes despite the imbalance in their representation.

Weighted Average: Takes into account the support for each class, and again, the metrics are perfect (1.00).

NEURAL NETWORK:

Training Time: The Neural Network model took significantly longer to train, with 2.8411 seconds.

Memory Usage: It required more memory, with 0.1895 MB used and a peak of 4.4129 MB.

Performance Metrics:

Precision, Recall, and F1-Score: Like the Random Forest Classifier, the Neural Network also achieved perfect scores across all classes.

Support: The distribution of true occurrences is the same as for the Random Forest.

Macro and Weighted Averages: Both are perfect at 1.00.

CHAPTER-8

STRENGTHS, WEAKNESS & IMPROVEMENTS:

8.1 (a) Analysis for Crop Recommendation

Model Performance Analysis for Crop and fertilizer Recommendation: The performance of various machine learning models on the Crop and fertilizer Recommendation project has been visualized in the bar chart, showing precision, recall, F1-score, and accuracy. Let's dive into a detailed analysis of each model's strengths, weaknesses, and potential improvements, and conclude with why Naive Bayes is outperforming the others.

Naive Bayes:

Strengths: Naive Bayes is simple, fast, and performs exceptionally well when the assumption of feature independence holds. It's particularly effective in high-dimensional spaces, which might be the case with our crop dataset.

Weaknesses: The assumption of feature independence rarely holds true in real-world data, which can limit its performance in some scenarios. Naive Bayes also struggles with zero-frequency problems where it assigns zero probability to unseen features/labels combinations.

Improvements: Applying smoothing techniques like Laplace estimation can help with zero frequency problems. Feature engineering to reduce dependency among variables can also improve performance.

Neural Network:

Strengths: Neural Networks are highly flexible and can model complex non-linear relationships, making them suitable for the diverse and complex data typically found in crop and fertilizer recommendation datasets.

Weaknesses: They require a large amount of data to train and are not as interpretable as simpler models. They can also overfit if not properly regularized.

Improvements: Using dropout, regularization techniques, and proper validation strategies can help prevent overfitting. Neural architecture search can optimize the network structure.

Why Might Naive Bayes Perform Better?

The high performance of Naive Bayes suggests that the dataset likely has features that are relatively independent, a condition where Naive Bayes thrives. Its simplicity also helps to avoid overfitting, a problem that more complex models can sometimes face. Additionally, if the data has many categorical features or features following a probability distribution that Naive Bayes assumes, it can outperform other models.

8.1 (b) Analysis for Fertilizer Recommendation

Random Forest:

Strengths: Excellent for handling varied data types and complex relationships. Robust against overfitting due to ensemble nature.

Weaknesses: Can be computationally intensive. Interpretability is less straightforward than simpler models.

Improvements: Feature selection and hyperparameter tuning can enhance performance. Simplifying the model could improve interpretability and reduce computational load.

Neural Network

Strengths: Highly adaptable to complex, non-linear relationships. Exceptional in large datasets and diverse feature sets

Weaknesses: Prone to overfitting. Requires substantial data for training. Less interpretable.

Improvements _: Regularization techniques and proper validation can reduce overfitting. More data and improved architecture could enhance performance.

Analysis of Model Performance

The superior performance of Random Forest in this context can be attributed to several factors:

Data_Characteristics: The dataset might have features and relationships well-captured by the decision trees in Random Forest.

Overfitting_Avoidance: Random Forest naturally avoids overfitting better than Neural Networks, especially if the dataset isn't massive.

Complexity_Balance: Random Forest strikes a balance between handling complex relationships and not becoming too complex itself, unlike Neural Networks which can become overly complex.

The Neural Network's slightly lower scores might be due to overfitting, the need for more data, or complexity that isn't necessary for this specific dataset.

CHAPTER-9

CONCLUSION

Comprehensive Conclusion on Model Selection for Crop and Fertilizer Recommendation

Model Suitability: Model choice should be tailored to the dataset's unique traits and the project's objectives. Random Forest offers a balanced approach with high accuracy and low risk of overfitting, suitable for varied scenarios.

Random Forest Advantages: It handles complex data relationships effectively, with less computational demand, making it ideal for agricultural datasets where performance and usability are key.

Neural Networks Considerations: While Neural Networks are adept at managing complex datasets, they require more data and meticulous hyperparameter tuning to fully leverage their capabilities.

Naive Bayes Appropriateness: For datasets with independent features and possibly small or imbalanced data, Naive Bayes is efficient and simple, though caution is advised for more complex datasets where feature independence is not assured.

Cross-Validation Importance: Ensuring that the model performs well on unseen data through cross-validation is crucial, affecting the choice between simpler models like Naive Bayes and more complex ones like Neural Networks.

Final Decision Factors: The decision on the optimal model, be it Naive Bayes or Neural Networks, hinges on a balance between predictive performance, model transparency, and the dataset’s complexity.

REFERENCES

1. Prediction of crop yield and fertilizer recommendation using machine learning algorithms by K.N. Sanghvi
2. Rakesh Kumar, M.P. Singh, Prabhat Kumar and J.P. Singh proposed utilization of seven AI procedures i.e., ANN, Decision Tree, Random Forest, GBDT and Regularized Gradient Forest for crop determination.
3. Machine Learning: Applications in Indian Agriculture, 2018:S. Bhanumathi Et al.
4. Recommender System lecture python notebook by Prof. Junwei Huang
5. [Crop Recommendation using ML] (https://ieeexplore.ieee.org/document/9734173)
6. Kaggle : (https://www.kaggle.com/datasets/atharvaingle/crop-recommendation-dataset)
7. Fertilizer Kaggle: (Plant Disease Classification - ResNet- 99.2% | Kaggle)
8. Microsoft Bing AI chat